ZAMBIA INFORMATION COMMUNICATION TECHNOLOGY (ICT) JOURNAL

Volume 2 (Issue 2) (2018) Pages 16-27

Design and Development of an optimal algorithm to assign applicants to suitable teaching positions

Mumbi Chishimba
School of Science, Engineering and Technology
Mulungushi University
chishimba.mumbi@gmail.com

Douglas Kunda
School of Science, Engineering and Technology
Mulungushi University
dkunda@mu.edu.zm

***Abstract* - Resource allocation has always been an area of interest in the area of computing. Areas such as machine learning provide many solutions to the problem of resource allocation. The issue addressed in this study is the issue of optimal allocation of applicants (teachers) to positions in schools where their area of specialization will be better applied. We develop an algorithm that is able to allocate applicants to schools based on the applicant qualifications and the school's needs. We use the principles of resource allocation and machine learning in order to create an application to allocate applicants to schools where their qualifications are most suited. Methods used include classification techniques in machine learning, regression and similarity comparison. For the identification of subjects an applicant in proficient in, various machine learning algorithms are tested to determine which machine learning algorithm was best. The actual process of identifying which applicant qualifies for a school position is also tested against sequential assignment if applicants to schools. The results of this was the algorithm based assignment of applicants to schools which produced more accurate assignment of applicants to schools than the sequential assignment of applicants. The aim of this algorithm is to provide a solution to that automatically identifies the needs (subjects) of a school, determine which needs have a higher priority, identify the qualifications of the applicants and assign the applicants to the school according to the school's needs and the applicant's qualifications.**

Keywords: Machine-learning, Resource-Allocation, Decision-Trees, Classification

I. Introduction

E-Government offers great opportunities for social, political and economic development especially in developing countries like Zambia. As a result of this, e-government is becoming a greater field of interest in the country as it provides many opportunities for development in Zambia. E-Government is defined as "the use of information and communication technologies and its application by the government for the provision of information and public services to the people" [1]. The area of interest for this study is to assess how best to use E-Government to improve the process of government deciding on how to place new applicants in teaching and health staff in order to best utilize their skill set. To accomplish this task, techniques such as data mining and machine learning are used in order to develop algorithms that will assist in the selection process. Data mining is useful in these cases because data mining may be used to extract useful information from large amounts of data [2]. Based on the data that is mined, it is possible to predict a number of things related to the data including, in our case, predicting which place a new applicant is best suited.

This paper discussed the process of developing an applicant selection solution that selects applicants and assigns them to schools where they are most needed. The study begins with the literature review which delves into some machine learning techniques and also goes through a method of measuring similarity. After the literature review, the next portion discusses the implementation of the applicant selection and the algorithms that are implemented in the process of applicant selection. The next section is the discussion in which the implementation and testing of the solution are expounded upon after which the conclusion comes.

Zambia, as of 2017 has an unemployment of 7.79% [3]. Considering that the "normal" rate of unemployment is around 5% [4], it is safe to state that Zambia has considerably high levels of unemployment. The implication of this is that since every year produces new graduates, a considerable number of the graduates are not able to immediately find employment especially in fields such as teaching. Coupled with the high rates of corruption in Zambia [5], the process of selection of applicants who are given jobs by the government is also something that is subject to compromise which would negatively affect the applicants who wish to acquire employment.

In order to curb the rates of corruption in the process of hiring and to create a level playing, merit driven approach to the process of employing teachers in government, it is then necessary to develop a system that automatically assigns employees to teaching positions based on merit driven factors and also assign those teachers where they would be most needed. A machine driven algorithm assures transparency and efficiency in the process of assigning in that the only factors it would take in are the qualifications of the applicant and match them to the needs of the school and also select the most qualified applicant for the job.

The aim of this study is to provide a solution to that automatically identifies the needs (subjects) of a school, determine which needs are to have a higher priority, identify the qualifications of the applicants and assign the applicants to the school according to the school's needs and the applicant's qualifications.

The objectives included:

1. To develop an algorithm that assesses schools and discover which subject specializations are needed most.
2. To determine which subjects an applicant is specialized in
3. To develop an algorithm that will assess applicants in order to place the best qualified applicants in the positions

For this study, an application comprising three components was developed. The components include the component that processes applicants to predict which subjects they are proficient in, a component to order school needs and assign applicants to the schools and a UI component to facilitate the usage of the application.

The results of the testing of the application indicated that the application was better at allocation of the applicants than the sequential assignment of applicants to the schools. A number of algorithms were tested for the component that predicts the subjects that an applicant is proficient in and it was discovered that the Decision Tree Classifier yielded better performance in a shorter period of time.

II. Related Works

The problem considered in this paper is the problem of assigning applicants to schools based on the needs of the school and the proficiency of the applicants themselves. This means that there is a need for an algorithm to identify which of the schools needs are the most pressing, which of the applicants are qualified to teach at the school and assign the most qualified of the applicants to the school.

In [6], the authors study "a practical problem of resource assignment and optimal human resource allocation scheduling." In this paper, the authors aim to minimize the down time of some of the employees in order to maximize output from the employees. In order to achieve this, the authors [6], analysed the employee's competencies (skills) in needed to complete tasks.

The authors of [7] take a different approach to the issue of assignment of staff in that in their paper, they tackle the issue of resource allocation in staff to work shifts. They make use of lexicographic goal programming (LGP) taking into account the approach of fairness. The authors set a number of parameters such as the shift assignment time, number of shifts assigned per day, qualification of the assigned staff, off days, number of shifts assignable for the type of employee and the maximum workload for an employee. These parameters are used to determine the success of the algorithm. In [8], the authors present a "mathematical programming model that assigns workers to tasks, rotates workers between tasks and determine the training schedule" with the objective of "minimize the total costs including training cost, flexibility cost, and productivity loss cost" in a manufacturing environment.

The solution proposed in this study consist of an algorithm that assigns teachers to teaching positions in schools. This means that the teacher's qualifications are assessed to find out where to classify them, order the schools according to which school needs more teachers and then assign the teachers to the schools. This essentially means that the algorithm needs to be split into three portions to satisfy each of the requirements. The main difference between the reviewed literature and the proposed solution is that proposed solution aims to make use of machine learning techniques to solve the problem of teacher allocation. For instance regression is used in the process of ordering the school needs while decision trees will be used to classify the qualifications of the teachers. By combining these machine learning techniques, proposed solution carries out the task of assigning teachers to teaching positions to schools. An advantage of using machine learning is that it is easier to adapt to evolving business needs since the models can easily be trained with data that suits the current needs of an organisation.

A. Machine learning techniques

Machine learning is defined as the scientific field of study where "machines learn for experience" [9]. From this, a machine, with the help of a machine learning model, gets some input and, from the input using the learning model, performs the act of learning to produce some output. There are three general categories of machine learning namely supervised learning, unsupervised learning and reinforcement learning.

Supervised learning

Supervised learning algorithms are algorithms where the "system must 'learn' inductively a function called a target function, which is an expression of a model describing the data" [9]. In its essence, a supervised learning algorithm takes in some input variables and their corresponding outputs in order to train a model to predict the values (predictions) of future inputs [10]. Input variables (data) whose output is known are called training data (set) and it is with this data that a model is trained to predict results [9]. There are two kinds of learning tasks under machine learning namely classification and regression. Under supervised learning, "Classification models try to predict distinct classes, such as e.g. blood groups, while regression models predict numerical value" [9]. Some techniques used in supervised learning include Decision Trees (DT) [11], Rule Learning and Instance Based Learning (IBL) techniques such as k-Nearest Neighbours (kNN), Genetic Algorithms (GA), Artificial Neural Networks [12] and Support Vector Machines [9].

Classification

Classification is a machine learning algorithm are techniques which are used to predict the group that a data instance belongs to base on the features of the data instance [13] [14] [15]. This typically involves using the knowledge/features present in the data and in order to determine which class which the data instance belongs to. For instance, using features such as height, hip-to-shoulder ratio and average hair length, a classification model may be trained to take these variables for a data instance and classify it as either male or female. Classification may be performed using single-label learning where a data instance is assigned one class label or multi-label learning where a data instance may be assigned a number of class labels [16] [17]. Figure 1 and Figure 2 illustrate the process of training and predicting data in classification.

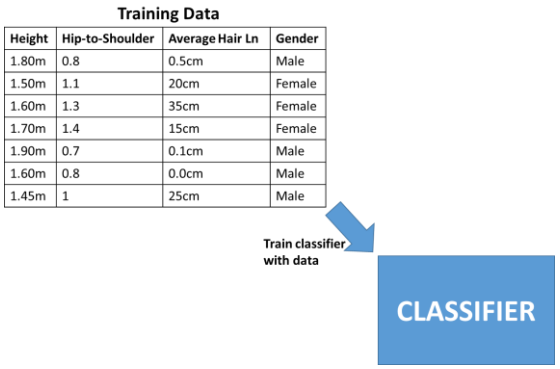

Figure 1: Training of classifier with data

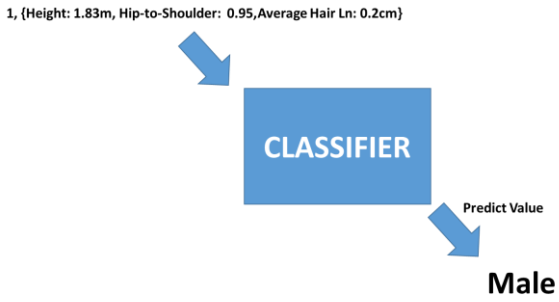

Figure 2: Prediction of class value of data

A more advanced form of classification is a branch called Multi-Label classification [18]. In the traditional form of classification, the algorithm can only assign one class label to a data instance. In multi-label classification on the other hand, it is possible to assign multiple class labels to the data instances when predicting [19].

Regression

Regression analysis is a widely used analysis method for analyzing multi-factor data [20]. Regression analysis is widely used to express the relationship between variables of interest in an equation [20]. An example of a form of regression that is used is linear regression which is the most widely used to determine correlation [21].

Unsupervised learning

Unsupervised learning is the type of learning where the system itself tries to discover the hidden structure of data and the associations between the data [9], [22]. Under unsupervised learning, the training data instances have no corresponding labels [9]. Popular unsupervised learning methods include Association Rule Mining [23] and Clustering [24].

Association rule mining

Association rule mining is defined as the process of "identifying all rules from the transaction data that satisfy the minimum support and confidence constraints" [25][26][27][28]. In its essence, Association rule mining "consists in finding interesting 'if-then' rules between feature value combinations in a dataset" [23]. In its essence, an association rule denotes the rule between two or more features of a data set. For instance, if we have A→B where A and B are feature values in a data set, we can conclude to say then when A appears, B also appears [23].

Clustering

Clustering, as the name suggests, is an algorithm where data is partitioned subgroups or clusters in order to identify the similarities between the data clusters [24]. Unlike supervised learning algorithms like classification, clustering has no predefined classes for its data [24]. The idea in clustering is that each data point in a cluster is more similar to other data points in a cluster than data points in other clusters. Figure 3 and Figure 4 illustrate how clustering is performed

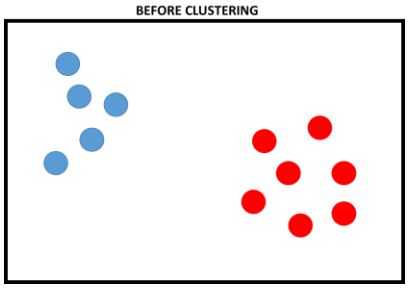

Figure 3: Before clustering

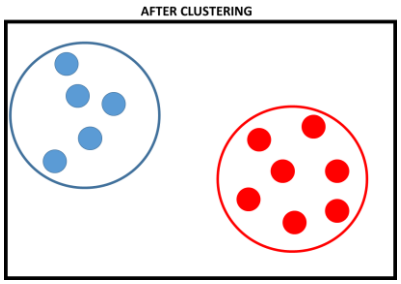

Figure 4: After clustering

Clustering includes methods such as k–medoids, k-means and hierarchical clustering [29]. Commonly used clustering algorithms include partitioned clustering, hierarchical clustering and density based clustering [24].

- *Partitioned clustering*: In partitioned clustering method, the datasets having 'n' data points partitioned into 'k' groups or clusters where each cluster has at least one data point and each data point belongs to a cluster [24]. In this clustering method, there is a need to define the number of clusters for the dataset before partitioning begins [24].
- *Hierarchical clustering*: In this clustering method, there is no need to define the number of clusters before partitioning begins. This method decomposes the data points using either the top-down or bottom-down approaches [24].
- *Density based clustering*: Partitioned based clustering and hierarchical clustering are unsuitable for discovering cluster or arbitrary shapes [24]. Density based clustering on the other hand can efficiently handle outliers and arbitrary shaped cluster data.

Reinforcement learning

Reinforcement learning techniques are those in which systems attempt to learn through interacting with the environment in order to maximize some notion of cumulative reward [9] [30]. In this method of learning, the agent observes the state it is in and receives a reward based on the state at a particular time [30]. As a result, reinforcement learning is

suitable for fields such as robotics where engineers can design sophisticated and hard-to-engineer behaviours [31]–[33].

Data Mining and Machine Learning Methods

There are a number of techniques that can be used in the process of data mining and machine learning. Some of these methods include but are not limited to:

- *Decision trees* [34] : These are tree flow-chart like structures which contains nodes which may have child nodes [2]. Essentially, decision trees use the tree-like structure to represent the attributes and possible outcomes [35]. All internal nodes of a decision tree have two or more children and each leaf node of the decision tree represents a class label. Decision trees are found favourable because the decision trees themselves represent rules which are also easily understandable by humans [2]. Decision trees are based on the top-down attribute selection approach, DT algorithm first identifies the initial node of the tree and then it uses a series of 'if then' steps along with the attribute selection method in order to complete the predictive model [35]. Common decision tree algorithms include ID3, CART and C4.5 [35].
- *Artificial Neural Networks (ANN):* Artificial neural networks are computer classification model that are considerably popular in machine learning [36]. An artificial neural network is a mathematical model which consists of an interconnected group of artificial neurons, and it processes information using a connectionist approach to computation [2]. Neural networks are adaptive in nature meaning that they change structure during the learning phase [2]. Neural networks are classified as supervised learning algorithms and are arranged into three layers namely the input layer, hidden layer and the output layer [35]. A big criticism of neural networks is that they have poor performance [35]. A great advantage on the other hands is that it can generate robust models that are more accurate, highly adaptive and are very flexible in noise tolerance.
- *Naïve Bayes:* This classifier is suited to situations where the dimensionality of the inputs is high [37]. The Naïve Bayes classifier can often outperform more sophisticated classification methods because of its simplicity [37]. The Naïve Bayes classifier makes a strong independence of features/classes i.e. it assumes that the presence of absence of a feature is independent from that of any other feature [38]. Naïve Bayes is based on Bayesian theorem and can be written as:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Equation 1: Bayesian theorem algorithm

- *Random Forest:* Random forests are based on decision tree classifiers and are also known as Classification and Regression Trees (CART) [36]. Random forests use a number of decision trees to improve a prediction models performance [34]. In the collection of decision trees, each tree is created by first selecting a small group of independent variables randomly to split on at each node and the calculating the best split [6]. Because of how complex random forest models can be, (hundreds of randomized trees) and the voting mechanism employed to select the beat split, random forests can be considered a black-box model, as there is no simple way to explain its predictions [39].
- *Bayesian Belief Network:* These types of networks follow Bayes theorem but unlike the Naïve Bayes classifier, these types of networks do not make the assumption that variables are independent of one another [35]. Bayesian Belief networks are directed acyclic graph (DAG) where the nodes have one-to one correspondence with the attributes and where any node in the network can be selected as a 'class attribute' [35] .
- *Support Vector Machines:* Support vector machines are supervised learning algorithms that are based on the transformation of a mathematical function by another mathematical function called the 'kernel', by which one identifies the greatest distance between the most similar observations that are oppositely classified [36].They have three main properties namely: 1) "SVM constructs a maximum margin separato--a decision boundary with the largest possible distance to example points," [40] 2) "SVM creates a linear separating hyperplane, but it has the ability to embed the data into a higher-dimensional space, using the so-called Kernel Trick and," [40] 3) SVM is a nonparametric method. It retains training examples, and potentially needs to store them all. In practice, it often ends up retaining only a small fraction of the number of examples; sometimes as few as a small constant times the number of dimensions [40].

B. Decision trees

Decision trees prove to be flexible in that from the literature reviewed, decision trees [41][42][43][35][1] proved to be a relatively effective and popular method of problem solving with some studies [42] finding that decision trees produced more accurate results over other algorithms. A noted advantage of decision trees is that decision treed are simple and prompt data classifiers [1]. They are also noted as being popular and powerful for both classification and prediction [2] and also since decision trees represent rules, they are easily readable by human beings. A classic example of decision tree is action is that of trees generated from the IRIS dataset [44] as shown in Table 1 and Figure 5

Table 1: Sample of IRIS dataset.

Sepal Length	Sepal Width	Petal Length	Petal Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
5	3.4	1.5	0.2	setosa
4.4	2.9	1.4	0.2	setosa

Sepal Length	Sepal Width	Petal Length	Petal Width	Species
4.9	3.1	1.5	0.1	setosa
5.4	3.7	1.5	0.2	setosa
4.8	3.4	1.6	0.2	setosa
4.8	3	1.4	0.1	setosa
4.3	3	1.1	0.1	setosa

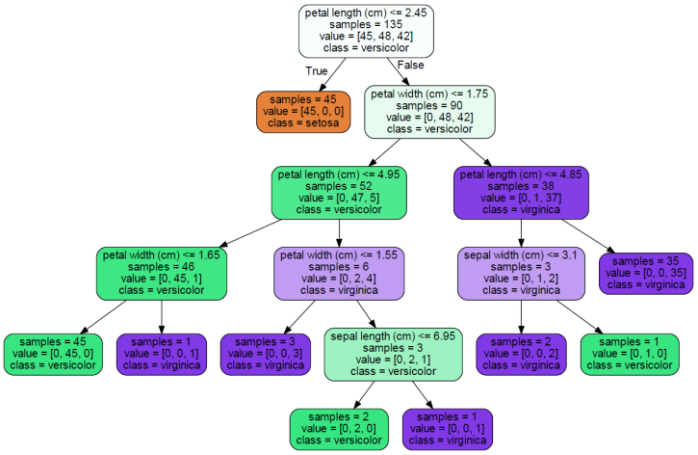

Figure 5: Example of tree generated from the dataset

Artificial Neural Networks on the other hand were found to be better at analysing non-linear relationships [36] and in general provide better classification results. Three problems identified with neural networks on the other hand are that neural networks perform poorly for unbalances data, validation is insufficient to provide a satisfactory error rate as the training set becomes larger and the selection of hidden layers is difficult given the relationship between computing time and higher predictability [36]. Naïve Bayes and Support Vector Machines in one study [35] produces weaker results that those produced by Decision Trees and Artificial Neural Networks. In another study comparing an Improved Decision Tree, a Decision Tree, and Artificial Neural Network and a Naïve Bayes algorithm, [45], it was found that that Naïve Bayes generally produced worse results than its counterpart algorithms. The one area where Naïve Bayes provides an advantage is in cases where a small amount of data is needed to estimate the parameters and where the input dimensionality is high [35]. This study will involve evaluating the algorithms that are in place in order to try and see which algorithm/combination of algorithms will best be able to place applicants in places where their skill set will be most needed. This means collecting data from the relevant sources, sorting out the data and adapting an algorithm to sorting out the task.

There are various types of decision tree algorithms available for use including ID3 [46], [47], C4.5 [48], CART [14], CHAID [49] and MARS [50] to name a few.

- *Iterative Dichotomiser 3 (ID3)*: This is a decision tree algorithm used to generate a tree from a data set [47]. The algorithm learns by constructing the decision tree in a top-down fashion based on the divide and conquer strategy [47]. The algorithm employs a greedy search through the given sets to test each attribute at every tree node [46]. This algorithm also uses information gain to choose the splitting attribute and only accepts categorical attributes in building a tree model [37]. Another feature of ID3 is that it does not support pruning of nodes [37]. ID3 is applied to calculate logistic performance and is applicable in the field of computer crime forensics and to diagnose and predict diseases [47, p. 3].
- *C4.5*: This algorithm is serially implemented like the ID3 algorithm with the added benefit that this algorithm supports pruning of the decision tree where internal nodes may be replaced with a leaf node in order to reduce the error rate [47]. This algorithm is capable of handling both categorical and continuous data [51]. The C4.5 algorithm splits the attribute values into two partitions based on the selected threshold such that all the values above the threshold in such that all the values above the threshold as one child and the remaining as another child [52] in order to handle continuous attributes in the building of the decision tree. To select the splitting attribute values used in the creation of the decision tree, C4.5 uses gain ratio impurity [47]. This algorithm is the most used algorithm for building decision trees and is suitable for real world problems because it deals with numeric attributes and missing values [47].
- *CART*: The Classification And Regression Trees [51] algorithm is another decision tree algorithm considered in this study. This algorithm is capable of handling both categorical and continuous attributes in the building of decision trees [51]. The CART algorithm used the GINI index as an attribute selection method when building the decision tree [37]. The algorithm also uses cost complexity pruning to remove the unreliable branches from the decision tree to improve its accuracy [37].
- *CHAID*: Chi-squared Automatic Interaction Detection (CHAID) is another decision tree algorithm. CHAID, as a market segmentation method, is more sophisticated method than other forms of multivariate analysis [53]. The CHAID algorithm works in steps namely (1) best partition for each predictor is found, (2) predictors are compared and the best ones are chosen and (3) The data is subdivided according to the chosen predictor. The strengths of the CHAID method include (1) the Chi-Square method used for attribute, (2) nominal types and interval variables can be considered as predictors, (3) continuous variables can be chosen as criterion variables and (4) a criterion variable can be established [53].
- *MARS*: Multivariate Adaptive Regression Splines (MARS) is a flexible procedure to organize relationships between a set of input variables and the target dependent variables [54]. The MARS algorithm is non-linear and non-parametric regression method [55] is based on the divide and conquer strategy where training data sets are "partitioned into separate piecewise linear segments (splines) of differing gradients (slope) " [54]. The MARS algorithm "divides the space of predictors into multiple knots and then fits a spline function between these knots" [56]. Also, the MARS algorithm makes no assumptions about the underlying functional relationship between the input variables and the output is required [55].

Table 2 shows a comparison of the decision tree algorithms discussed:

Table 2: Comparison of the algorithms

	ID3	C4.5	CART	CHAID	MARS
Attribute Selection	Infor-mation gain [47]	Gain Ratio [47]	GINI Index [37]	Chi-Square [53]	MARS-plines[16]
Attributes Accepted	Cate-gorical [57]	Cate-gorical and cont-inuous [57]	Cate-gorical and cont-inuous [58]	Cate-gorical and cont-inuous [49]	Categ-orical and cont-inuous[56]
Pruning algorithm	Based on the classif. error [59]	Based on classif-ication error [59]	Cost Comp-lexity [59]	The CHAID algorithm is able to expand each node to two or more branches as well as to produce the final tree; there is no need for pruning [60]	The basic functions with the least cont-ribution are eliminated using backward deletion [56]
Handles Missing Values	No [46]	Yes [57]	Yes [14]	Yes	Yes
Tree Struc-ture	Multi tree [59]	Multi -tree [59]	Binary [59]	Multi-way[61]	Binary[62]

A measure that will be used in the comparison of applicants to the school position specifications is Jaccard Similarity. Jaccard similarity is a measurement method that that is used to compare the similarity level of two different data sets [63]. The equation below shows the formula for calculating Jaccard similarity

$$jaccardsim(A,B) = \frac{P(A \cap B)}{P(A \cup B)}$$

The measuring of Jaccard similarity between two data sets comes from the division of a common set of features by the total number of features between the two sets of data. From this calculation, the data sets with the most similar objects get a higher score than data sets that do not have a lot of similar objects. To apply Jaccard similarity to this study, the set of subjects that a school requests for will be compared with the list of applicants and the subjects that the applicants are qualified to teach in order to find out which applicant subjects best align with the set of subjects that the school requests for. The applicant with the highest Jaccard score will then be assigned to the school.

III. Methodology

The research methodology employed is the experimental research methodology. In this type of research, a researcher of a particular topic identifies a set of variables to be experimented on then tries to determine if a change on one variable will affect other variables in the phenomena being observed [64]. This is suited for data mining based research projects because one of the greatest steps used in data mining is the identification of variables that will be used in the mining process.

An example of a model applicable to this study is the model proposed by [65]. This is because of the clear flow illustrated by the model as well as the applicability to this type of study. Figure 6 shows the model proposed by [65]

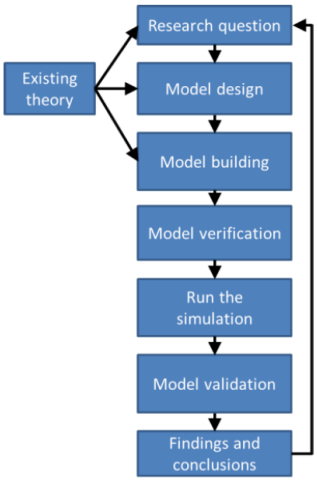

Figure 6: Model used for the study [65]

The model consists of seven steps as defined by [65] which include:

- *The research question* where the research questions suitable to the study are identified,
- *The model design* where a specification of the target to be modelled is specified,
- *The model building* where where the model designed is implemented (built),
- *The model verification* where the model that is built is executed to test whether or not the model is working as it should,
- *The results simulation* where virtual experiments are run under different conditions,
- *The model validation* where the process of confirming the results of the simulation with empirical data,
- *The findings/conclusion* where responses to the research questions are presented,

[66] proposes a simulation testing model which is also conceptually used in this study. Figure 7 shows a simplified version of the modelling process as illustrated by [66].

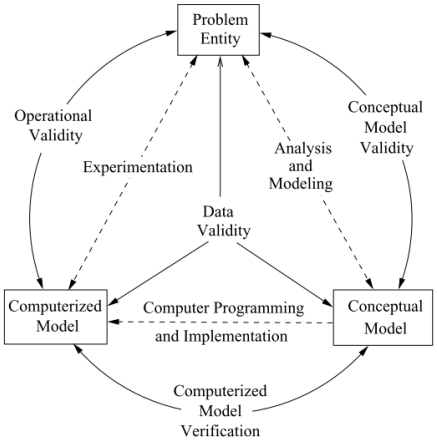

Figure 7: Simplified version of the modelling process [66]

This modelling process primarily consists of three main results namely the problem Entity, a Conceptual Model for the problem entity and a computerized model based on the

conceptual model. From the Problem Entity, the process of analysis and modelling of the problem are performed and the result of the analysis and modelling is a conceptual model. Once the conceptual model is obtained, it is then implemented (in our case, programmed) and the result is a computerized model. Having obtained the computerized model, operational validity is performed from where inferences about the problem entity are obtained by conducting computer experiments on the computerized model [66].

For this study, the problem entity inform the research objectives while the conceptual model is obtained from the results of the model design phase which also confirms the conceptual model validity. The computerized model is obtained from the results of the model building phase where the solution is implemented and by obtaining the computerized model, computerized model verification is also performed. Operational validity is performed during the model verification and validation phase after which inferences about the problem entity are obtained.

To analyse the data, it will be necessary to compare a number of algorithms in order to see which algorithm performs better. This means that comparing the rate of success of the algorithms and other factors such as efficiency of the algorithms also plays a big role in the study.

IV. Algorithm and Results

A. *Description of algorithms*

The solution to the problem involved the development of three main components namely the component to process the applicants, the algorithm to process the various schools and the algorithm to assign the teachers to the schools. The algorithms are discussed below:

B. *Teacher selection algorithm*

The teacher selection algorithm begins by first of all filtering out the applicants depending on the criteria of preference. For instance, applicants may be chosen from a specific province only or they may be chosen to fall within a specified age bracket or a combination of both. Applicants may also be limited to specific districts of choice. Once the applicants and filtered from the system, the applicants will then be pushed through the selection algorithm which is responsible for identifying the subjects that the teachers are most competent in and will then assign these subjects to the student as their primary teaching subjects. The algorithm to assign subjects to teachers uses the One vs. the Rest Classifier which uses a Decision Tree Classifier for the actual prediction. In order to perform the subject specialisation predictions for the list of applications, applicants are first of all pulled from the data source. The applicant's details that are necessary include the applicant identification and the applicant's most successful grades. The applicant records are then each pushed through the One vs. The rest decision tree based classifier in order to predict which subjects the applicant is more proficient in. After these are predicted, the proficient subjects are then appended to the applicant record and are stored to be used later. Figure 8 is the algorithm that is used in the process of prediction.

Algorithm 1 Process applicant qualifications

```
1: procedure ProcessApplicants()
2:     applicants ← db.applicantList
3:     classif ← OneVsRestClassifier(DecisionTreeClassifier())
4:     applicantsAndQualifications ← newList()
5:     for applicant in applicants do
6:         applicantsAndQualifications.add(classif.predict(applicant))
7:     return applicantsAndQualifications
```

Figure 8: Algorithm to process applicant qualifications

C. *School and school subject sorting algorithm*

The next part of the solution involves the use of an algorithm for process the school needs and assign priority as to which school has the first priority in the process of assigning teachers. This involves ordering the requested subject sets for each school by their priority and then ordering the schools to determine which school gets a higher priority in the assigning of applicants. The structure the school requests is illustrated below:

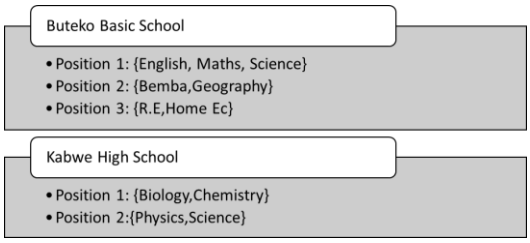

Figure 9: Schools with requested positions

As seen from Figure 9, there are two schools namely Buteko Basic School and Kabwe High School. Buteko Basic School has three positions available with each position needing multiple subjects while Kabwe High School has two positions with each position having multiple subjects as well. For the sake of discussion, each position will be referred to as a subject set hence forth. Under optimum circumstances, the desired outcome for Buteko Basic School is to have three applicants assigned to it where each applicant perfectly satisfied the requirements for the positions they are assigned to. The algorithm will have to first of all get a unique set of subjects for each of the schools and order the subjects according to their priority. The algorithm shown in Figure 10 illustrates how the solution obtains a unique set of subjects from each of the schools:

Algorithm 2 Get unique list of subjects from set of school subjects

```
1: procedure GetUniqueSubjects()
2:     schools ← newList
3:     for school in schoolsWithSubjectSets do        ▷ E.G: (School:
   KTC,Subjects:[(Math,Eng),(Sci,Bem)])
4:         schools.add(school, uniquesubjectSet)      ▷ Final:
   (School:KTC,Subjects:[Math,Eng,Bem])
5:     return schoolsWithSubjectSets
```

Figure 10: Get unique list of subjects from the school positions

For each of the unique subjects obtained in Figure 10, the pass ratio for each subject over a number of previous years is calculated. The pass ratio formula is shown in the equation below:

$$y = \left(\frac{p}{p+f}\right)$$

Equation 2: Pass ratio formula

Where p = number of pupils who passed in the year and f is the total number of pupils who failed

After calculating the pass ratio, simple linear regression is then used to determine whether or not there is an overall improvement in performance of the school in a particular subject and how much that improvement is or vice versa. The algorithm used to calculate the pass ratio and the linear regression are shown in Figure 11

Algorithm 3 Get unique list of subjects from set of school subjects and calculate their regression gradient

```
procedure ProcessSchoolSubjects(schoolsWithSubjectSets)
    schoolProcessedSubjects ← newList
    for school in schoolsWithSubjectSets do
        subjectPerformanceList ← newList
        for subjectPerformance in school.subjectPerformance do
            for year in subjectPerformance do
                subjectDetail ← year, (pass/(pass + fail))
            subgrad ← Gradient(LinearRegression(subjectDetail))
            subjectPerformanceList.add(subject, subgrad)
        schoolProcessedSubjects.add(subjectPerformanceList)
    return schoolProcessedSubjects
```

Figure 11: Get unique list of subjects from set of school subjects and calculate their regression gradient

The linear regression is primarily used to determine whether the subject's performance is improving of getting worse and to what degree each is happening. Once these are obtained, there is then a need to order the unique subjects by their performance gradient in order to better understand which subject is performing better that the other and which one gets a higher priority. Figure 12 shows the algorithm used to perform the subject ordering.

Algorithm 4 Order the unique subjects by their regressed gradient value

```
procedure OrderSchool(schoolProcessedSubjects)
    sorted ← False
    for school in schoolProcessedSubjects do
        for subject in school do
            if subject.subgrad > next(subject.subgrad) then  ▷ If the subject in the list's gradient is greater than the next
                swap subject and next(subject)
                sorted ← True
    if sorted is true then OrderSchool(schoolProcessedSubjects)
    return schoolProcessedSubjects
```

Figure 12: Order the unique subjects by their regressed gradient value

The next step is to order the original school positions according to the subject priority ordering performed in Algorithm 4. For example, if R.E in Figure 9 has a higher priority, position 3 will then be moved to a higher priority in the queue. Figure 13 illustrates the process of ordering the school positions.

Algorithm 5 Order the original school subject sets by the ordered unique set of subjects for the school

```
procedure OrderSchoolByOrderResults(schWithSubjectSets, schProcSubjects)
    sorted ← False
    for schNeeds,schProc in schWithSubjectSets,schProcSubjects do
        for subjectNeeds,subjectProc in schNeeds,schProc do
            sim1 ← jaccardsimilarity(subjectNeeds, subjectProc)
            sim2 ← jaccardsimilarity(next(subjectNeeds), subjectProc)
            if sim2 > sim1 then
                move subjectNeeds.subjectSet to front of subject set list
    if sorted is true then OrderSchoolByOrderResults(schWithSubjectSets,schProcSubjects)
    return schWithSubjectSets
```

Figure 13: Order the original school subject sets by the ordered unique subject set

Once the school positions are sorted in order, the next step is to now assign applicants to the positions.

D. *Applicant-to-position assigning algorithm*

Once the process of sorting through the schools is completed, the next phase is to then assign the teachers that were processed to the schools that were sorted. To do this, a third algorithm is employed. This algorithm works by iterating through all the schools and assigning the teacher most suited to a school's requested subject. Since the requested subjects are grouped (e.g. Kashikishi Secondary School may want two teachers. One proficient in Chemistry and Biology, the other proficient in Mathematics, Science and Bemba), the algorithm has to compare the qualifications of a teacher versus the requirements of a school. For instance, if Buteko Basic School in Luanshya wants a teacher which can teach Geography and Religious Education, the algorithm will search through the teachers to find the teacher who matches the needs of the school more closely than other teachers. This is done by using Jaccard Similarity to determine which teacher has the most matching qualifications as illustrated in Figure 14

Algorithm 6 Assign Applicants to the schools

```
procedure AssignApplicants(schWithSubjectSets, applicantsAndQualifications)
    for school in schWithSubjectSets do
        for subjectSet in school do
            similarity ← (selectedApplicant ← none, jaccardSim ← 0)
            for applicant inapplicantsAndQualifications do
                simScore ← jaccardsimilarity(applicant.subjects, subjectSet)
                if simScore > similarity.jaccardSim then
                    similarity ← (selectedApplicant ← applicant, jaccardSim ← simScore)
            Add applicant to school subject set
            Remove applicant from applicants list
    return schWithSubjectSets
```

Figure 14: Assign Applicants to the positions

Two testing methods were used to test the solution. The first method used was the testing of the various classifiers in the process of predicting the subjects the applicants are qualified for and the second was the testing of the solution to compare the algorithmic assignment of applicants to the schools and the sequential assignment of applicants to the schools.

E. *Testing of algorithms*

For this portion of the testing, a number of algorithms were applied to the process of selecting subjects that an applicant has a higher aptitude for. The machine learning algorithm that was used is the One Vs. the Rest classifier and the algorithms that were used in the One Vs. the Rest classifier include Linear Discriminant Analysis (LDA), the Decision Tree Classifier, multilayer perceptron classifier (MLPClassifier), Support Vector Classification (SVC), Quadratic Discriminant Analysis, Gaussian Naive Bayes, the K-Neighbors Classifier, the Random Forest Classifier, Gaussian Process Classifier and the Ada Boost Classifier. Table 3 shows the Testing of the machine learning methods:

Table 3: Testing of the machine learning methods

Classifier	Parameters	Accuracy (/1)	Time to predict 1200 records
LDA	none	0.025	3, 570 ms
Decision Tree Classifier	max_depth =5	0.95	4, 841 ms
MLPClassifier	alpha=1	0.0375	5, 731 ms
SVC	Kernel ="linear", C=0.025	0.0375	5, 923 ms
SVC	gamma=2, C=1	0.0375	6, 224 ms
Quadratic Discriminant	none	0.0375	6, 961 ms

Analysis			
QDA	none	0.0375	7, 037 ms
GaussianNB	none	0	7, 908 ms
KNeighbors Classifier	3	0.025	17, 174 ms
Random Forest Classifier	max_depth =5, n_estimators =10, max_features =1	0.0375	37, 220 ms
Gaussian Process Classifier	1.0 * RBF(1.0)	0.0375	108, 794 ms
AdaBoost Classifier	none	0.95	128, 972 ms

To perform this testing, the algorithms were used to process 1, 200 records. The 1, 200 records were randomly generated is order to provide a variety of data for the prediction process.

From the results in Table 3, it was observed that LDA was the fastest method which took 3, 570 ms to process 1, 200 records. The slowest of the algorithms was AdaBoostClassifier which took 128, 972 ms. However, while the LDA algorithm was fast in classification, it only produced an accuracy rate of 2.5%. The AdaBoostClassifier on the other hand had an accuracy rate of 95%. The other algorithm that averaged an accuracy rate of 95% was the Decision Tree Classifier. The Decision Tree Classifier which took 4, 841 ms to process 1, 200 records. With these statistics considered, the Decision tree classifier was chosen because it had the highest classification accuracy in the shortest amount of time.

The second testing method that was used was to test the accuracy of classification between sequential assignments of students and the use of the teacher assignment algorithm to assign teachers to the schools. Table 4 shows the testing results between sequential and Algorithmic Assignment.

Table 4: Sequential vs. Algorithmic Assignment of Applicants

School	Requested Subjects	Sequential Satisfied	Algorithm Satisfied
Chamuka (Chibombo, Central)	[English]	Yes	Yes
	[History]	No	Yes
	[Civic Ed, Christian RE]	Yes	Yes
	[Christian RE]	Yes	Yes
Balaka iri Centre (Chibombo, Central)	[Lunda]	No	Yes
	[History]	Yes	Yes
	[Civic Ed]	No	Yes
Chabona basic (Chibombo, Central)	[Civic Ed, Christian RE, Geography]	No	Yes
Kapupulu basic (Luanshya, Copperbelt)	[French]	Yes	Yes
	[History, French]	No	Yes
Bwaba community (Chibombo, Central)	[English]	Yes	Yes
	[Civic Ed]	No	Yes
	[Lunda]	Yes	Yes
	[Christian RE]	Yes	Yes
	[Chitonga]	Yes	Yes
	[Geography]	Yes	Yes
	[Lunda]	Yes	Yes
Chikuse upper basic (Chibombo, Central)	[French]	Yes	Yes
	[Agric Sci]	Yes	Yes
	[English]	Yes	Yes
	[Icibemba]	Yes	Yes
	[Christian RE]	Yes	Yes
Kalebuka community (Chibombo, Central)	[Luvale]	Yes	Yes
	[English, Christian RE, French]	No	No
	[Christian RE, Geography]	No	Yes
	[Christian RE]	Yes	Yes
	[Icibemba]	Yes	Yes

F. *User Acceptance Testing*

To measure user acceptance, a questionnaire based on the technology acceptance model was given out to a number of users and responses were analysed. Figure 15 shows the descriptive statistics from the questionnaire responses.

Valid cases = 11; cases with missing value(s) = 5.

Variable	*N*	*Mean*	*Std Dev*	*Minimum*	*Maximum*
PU1	11	4.91	.30	4.00	5.00
PU2	11	4.36	.50	4.00	5.00
PU3	11	4.64	.50	4.00	5.00
PU4	11	4.73	.47	4.00	5.00
PU5	10	4.00	1.05	2.00	5.00
PEOU1	11	1.64	.67	1.00	3.00
PEOU2	11	2.09	.70	1.00	3.00
PEOU3	11	1.73	1.19	1.00	5.00
PEOU4	11	4.00	1.00	2.00	5.00
PEOU5	11	1.73	.79	1.00	3.00
PEOU6	11	4.64	.50	4.00	5.00
PEOU7	11	4.45	.52	4.00	5.00
PEOU8	11	4.55	.52	4.00	5.00
SAT1	7	4.86	.38	4.00	5.00
SAT2	11	4.45	.52	4.00	5.00
SAT3	11	4.36	.50	4.00	5.00
AU1	7	4.43	.53	4.00	5.00
AU2	11	4.45	.52	4.00	5.00
Overall	11	4.45	.52	4.00	5.00

Figure 15: Descriptive statistics from the responses

A total of 11 responses were obtained from the respondents. Table 5 shows the inferences drawn from analysing the mean results for the responses of each question where Figure 16 is the key that is used to draw the inferences.

Table 5: Inferences drawn from the mean scores of each of the responses

Construct	Question	Inference from analyzed data (Mean considered)
PU1	Using the system reduces the amount of time spent on the process of assignment	Strongly Agree
PU2	Using the system ensures that the process of assigning applicants in meritorious	Strongly Agree
PU3	Using the system makes it easier to the work	Strongly Agree
PU4	Using the system automates the process of assigning the applicants to schools	Strongly Agree
PU5	Using the system improves the jobs performance	Agree
PEOU1	I often became confused when using the system	Strongly Disagree
PEOU2	It is easy to make errors in the use of the system	Disagree
PEOU3	Interacting with the system is often frustrating	Strongly Disagree
PEOU4	I find the completion of the assignment process is relatively easy	Agree
PEOU5	I find that the system is cumbersome to use	Strongly Disagree
PEOU6	My interaction with the system is easy to understand	Strongly Agree
PEOU7	The system helps in performing goals of the assignment body	Strongly Agree
PEOU8	Overall, I find the system easy to use	Strongly Agree
SAT1	I am completely satisfied with the assignment process of the system	Strongly Agree
SAT2	I feel confident in using the applicant assignment application	Strongly Agree
SAT3	I believe that using the applicant assignment software will improve the process of selecting applicants	Strongly Agree
AU1	I found the system straightforward to use	Strongly Agree
AU2	I found it easy to collect results used in the application	Strongly Agree
Overall		Strongly Agree

0 to 1	Not answered
1+ to 2	Strongly disagree
2+ to 3	Disagree
3+ to 4	Agree
4+ to 5	Strongly agree

Figure 16

From the responses gotten, it was noted that the system was functionally useful in the process of assigning applicants to school positions but some improvement in the user interface improvements to make the application even more user friendly should be considered.

V. Discussion

From the results obtained from the testing of algorithms, it is observed that the One vs. the Rest classifier is suitable for the classification of applicant subjects because of the fact that the One vs. the Rest classifier is a multi-label classifier which is necessary in the case where applicants need multiple labels i.e. multiple subjects assigned to them. We also tested various algorithms with the One vs. the rest classifier and it was discovered that of all the algorithms that were discussed, the Decision tree classifier was the most suitable.

The other testing that was carried out i.e. the comparing of the sequential assignment of the applicants to the schools vs. the algorithm based assignment of teachers to the schools showed that the algorithm based selection of applicants was better because it was better able to assign applicants to schools based on their qualification and also based on the need of the school. This is because the algorithmic assignment of applicant took into consideration the applicant's qualifications and their similarity to the school's needs. In this case, we note that of the 27 places that were available, the sequential selection was able to successfully assign 19 places while the algorithmic selection was able to assign 26 places where even the place that was not completely assigned had an applicant whose subjects matched the school as closely as possible. This is made possible by the fact that while the sequential selection of applicants simply picks up the applicant that is next in the list, the algorithm based selection of the applicants actually loops through the applicants and compares that applicant's qualifications with the school's needs in order to determine which of the applicants qualifications closest matches the needs of the school of the school. This way, even if the algorithm based selection does not find an applicant who perfectly matches the school's needs, it will find the applicant whose qualifications at least matches the schools needs closest.

VI. Conclusion

In this paper, we have proposed the development of a solution for the assignment of applicants to the places where their qualifications would be best suited. From there, a number of algorithms namely the algorithm to process applicant qualifications, to get unique list of subjects from the school positions, to get unique list of subjects from set of school subjects and calculate their regression gradient, order the unique subjects by their regressed gradient value, order the original school subject sets by the ordered unique subject set and to assign applicants to the positions were discussed. Upon discussing the algorithms, the resting of the algorithms were then discussed. The testing phase included the testing of the classification algorithms and the comparison of sequential selection of the applicants vs. the algorithm based selection of applicants was also discussed. From this study, it has been observed that the process of assigning applicants to teaching positions is done with a few button clicks which improves upon the process of manually assigning applicants to teaching positions which would take hours or even days.

Future studies will involve additional validation of the algorithm and extending it to address other areas such as placement of medical staff in clinics and hospitals.

2 References

[1] V. P. Breşfelean, "Analysis and predictions on students' behavior using decision trees in weka environment," in *Proceedings of the International Conference on Information Technology Interfaces, ITI*, 2007, pp. 51–56.

[2] A. Osofisan, O. Adeyemo, and S. Oluwasusi, "Empirical Study of Decision Tree and Artificial Neural Network Algorithm for Mining Educational Database," *Afr. J. Comput. ICT ©*, vol. 7, no. 2, pp. 187–196, 2014.

[3] Trading Economics, *Zambia Unemployment Rate | 1986-2018 | Data | Chart | Calendar | Forecast*. .

[4] Federal Reserve Bank of San Francisco, *FEDERAL RESERVE BANK OF SAN FRANCISCO What is the relationship between inflation and GDP*. 1999.

[5] *Zambia Corruption Report*. .

[6] R. Gafarov Evgeny, A. Lazarev Aleksandr, and V. Zinovyev Aleksandr, "Algorithms for workforce assignment problem," *Proc. 2017 10th Int. Conf. Manag. Large-Scale Syst. Dev. MLSD 2017*, no. 4, pp. 1–3, 2017.

[7] T. Rihm and P. Baumann, "Improving fairness in staff assignment: An approach for lexicographic goal programming," *IEEE Int. Conf. Ind. Eng. Eng. Manag.*, vol. 2016–January, pp. 1247–1251, 2016.

[8] N. Azizi and M. Liang, "An integrated approach to worker assignment, workforce flexibility acquisition, and task rotation," *J. Oper. Res. Soc.*, vol. 64, no. 2, pp. 260–275, 2013.

[9] I. Kavakiotis, O. Tsave, A. Salifoglou, N. Maglaveras, I. Vlahavas, and I. Chouvarda, "Machine Learning and Data Mining Methods in Diabetes Research," *Comput. Struct. Biotechnol. J.*, vol. 15, pp. 104–116, 2017.

[10] R. Semaan, "Optimal sensor placement using machine learning," *Comput. Fluids*, vol. 159, pp. 167–176, 2017.

[11] S. Goldman and Y. Zhou, "Enhancing Supervised Learning with Unlabeled Data."

[12] F. Amato, A. López, E. M. Peña-Méndez, P. Vaňhara, A. Hampl, and J. Havel, "Artificial neural networks in medical diagnosis," *J. Appl. Biomed.*, vol. 11, no. 2, pp. 47–58, Jan. 2013.

[13] K. Kumartripathi, "Discrimination Prevention with Classification and Privacy Preservation in Data mining," *Procedia Comput. Sci.*, vol. 79, pp. 244–253, 2016.

[14] S. Kalmegh, "Analysis of WEKA Data Mining Algorithm REPTree , Simple Cart and RandomTree for Classification of Indian News," *Int. J. Innov. Sci. Eng. Technol.*, vol. 2, no. 2, pp. 438–446, 2015.

[15] E. C. Gonçalves *et al.*, "Prediction of carbonate rock type from NMR responses using data mining techniques," *J. Appl. Geophys.*, vol. 140, pp. 93–101, 2017.

[16] D. S. Kumar, D. Senthil, and K. S. Sukanya, "Feature Selection using Multivariate Adaptive Regression Splines," *Int. J. Res. Rev. Appl. Sci. Eng. IJRRASE*, vol. 8, no. 1, pp. 17–24, 2016.

[17] G. Tsoumakas and I. Katakis, "Multi-Label Classification," *Int. J. Data Warehous. Min.*, vol. 3, no. 3, pp. 1–13, 2007.

[18] W. Bi and J. T. Kwok, "Efficient Multi-label Classification with Many Labels."

[19] H.-F. Yu, P. Jain, P. Kar, and I. S. Dhillon, "Large-scale Multi-label Learning with Missing Labels."

[20] D. C. Montgomery, E. A. Peck, and G. G. Vining, *Introduction to Linear Regression Analysis*. John Wiley & Sons, 2012.

[21] C. M. Lynch *et al.*, "Prediction of lung cancer patient survival via supervised machine learning classification techniques," *Int. J. Med. Inf.*, vol. 108, no. August, pp. 1–8, 2017.

[22] A. Coates, H. Lee, and A. Y. Ng, "An Analysis of Single-Layer Networks in Unsupervised Feature Learning," vol. 15.

[23] A. Y. Rodríguez-González, J. F. Martínez-Trinidad, J. A. Carrasco-Ochoa, and J. Ruiz-Shulcloper, "Mining frequent patterns and association rules using similarities," *Expert Syst. Appl.*, vol. 40, no. 17, pp. 6823–6836, 2013.

[24] D. Tomar and S. Agarwal, "A survey on data mining approaches for healthcare," *Int. J. Bio-Sci. Bio-Technol.*, vol. 5, no. 5, pp. 241–266, 2013.

[25] C. H. Weng, "Revenue prediction by mining frequent itemsets with customer analysis," *Eng. Appl. Artif. Intell.*, vol. 63, pp. 85–97, 2017.

[26] S. Mabu, C. Chen, N. Lu, K. Shimada, and K. Hirasawa, "An Intrusion-Detection Model Based on Fuzzy Class-Association-Rule Mining Using Genetic Network Programming," *Appl. Rev.*, vol. 41, no. 1, 2011.

[27] L. Galárraga, C. Teflioudi, K. Hose, and F. M. Suchanek, "AMIE: Association Rule Mining under Incomplete Evidence in Ontological Knowledge Bases."

[28] E. García, C. Romero, S. Ventura, and C. De Castro, "An architecture for making recommendations to courseware authors using association rule mining and collaborative filtering."

[29] S. Lim, C. S. Tucker, and S. Kumara, "An unsupervised machine learning model for discovering latent infectious diseases using social media data," *J. Biomed. Inform.*, vol. 66, pp. 82–94, 2017.

[30] A. Nair *et al.*, "Massively Parallel Methods for Deep Reinforcement Learning," *arXiv:1507.04296*, p. 14, 2015.

[31] J. Kober, J. A. Bagnell, and J. Peters, "Reinforcement Learning in Robotics: A Survey."

[32] V. Mnih *et al.*, "Playing Atari with Deep Reinforcement Learning."

[33] H. Van Hasselt, *Reinforcement Learning in Continuous State and Action Spaces*. 2013.

[34] I. Batmaz, S. Danişoğlu, C. Yazici, and E. Kartal-Koç, "A data mining application to deposit pricing: Main determinants and prediction models," *Appl. Soft Comput. J.*, 2017.

[35] I. Dutta, S. Dutta, and B. Raahemi, "Detecting Financial Restatements Using Data Mining Techniques," *Expert Syst. Appl.*, vol. 90, pp. 374–393, 2017.

[36] F. Barboza, H. Kimura, and E. Altman, "Machine learning models and bankruptcy prediction," *Expert Syst. Appl.*, vol. 83, pp. 405–417, 2017.

[37] A. K. Pal and S. Pal, "Evaluation of Teacher â□™ s Performance : A Data Mining Approach," *Int. J. Comput. Sci. Mob. Comput.*, vol. 2, no. 12, pp. 359–369, 2013.

[38] S. Elayidom, S. M. Idikkula, and J. Alexander, "A Generalized Data mining Framework for Placement Chance Prediction Problems," *Int. J. Comput. Appl.*, vol. 31, no. 3, pp. 40–47, 2011.

[39] M. Bohanec, M. Kljajić Borštnar, and M. Robnik-Šikonja, "Explaining machine learning models in sales predictions," *Expert Syst. Appl.*, vol. 71, pp. 416–428, 2017.

[40] J. Jia, M. Iaeng, and M. Mareboyana, "Machine Learning Algorithms and Predictive Models for Undergraduate Student Retention," *Proc. World Congr. Eng. Comput. Sci. 2013*, vol. I, pp. 23–25, 2013.

[41] O. Matei, T. Rusu, A. Petrovan, and G. Mihuț, "A Data Mining System for Real Time Soil Moisture Prediction," *Procedia Eng.*, vol. 181, pp. 837–844, 2017.

[42] R. Asif, A. Merceron, S. A. Ali, and N. G. Haider, "Analyzing undergraduate students' performance using educational data mining," *Comput. Educ.*, vol. 113, pp. 177–194, 2017.

[43] S. S. Ghannadpour, A. Hezarkhani, and T. Roodpeyma, "Combination of separation methods and data mining techniques for prediction of anomalous areas in Susanvar, Central Iran," *J. Afr. Earth Sci.*, vol. 134, pp. 516–525, 2017.

[44] R. Kohavi, "A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection," 1995.

[45] S. Sivakumar and R. Selvaraj, "Adaptive Model for Campus Placement Prediction using Improved Decision Tree."

[46] W. Peng, J. Chen, and H. Zhou, "An Implementation of IDE3 Decision Tree Learning Algorithm," *Mach. Learn.*, vol. 1, pp. 1– 20, 2009.

[47] O. O. Adeyemo, T. O. Adeyeye, and D. Ogunbiyi, "Comparative Study of ID3/C4.5 Decision tree and Multilayer Perceptron Algorithms for the Prediction of Typhoid Fever," *Afr. J. Comput. ICT Afr. J. Comput. ICT Ref. Format Afr J Comp ICTs*, vol. 8, no. 1, pp. 103–112, 2015.

[48] S. Sathyadevan and R. R. Nair, "Comparative Analysis of Decision Tree Algorithms: ID3, C4.5 and Random Forest," in *Computational Intelligence in Data Mining - Volume 1*, Springer, New Delhi, 2015, pp. 549–562.

[49] A. H. Rodríguez *et al.*, "Procalcitonin (PCT) levels for ruling-out bacterial coinfection in ICU patients with influenza: A CHAID decision-tree analysis," *J. Infect.*, vol. 72, no. 2, pp. 143–151, Feb. 2016.

[50] O. Kisi, "Pan evaporation modeling using least square support vector machine, multivariate adaptive regression splines and M5 model tree," *J. Hydrol.*, vol. 528, pp. 312–320, 2015.

[51] S. Yadav, B. Bharadwaj, and S. Pal, "Data mining applications: A comparative study for predicting student's performance," *Int. J. Innov. Technol. Creat. Eng.*, vol. 1, no. 12, pp. 13–19, 2012.

[52] Saurabh Pal, "Analysis and Mining of Educational Data for Predicting the Performance of Students," *Int. J. Electron. Commun. Comput. Eng.*, vol. 4, no. 5, pp. 1560–1565, 2013.

[53] F. M. Díaz-Pérez and M. Bethencourt-Cejas, "CHAID algorithm as an appropriate analytical method for tourism market segmentation," *J. Destin. Mark. Manag.*, vol. 5, no. 3, pp. 275–282, 2016.

[54] W. Zhang and A. T. C. Goh, "Multivariate adaptive regression splines and neural network models for prediction of pile drivability," *Geosci. Front.*, vol. 7, no. 1, pp. 45–52, 2014.

[55] W. Zhang, A. T. C. Goh, Y. Zhang, Y. Chen, and Y. Xiao, "Assessment of soil liquefaction based on capacity energy concept and multivariate adaptive regression splines," *Eng. Geol.*, vol. 188, pp. 29–37, 2015.

[56] K. Haleem, A. Gan, and J. Lu, "Using multivariate adaptive regression splines (MARS) to develop crash modification factors for urban freeway interchange influence areas," *Accid. Anal. Prev.*, vol. 55, pp. 12–21, 2013.
[57] S. K. Yadav, "Mining Education Data to Predict Student â€ ™ s Retention : A comparative Study," vol. 10, no. 2, pp. 113–117, 2012.
[58] R. Timofeev, "Classification and Regression Trees (CART) Theory and Applications," 1984.
[59] X. Niuniu, "Review of decision trees," *2010 3rd Int. Conf. Comput. Sci. Inf. Technol.*, pp. 105–109, 2010.
[60] M. Abdar, M. Zomorodi-Moghadam, R. Das, and I.-H. Ting, "Performance analysis of classification algorithms on early detection of liver disease," *Expert Syst. Appl.*, vol. 67, pp. 239–251, 2017.
[61] M. Ali *et al.*, "Comparison of Artificial Neural Network and Decision Tree Algorithms used for Predicting Live Weight at Post Weaning Period From Some Biometrical Characteristics in Harnai Sheep," *Pak. J Zool*, vol. 47, no. 6, pp. 1579–1585, 2015.
[62] O. Kisi and K. S. Parmar, "Application of least square support vector machine and multivariate adaptive regression spline models in long term prediction of river water pollution," *J. Hydrol.*, vol. 534, pp. 104–112, 2016.
[63] S. Niwattanakul, J. Singthongchai, E. Naenudorn, and S. Wanapu, "Using of Jaccard Coefficient for Keywords Using of Jaccard Coefficient for Keywords Similarity," no. March, 2013.
[64] W. Goddard and S. Melville, *Research methodology: An introduction*. Juta and Company Ltd, 2004.
[65] S. Rose, N. Spinks, and I. A. Canhoto, *Management Research Applying the principles*. Routledge, 2015.
[66] R. G. Sargent, "Verification and Validation of Simulation Models," 2011.