ADCAIJ

Advances in Distributed Computing and Artificial Intelligence Journal

Vol.11 N.2

ADCAIJ.USAL.ES 2022

REGULAR ISSUE

Ediciones Universidad Salamanca

Advances in Distributed Computing and Artificial Intelligence Journal

REGULAR ISSUE

Vol. 11 N. 2

2022

Ediciones Universidad Salamanca

eISBN: 2255-2863
Volume 11, number 2
BISITE Researh Group
University of Salamanca, 2022

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal

eISSN: 2255-2863 - DOI: https://doi.org/10.14201/ADCAIJ2022112 - CDU: 004 -
IBIC: Computación e informática (U) - BIC: Computing & Information Technology (U) - BISAC: Computers / General (COM000000)

Regular Issue, Vol. 11, N. 2 (2022)

SCOPE

The Advances in Distributed Computing and Artificial Intelligence Journal (ADCAIJ) is an open access journal that publishes articles which contribute new results associated with distributed computing and artificial intelligence, and their application in different areas.

The artificial intelligence is changing our society. Its application in distributed environments, such as the Internet of Thing (IoT), electronic commerce, mobile communications, wireless devices, distributed computing, Big Data and so on, is increasing and becoming and element of high added value and economic potential in industry and research. These technologies are changing constantly as a result of the large research and technical effort being undertaken in both universities and businesses. The exchange of ideas between scientists and technicians from both academic and business areas is essential to facilitate the development of systems that meet the demands of today's society.

This issue will be focused on the importance of knowledge in advanced digital technologies and their involvement in the different activities of the public-private sector related to specialization in digital technologies and blockchain. The issue also includes articles focusing on research into new technologies and an in-depth look at advanced digital tools from a practical point of view in order to be able to implement them in organisations in peripheral and border areas.

We would like to thank all the contributing authors f or their hard and highly valuable work and members of 0631_DIGITEC_3_E project (Smart growth through the specialization of the cross-border business fabric in advanced digital technologies and blockchain) supported by the European Regional Development Fund (ERDF) through the Interreg Spain-Portugal V-A Program (POCTEP) . Their work has helped to contribute to the success of this issue. Finally, the Editors wish to thank Scientific Committee of Advances in Distributed Computing and Artificial Intelligence Journal for the collaboration of this issue, that notably contributes to improve the quality of the journal.

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal

eISSN: 2255-2863 - DOI: https://doi.org/10.14201/ADCAIJ2022112 - CDU: 004 -
IBIC: Computación e informática (U) - BIC: Computing & Information Technology (U) - BISAC: Computers / General (COM000000)

Regular Issue, Vol. 11, N. 2 (2022)

INDEX

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal
Regular Issue, Vol. 11 N. 2 (2022), 129-146
eISSN: 2255-2863
DOI: https://doi.org/10.14201/adcaij.26764

An Ensemble Classification and Regression Neural Network for Evaluating Role-based Tasks Associated with Organizational Unit

Ahmed Alrashedi[a] and Maysam Abbod[b]

[a] College of Business, Human Resources Department, University of Jeddah, Saudi Arabia
[b] Department of Electronic and Computer Engineering, Brunel University London
aalrashedi2010@gmail.com, Maysam.Abbod@brunel.ac.uk

KEYWORDS

neural network; genetic algorithm; decision trees; non-linear model

ABSTRACT

In this paper, we have looked at how easy it is for users in an organisation to be given different roles, as well as how important it is to make sure that the tasks are done well using predictive analytical tools. As a result, ensemble of classification and regression tree link Neural Network was adopted for evaluating the effectiveness of role-based tasks associated with organization unit. A Human Resource Manangement System was design and developed to obtain comprehensive information about their employees' performance levels, as well as to ascertain their capabilities, skills, and the tasks they perform and how they perform them. Datasets were drawn from evaluation of the system and used for machine learning evaluation. Linear regression models, decision trees, and Genetic Algorithm have proven to be good at prediction in all cases. In this way, the research findings highlight the need of ensuring that users tasks are done in a timely way, as well as enhancing an organization's ability to assign individual duties.

1. Introduction

The administration of an organization's human resources, such as recruitment, employment, deployment, and retention of workers, is known as human resource management (HRM) (Boxall, 2013). It is frequently referred to as «the act of considering humans as resources and assets». This typically covers all the management duties and policies toward dealing with individuals in an organization. It also concerns the working relationships of an organization as well as the policies and tasks associated

with individual effort (Lim et al., 2017). There's no denying that employees are a valuable asset to any organisation, and the goal is to make efficient use of them in order to reduce risk and maximise return on investment. To do so, it's critical to keep accurate records and to use a straightforward management system appropriate for the situation in question (Saiz-Rubio and Rovira-Más, 2020). There are numerous considerations to be taken into account when developing an automated system for managing people at work in order to achieve the organization's mission objectives (Sony and Naik, 2020). In fact, it is necessary to develop a comprehensive set of parameters that encompasses all aspects of organizational strengthening in order to achieve success (Margherita et al., 2021). These should include everything from the people's culture to their working conditions and responsibilities. In order to achieve the company's objectives, a human resource system could be assumed by an automated-aided system that will handle that task. This system will be able to use automated utilization of resources to effectively hire employees with the necessary skills and assist with training and development of current employees (Wheeler and Buckley, 2021).

In light of the large number of human resource management systems currently available and the widespread adoption of these systems by many organizations, the majority of studies concentrate on the design and development of new human resource management systems to deal with an increase in organizational commitment. Artificial intelligence applications in human resource management have also been shown to have a wide range and impact in prior studies (Qamar et al., 2021) In the future, artificial intelligence (AI) in HR management is projected to have a positive impact on performance evaluation and other organisational processes.The work of Jin and Wang (2020), who use the multi-mode fuzzy logic control algorithm to evaluate the comprehensive level of employees' competence by establishing the degree of membership of work ability, has highlighted many aspects of the operations of these systems. This work is critical to these because it demonstrates the effectiveness of the multi-mode fuzzy logic control algorithm. Motivated by the study of Laudon and Laudon (2015) in articulating a general consensus about specialized functional assignments that the HRMS deals with, there is a need to evaluate how it is used and invested to facilitate human resource management in an administrative capacity. The research also revealed that a system dedicated to managing human resources needs to be connected to functional information subsystems, especially at the strategic level. On the other hand, Maier et al. (2013) reveals that with HRMS, managers get time to conduct essential strategic tasks by shortening wait times for responsibilities such as data entry. Although it is also indicated that when these users are unhappy with the system, it is possible that it will fail because they will not provide the information that is needed when it is needed (Melville et al., 2010). This articulating that coordination of the actions of the various elements is a critical task in HRMS (Schultze and Leidner, 2002).

It was long before computers were introduced that control information systems were implemented. According to Jääskeläinen et al. (2020), it first appeared in the field of account administration and other administrative areas. Considering the dynamic and progressive organizations have digital desires and are interested in using HRMS. Many organizations show how businesses should leverage their implementation of HRMS in order to better meet operational requirements while also considering the longer-term effect on business operations by increasing the amount of technical and digital activity delivered (Jawabreh et al., 2020). That is why this study designed and developed HRMS in order to understand and interpret the outcomes of its activities. The output or performance measurements must be evaluated and also to discover their capabilities, skills, and task assignments and how they carry them out. Hence, a machine learning algorithm was proposed and applied. This is justified by the fact that nowadays; human resource management software includes more than just undertaking a single

task. It also includes recruiting and record-keeping, training, and performance appraisal, all of which have helped to shift HRM from a task-oriented to a people-oriented perspective (Bilgic, 2020).

1.1. Overview of the Development of HRSM

The results of this study have resulted in the development of a programme for human resource management (HRM) that evaluates employee performance. In order to accomplish this, the programme intends to take advantage of the most up-to-date technologies and artificial intelligence. Employee information, work performed and how it was performed, previous and current evaluations from previous and current supervisors, available training programmes, and employee complaints can all be obtained from the website by the manager. This is intended to be as accurate as possible, similar to the study performed in Ullah et al. (2021). When combined, this information provides a clear and accurate picture of the employee, as well as guidance on how to develop and benefit from them to the greatest extent possible. Managers can quickly and easily navigate between website pages in order to gather information in a more organised fashion. Once the programme has been run through once, the manager will have a comprehensive understanding of their employee's history, from the time of his or her hire to the present day and everything in between. Thus, the Internet's progress has a substantial impact on the employment of AI in human resources. As a result of the widespread usage of AI for HR management in the public sector, synchronous implementation might be widely deployed (Abdeldayem and Aldulaimi, 2020). AI in HR management has the ability to develop into hitherto unimagined schemes, as this claim suggests. Causation, randomness, and process formalisation have all been found to be economically efficient and socially acceptable for AI-supported practises in human resources management (Tambe et al., 2019). Artificial intelligence and human resource functions have a positive association, based on an examination of the linkage, between innovativeness and the ease with which artificial intelligence can be applied in human resource management operations. As a result, one of the factors for determining the AI tool's effectiveness and efficiency is based on the AI tool's concept and user friendliness (Bhardwaj et al., 2020). Artificial intelligence has also been shown to be cost-effective and time-saving in HR operations. Managing tasks, coordination, and control now takes a fraction of the time it used to. As a result, in this instance, time is of the essence. Artificial intelligence must be trusted as a decision-making tool by businesses (Kolbjrnsrud et al., 2016).

Through the use of this programme, Human Resources Management will benefit from information technology, which will allow it to record information, save information, and retrieve information in an accurate, timely, and secure manner, and increase its ability to communicate, either with its employees or with other Society companies, as well as assist it in the process of making the best decision possible It is also anticipated that the artificial intelligence that has been used in this programme will be beneficial, because artificial intelligence has the ability to simulate human intelligence by machines, and it uses the available databases to connect and analyse it, then infer correct solutions with accuracy and speed that exceeds human capacity.

This programme will enable the Human Resources Department to obtain comprehensive information about their employees' performance levels, as well as determine their capabilities, skills, and tasks that they perform and how they accomplish them, through the use of technology. The system has provision for providing individual to record a voluntary task (see Figure 1).

In addition, the administration will be fully aware of their previous experiences, the level of their attendance, the training that they have received and how it has affected their performance, as well as the complaints that have been lodged against them and the validity of those complaints. Obtaining this

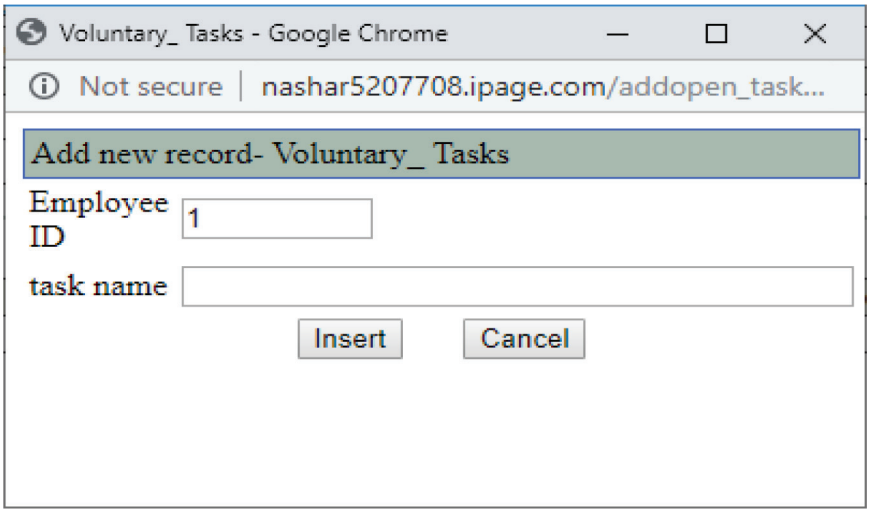

Figure 1. The provision to Record Voluntary Tasks

information would allow the human resources department to develop a comprehensive picture of the employee's performance levels and to implement appropriate training programmes to enhance the employee's distinguishing characteristics while also addressing the employee's weaknesses. HRM would be able to place each employee in the appropriate position as a result, allowing the company to reap the maximum benefit from each and every employee.

Additionally, this programme is regarded as a tool to help the employee understand his or her responsibilities and duties, which will serve as the basis for his or her evaluation. He would be able to perform his duties effectively if he was familiar with the work procedures. It would also examine its previous and current evaluations in order to determine its realistic level from the administration's perspective, with the option of objecting to and discussing the results of the evaluations in question. He would be more satisfied with the organizational decisions if he could resolve this issue. Aside from that, the employee would be able to recommend specific training programmes that would assist them in increasing their efficiency and productivity. It should be clear from the foregoing that this programme has numerous benefits that accrue to HRM, direct managers, senior managers, and employees as a whole, and that these benefits are returned to the organisation in the form of benefits that enable it to achieve its goals with less time, less effort, less cost, and higher production.

1.2. Overview of Ensemble Classification and Regression Neural Network

A combination of predictive analytics and Neural Networks was used in this study (NN) and that is why is describe as an Ensemble. Instead of using predictors to obtain insights into data and its structure, neural networks (NN) strive to efficiently reflect the underlying qualities of the data in terms of accuracy and development over time, in addition to delivering good predictions outcomes (Karlaftis and Vlahogianni, 2011). Given the fact that predictive analytics can also explain phenomena under investigation through interpretations, NN applications do not target interpretation but rather marginal effects and signs that are more flexible than other predictive analytics because

a functional form can be approached through learning rather than being assumed a priori, as is the case with some predictive analytics.

To improve prediction, ensembles integrate numerous claims or apply multiple learning methods, where the best of them can be produced utilising the best of any of the component learning algorithms (Usman et al., 2020). Unlike traditional models, alternative models are available in machine learning groups in a limited number, but they often provide a considerably more flexible structure for such models (Gong et al., 2020). Ensemble predictions are typically more computationally intensive than single model predictions, and this is true in the majority of circumstances. In some ways, ensemble learning can be viewed as a means to compensate for inefficient learning algorithms by performing a significant amount of additional computing (Duan, et al., 2007). On the other hand, the alternative is to get a great deal more knowledge about a single non-ensemble system. An ensemble system can be more efficient in terms of total accuracy improvement by spreading the same increase in computing, storage, or communication resources among two or more methods, rather than increasing the resource usage for a single method (Folino et al., 2021).

In this study, bagging is the type of Ensemble method used; but other Ensemble techniques such as boosting and stacking of the ensemble methods can be used (Dou et al., 2020). These techniques are now widely implemented and have been extensively investigated in several areas of studies. Bagging dwells on aggregating various machine learning techniques usually on classification and regression technique (Breiman, 1996; Bühlmann, 2012; Yariyan, et al., 2020). It means placing equal weight in the ensemble bag on each model. Bagging is possible to train ensemble models by using a randomly selected subset of the training sets, which increases the variance of the ensemble models. In addition, new versions of bagging that include online bagging are currently adopted (Seni and Elder, 2010). Neural bagging network for the prediction of Urban traffic flow has been proposed in Moretti et al. (2015). Furthermore, this study combining four different approach that includes neural networks to forecast regression across the entire hierarchical ensemble predictors for the purpose of classification and regression neural network of evaluating role-based tasks associated with the organizational unit. The machine learning classification and regression predictor were used to determine which phase the individual tasks should be structured in order to produce more accurate results, and which phase the organisational individual tasks should not be structured.

2. Methodology

This study extends from the design and development of HRMS in Ahmed et al. (2021) and conceptualized linear regression models, decision trees, and a Genetic Algorithm in order to simulate the performance of the role-based tasks associated with organizational unit. This is similar to the approach of Vu et al. (2020) who proposed an agent-based simulation modelling software architecture for building social mechanisms. It is also toward Trejos, et al. (2016) implementation technique where the use of a genetic algorithm was applied minimize the least squares criterion when dealing with multiple linear regression's problem of variable selection. Hence, the current study will build a model in order to select the best one, by determine the validity of each attributes.

A number of considerations were taken into account when choosing these four algorithms, the first of which being the availability of computer resources. Artificial Neural Networks are more complex in terms of computation than traditional algorithms due to the additional complexity and time necessary for Decision Tree training, while Genetic Algorithm provides higher computational capability for

linear regression. When compared to other machine learning techniques, Linear Regression has much less processing capacity.

The «data» that is currently available is another rationale for using the four algorithms. In order to use Artificial Neural Networks in their applications, data must be normalised and scaled. Missing values in the data gathering might also lead to low accuracy. The employment of genetic algorithms, on the other hand, necessitates the normalisation and scaling of data. Missing values in the data gathering might also lead to low accuracy. When using a Decision Tree to analyse a situational analysis problem, no data normalisation or scaling is required. Another thing to keep in mind is that missing values in the data have no bearing on the decision tree development process. Linear regression is a statistical approach for determining the nature of the connection between variables in linearly separable datasets. It's usually utilised to figure out what kind of relationship exists between variables.

In addition to the first two justifications for using these four algorithms, the final justification is based on algorithm operations, where Artificial Neural Networks (ANNs) operations are inspired by biological processes, providing yet another justification for using the four algorithms in this study. It was created to imitate how a human brain analyses and processes information, to be more exact. The usage of genetic algorithms, on the other hand, is employed to address both limited and unconstrained optimization issues. Genetic algorithms function by using a natural selection process to simulate the process of biological evolution. The Decision Tree is a data mining method that is often used for, among other things, developing classification systems based on many covariates or developing prediction algorithms for a target variable. If you want to model the relationship between two variables, you can use linear regression. It works by fitting the data you've gathered into a linear equation. When two variables are compared, one is called an explanatory variable and the other is called a dependent variable, with the explanatory variable being the more important of the two.

2.1. Linear Regression

The ability to highlight the important or contribution of a variable among combination of variables dwells on the nature of the dataset under investigation. Regression analysis is one of the most important analysis that can determining y prediction how important or the level of contribution a variable can make (Spooner, et al., 2020) The ordinary least squares estimator is important in linear regression, and it may sometimes seem like there are no other estimators that are reasonable and applicable. While there are other options, most of them are good for specific scenarios (Montgomery, et al., 2021). Linear regression is a type of predictive analysis that is both simple and widely used (LR). In the context of the overall concept of regression, two aspects are discussed: (1) Does a collection of predictor variables do a good job of predicting an outcome variable? (2) Does a collection of predictor variables predict an outcome variable? Second, which variables are significant predictors of the outcome variable and in what ways do they influence the outcome variable are discussed. Rationally projected relationships between one dependent variable and one or more independent variables can be used to explain the relationship between two or more independent variables.

The naming of variables. There are several terms that can be used to describe the dependent variable in a regression. It is possible to name an outcome variable, a criteria variable, an endogenous variable, or a regression. There are several names for independent variables in regression analysis, including exogenous variables, predictor variables, and repressors. Determining the strength of predictors, predicting an outcome, and trend forecasting are three of the most common applications for regression analysis.

2.2. Genetic Algorithm

In computer science, a genetic algorithm (GA) is a search-based optimization methodology that is based on the principles of Genetics and Natural Selection. Also, it is employed in the search for ideal or near-optimal solutions to difficult problems that would otherwise take a lifetime to resolve by other means. It is also employed in the optimization of problems, as well as in science and machine learning applications. Throughout the history of mankind, nature has served as a great source of inspiration. GAs are search-based algorithms that are based on the concepts of natural selection and genetics, respectively. GA is a subset of a much broader computing branch known as Evolutionary Computation, which is a subset of GA. In computer science, there are a wide range of issues to consider, including NP-Hardness. What this essentially means is that even the most efficient computer systems take an extremely long time (sometimes years!) to solve the problem. GAs proved to be an effective tool in this situation, delivering functional near-optimal solutions in a relatively short period of time. It is possible that there is a pool or a population of potential alternatives to the problem under consideration in GAs. Recombination and mutation of these solutions result in the creation of new offspring. The process is repeated over successive generations of children. Each individual has a fitness value assigned to them, and the fitter individuals have a greater chance of mating and producing more fit individuals than the less fit individuals. Despite the fact that genetic algorithms are sufficiently randomized in nature, they perform significantly better than random local search because historical knowledge is frequently utilized.

2.3. Decision Trees

A decision tree is a useful algorithm for machine learning tasks that can be used for both regression and classification tasks. A decision tree is so named because the algorithm breaks down a large dataset into smaller and smaller pieces until the data is broken down into single instances that are then categorized, thus earning the term «decision tree». Imagine a tree with several leaves, which would represent the way the groups are divided if you were to imagine the algorithm's results. A decision tree has a lot in common with a flowchart. When using a flowchart, you begin at the beginning point, or root, of the chart and then move on to one of the next possible nodes based on how you respond to the filtering criteria of the starting node. If the process reaches a conclusion, it is repeated. Every internal node in a decision tree is subjected to some form of testing or filtering criteria, and all decision trees function in the same way. Those on the outside, known as «leaves», are the nodes that connect the tree's endpoints to the data point in question and serve as labels for that data point. The branches that lead from the internal nodes to the next node are made up of features or combinations of characteristics. Those paths that lead from the root to the leaves serve as the rules by which the data points are defined.

2.4. Neural Networks

Neural networks are mathematical models that store knowledge by utilising learning algorithms that are inspired by the human brain. Given that neural networks are used in computers, they are collectively referred to as a 'artificial neural network,' which stands for artificial neural network. Machine learning is a term that is frequently heard in this field these days, and it refers to the scientific discipline that is concerned with the design and development of algorithms that allow computers, such as sensor data or databases, to learn on the basis of data. In machine-learning research, one of the primary goals is for computers to learn to recognise complex patterns automatically and to make intelligent

data-based decisions based on that information. Aside from statistics and data processing, machine learning is also closely associated with fields such as pattern recognition, artificial intelligence, and pattern recognition. Despite the fact that neural networks are a popular machine learning platform, there are numerous other machine learning techniques, such as logistic regression and support for vector machines.

Deep learning algorithms are currently being used, and they provide better accuracy than standard neural networks. However, while they may appear to be black boxes on the surface, they are actually striving to do the same thing as every other model on the inside in order to make accurate predictions.

2.5. The Reason for Adopting the Four Types of the Algorithms

The fact that «an increasing number of scientific methods are turning to predictive analytics and machine learning to support and construct predictive models that will speed up discovery» (Gupta et al., 2021) is one of the reasons why this study used four different types of algorithms. In a similar vein, predictive analytics «incorporates and conducts prediction with the highest likelihood of success and the lowest amount of mistake feasible» (Zhang et al., 2020) Furthermore, the purpose of research incorporating predictive models into human resource management is to increase the dependability as well as the speed with which decisions are made. It was for this reason that predictive analytical tools, such as data modelling and machine learning, were employed in this work. Precision analytics has the ability to be applied to any unknown event in order to produce predictions about what will happen next. As shown in Table 1, a comparison of the four algorithms used to determine their performance is presented, which includes a discussion of the merits and disadvantages of each technique, as well as the rationale for selecting them to be used in the development of the performance appraisal system.

2.6. Datasets

The datasets for this study are generated from the evaluation instrument build for the research (see Table 2). These instruments comprise of Performance appraisal which is one of the core dimensions in this study, with 11 items, followed by «Performance and Software Benefit» with 6 items, «Performance Aims/Objectives» with 5 items, «Job Description» with 13 items, «Clarity» with 8 items, «Importance» with 5 items, «Utilization of Artificial Intelligence» with 11 items, and «Technology Adoption Factors» with 8 items.

The statistical method has demonstrated that all of these items are significantly correlated with one another, and the validation test on their relationships has demonstrated that they are highly significant. Using «Linear regression», it was determined that Technology (Artificial Intelligence) Adoption influences Use of artificial intelligence by 62 percent, whereas the Use of artificial intelligence influences «Performance software benefit of Performance Appraisal by 41 percent, and 74 percent to the «Performance aims/obligations» by Technology (Artificial Intelligence) Adoption. However, «Clarity of job description» has a positive impact on «Performance of software benefit of Performance Appraisal 51 percent 69 percent by »Performance aims/objectives of Performance Appraisal. «It has a positive impact on «Performance of software benefit of Performance Appraisal» and a negative impact on «Performance aims/objectives of Performance Appraisal». Genetic Algorithms, Decision Trees, and Neural networks are used in conjunction with the statistical approach due to the success of the statistical approach.

Table 1. Research Data Collection Instruments

Artificial Neural Network (ANN)	Genetic Algorithm (GA)	Decision Tree (DT)	Linear Regression (LR)
ANNs are more complex in computing terms than traditional algorithms.	A GA has a higher computational power than linear regression.	DT training is relatively expensive as it is more complex and takes more time.	LR has considerably lower computational power when compared to some of the other machine learning algorithms.
ANNs require normalization and scaling of data. Also, missing values in the data cause low accuracy.	GAs require normalization and scaling of data. Also, missing values in the data cause low accuracy.	A DT does not require normalization or scaling of data. Missing values in the data also do NOT affect the process of building a DT to any considerable extent.	LR fits linearly separable datasets almost perfectly and is often used to find the nature of the relationship between variables.
ANNs are biologically inspired computational networks. ANN are designed to simulate the way the human brain analyses and processes information.	GAs solve both constrained and unconstrained optimization problems based on a natural selection process that mimics biological evolution.	A DT is a commonly used data mining method for establishing classification systems based on multiple covariates or for developing prediction algorithms for a target variable.	LR attempts to model the relationship between two variables by fitting a linear equation to observed data. One variable is considered to be an explanatory variable, and the other is considered to be a dependent variable.

2.7. Performance Evaluation

Performance appraisal is one of the core dimensions in this study, with 11 items, followed by «Performance and Software Benefit» with 6 items, «Performance Aims/Objectives» with 5 items, «Job Description» with 13 items, «Clarity» with 8 items, «Importance» with 5 items, «Utilization of Artificial Intelligence» with 11 items, and «Technology Adoption Factors» with 8 items. The statistical method has demonstrated that all of these items are significantly correlated with one another, and the validation test on their relationships has demonstrated that they are highly significant. A technique known as «Linear regression» has been employed to determine the strength of their impact on predicting the influence of each variable, with the goal of determining the influence of Technology (Artificial Intelligence) Adoption. While the use of artificial intelligence increases the benefit of performance appraisals by 62 percent, it has a negative impact on the «Performance software benefit» of Performance Appraisal by 41 percent and 74 percent on the «Performance aims/objectives», while it increases the «clarity of job description» and the «importance of job description. «However, «Clarity of job description» has a positive impact on «Performance of software benefit of Performance Appraisal 51 percent 69 percent by «Performance aims/objectives of Performance Appraisal». It has a positive impact on «Performance of software benefit of Performance Appraisal» and a negative impact on «Performance

Table 2. Research Data Collection Instruments

Var no.	Name	min	max	average	variance
1	Knowledge of working and procedures	55.0	95.0	73.3	151.9
2	The ability to determine the working requirements	4.0	94.0	70.8	330.9
3	Knowledge of regulations and technical concepts related to work	51.0	91.0	73.9	138.8
4	The ability to determine the working procedures and timetable	50.0	94.0	74.0	98.0
5	Achieving the required task at the right time	59.0	91.0	76.3	59.3
6	Implementation quality and skills followed	70.0	90.0	77.1	43.3
7	The ability of audit and review	70.0	90.0	77.2	52.3
8	Capacity to develop	58.0	88.0	74.3	59.1
9	Optimal utilization of working hours	59.0	88.0	77.1	55.7
10	Ability to overcome the difficulties	50.0	89.0	72.5	132.7
11	Keeping up to date on new issues	50.0	89.0	74.9	97.2
12	Ability to communicate with others effectively	50.0	88.0	74.2	62.7
13	Effective participation in meetings	70.0	88.0	74.7	28.6
14	Initiative and able to provide alternative solutions in different tasks	70.0	93.0	76.7	62.6
15	Ability to train others and transfer the knowledge	70.0	88.0	74.2	29.2
16	The ability of discussion and expressing the opinion	70.0	89.0	73.8	27.7
17	The ability to estimate the risk	55.0	88.0	73.2	74.9
18	Addressing the growing challenges transparently	58.0	90.0	72.9	107.0
19	Well-behaved	60.0	90.0	76.4	69.9
20	Dependable	55.0	82.0	71.3	35.8
21	Accepting the instructions and willing to take action	53.0	90.0	74.1	99.0
22	Working skills in a team	50.0	87.0	72.3	60.7
23	Preservation of work property	51.0	80.0	72.6	57.9
24	Keeping work official secrets	50.0	80.0	70.8	56.0
25	Good-looking	55.0	80.0	70.8	51.4
26	Relationship with managers	59.0	91.0	76.2	46.5
27	Relationship with colleagues	59.0	94.0	76.3	46.7
28	Relationship with clients	53.0	89.0	73.1	93.0
output	Overall	65.9	83.8	74.1	32.3

aims/objectives of Performance Appraisal». Given the success of the statistical approach, the Genetic Algorithm, Decision Tree, and Neural Network are all used in conjunction with it.

In order to model the performance of the employees, a variety of algorithms have been used, beginning with a linear model such as regression, decision trees, and a Genetic Algorithm optimised linear model, and progressing to a non-linear model such as a neural network. It is necessary to conduct a

comparison study in order to determine the validity of each betting model before it can be used in the system to select the best one.

3. Simulation Analysis

MATLAB was used for all analyses. The train and testing sets are stored separately. The regression model was developed using MATLAB's regress function with 28 inputs and the out-put being employee performance. Other algorithms are required to look for non-linear parameters when using a linear regression model. GA is used to search for the model parameter which is the same structure as the model. To offset and bias in the data, a constant term is added. the search parameters are constricted to [0,3] The GA toolbox's default setting is «Population size: 200», «Coding: real number», «Selection: Stochastic», «Number of iterations: 1000», «Cross over probability: 80 percent» and «Mutation probability: 5 percent». The same training data is used to generate the decision tree. FitCTree is used to generate the tree. Neural networks are developed using training and testing data. models are developed using MATLAB neural network toolboxes (fitnet). To find the best network structure, various topologies were tested. Single hidden layer with 10 neurons, sigmoid activation function for the hidden layer, and linear activation functions for the output layer give the best performance.

4. Presentation of the Results and Discussion

The GA linear equation fitting is shown in Figure 2 using the training data, while the testing is show using the fitted equation obtained by the GA algorithm as shown in Figure 3. The training and testing model RMS are 0.0363, and 0.2111 respectively. It is noted that the training and testing errors are more balanced in comparison to the linear regression technique. Table 3 lists the variables and the associated parameters.

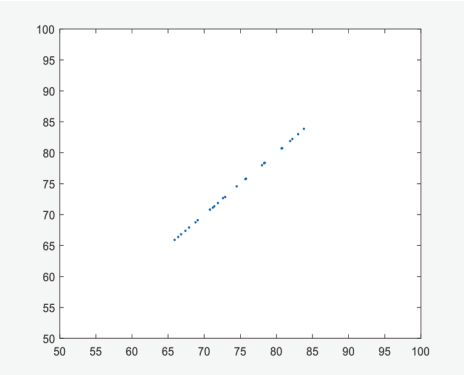

Figure 2. Genetic algorithm linear equation using training data

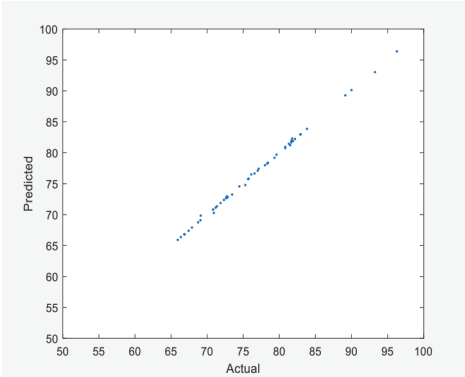

Figure 3. Genetic algorithm linear equation using testing data

Table 3. Genetic algorithm linear equation parameters fitting

Var no.	Name	Para.
1	Knowledge of working and procedures	0.0380
2	The ability to determine the working requirements	0.0365
3	Knowledge of regulations and technical concepts related to work	0.0220
4	The ability to determine the working procedures and timetable	0.0609
5	Achieving the required task at the right time	0.0247
6	Implementation quality and skills followed.	0.0533
7	The ability of audit and review	0.0142
8	Capacity to develop	0.0303
9	Optimal utilization of working hours	0.0523
10	Ability to overcome the difficulties	0.0314
11	Keeping up to date on new issues	0.0325
12	Ability to communicate with others effectively	0.0216
13	Effective participation in meetings	0.0617
14	Initiative and able to provide alternative solutions in different tasks	0.0357
15	Ability to train others and transfer the knowledge	0.0114
16	The ability of discussion and expressing the opinion	0.0462
17	The ability to estimate the risk	0.0109
18	Addressing the growing challenges transparently	0.0398
19	Well-behaved	0.0087
20	Dependable	0.0469
21	Accepting the instructions and willing to take action	0.0715
22	Working skills in a team	0.0173
23	Preservation of work property	0.0290
24	Keeping work official secrets	0.0630
25	Good-looking	0.0313
26	Relationship with managers	0.0231
27	Relationship with colleagues	0.0454
28	Relationship with clients	0.0416
	offset	0.0363
RMS: Training: 0.0363, testing: 0.2111		

The finding associated to decision trees reveals that the tree was created and displayed in the manner depicted in Figure 4. One of the disadvantages of using a decision tree is that it is dependent on one of the variables in order to begin at the root of the tree. If such information is not available for a specific employee, or if the employee's score is low, the final results will be influenced by this. Figure 5 depicts the relative importance of each independent variable in the decision-making procedure. It is

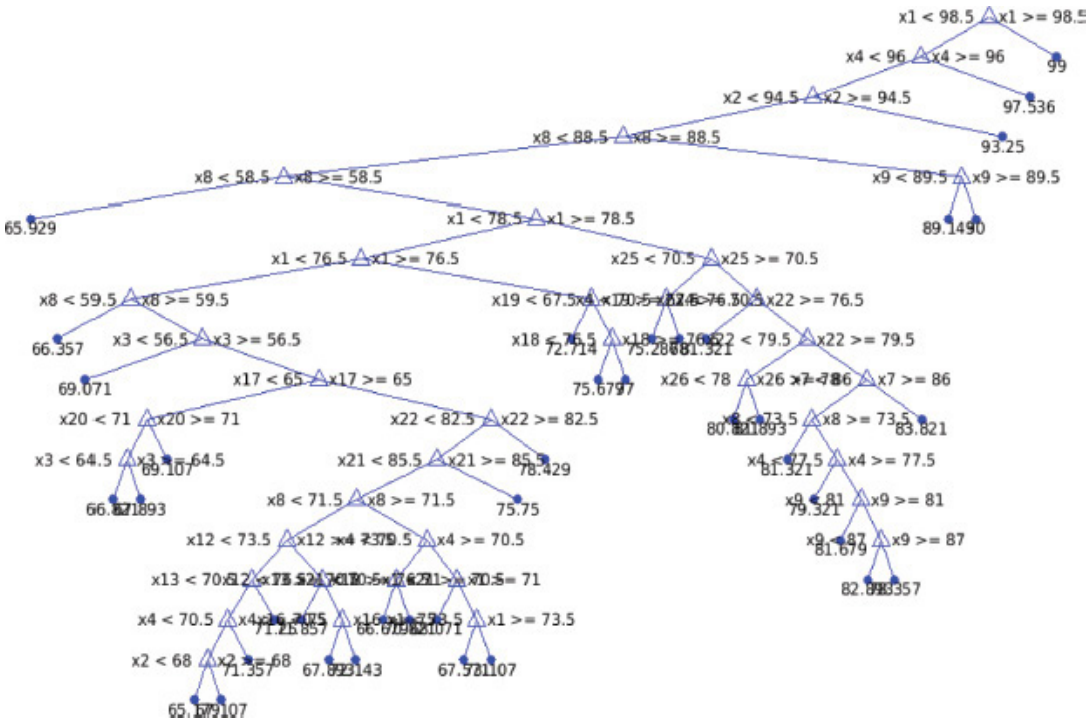

Figure 4. Decision tree given by the training data

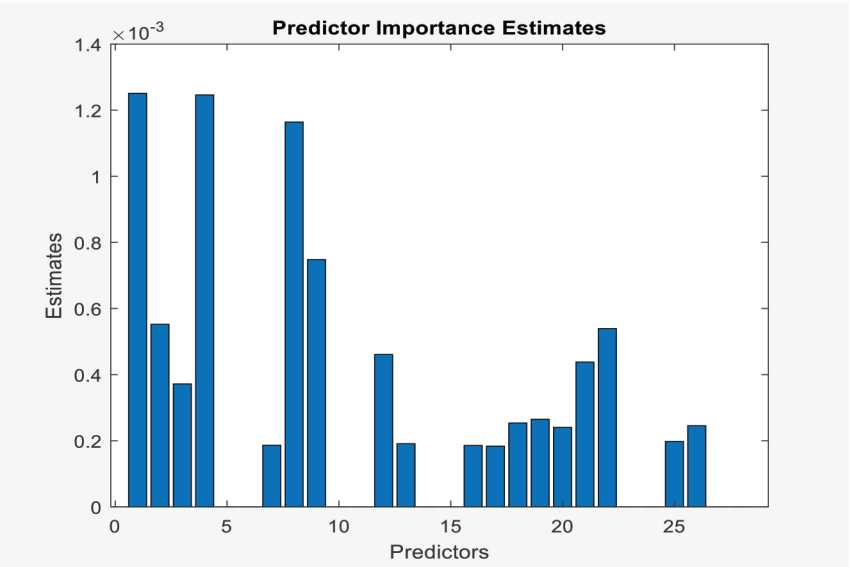

Figure 5. Decision tree Variables Importance

demonstrated that the knowledge of the job and the ability to determine the work procedure are the most important factors that differ from the linear model of performance.

The finding associated with Neural Network based on the fact that the network was trained using the default settings, the training performance of the network which took 784 epochs to reach the best fitting. Figure 6 present the model fitting.

The model fitting (predicted against actual) for the training, validation, and all data) indicated that the training and testing model RMS are 9.8114E-4, and 0.0970 respectively while the coefficient of R^2 for the training is perfect at 1. This is much better accuracy in comparison to other models. In particular, the testing accuracy is the least in comparison the GA linear equation. In order to evaluate the advantages and disadvantages of the various algorithms that have been used to create an artificial intelligence model for evaluating the performance of employees, a comparison table has been created. The key characteristics of each algorithm, as well as its accuracy, are depicted in Table 4. In terms of computing, ANNs are also more sophisticated than standard methods. Data normalisation and scaling are required by ANN. Low accuracy is sometimes caused by missing values in the data. Biologically inspired computational networks are referred to as ANNs. An artificial neural network (ANN) was created to mimic how the human brain analyses and processes information.

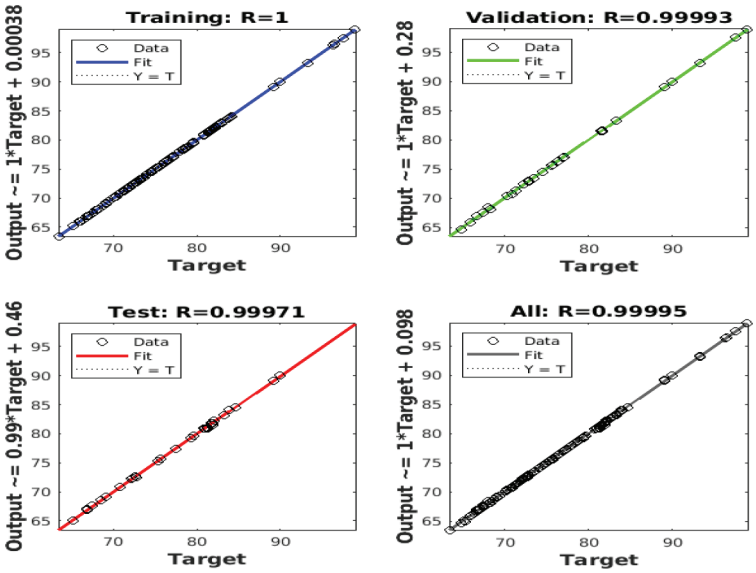

Figure 6. Neural Network Model Predictions

Table 4. Comparison of the four algorithms

Artificial Neural Network (ANN)	Genetic Algorithm (GA)	Decision Tree (DT)	Linear Regression (LR)
Train RMS = 9.8114E-4 **Test RMS = 0.0970**	Train RMS = 0.0363 Test RMS = 0.2111	Train RMS = 4.1282e-04	Train RMS = 2.8285E-14 Test RMS = 0.3211

GA outperforms linear regression in terms of processing power. GA necessitates data standardisation and scaling. Low accuracy is sometimes caused by missing values in the data. GA uses a natural selection approach that replicates biological evolution to tackle both confined and unconstrained optimization problems. Because of the complexity and time required, DT training is relatively costly. Data normalisation and scaling are not required while using DT. In addition, missing values in the data have no significant impact on the decision tree-building process. DT is a popular data mining method for creating classification systems with various variables or generating prediction algorithms for a target variable. When compared to other machine learning techniques, LR has a significantly lesser processing power. LR almost perfectly fits linearly separable datasets and is frequently used to determine the nature of the relationship between variables. By fitting a linear equation to observed data, LR seeks to model the relationship between two variables. One variable is regarded as an explanatory variable, while the other is regarded as a dependent variable.

Due to the fact that the nonlinear modelling technique and other advantages not accessible in other algorithms, the artificial neural network was found to have the best outcomes out of the four generated models used in the performance rating software.

5. Conclusion

This paper identifies that dedicated HRMS, specifically HRMS used for human resource-related functions, govern day-to-day HR management operations. There will typically be an entire department's tasks across the board in each company which must be flawless. That is why the study examined machine learning approach toward evaluating role-based tasks associated with organizational unit. To determine which algorithms produce better results on the programme, this paper has examined the algorithms (Genetic Algorithm, Decision Tree, and Linear Regression, as well as Neural Network) that have been tested on the programme. The ease with which individual roles can be assigned to users within an organisation, as well as the importance of ensuring that the tasks are carried out effectively, have been discussed throughout this paper. A Neural Network composed of classification and regression tree links was used to evaluate the effectiveness of role-based tasks associated with an organization unit as a result of the findings of this study. In order to obtain comprehensive information about their employees' performance levels, as well as their capabilities, skills, and the tasks they perform and how they perform them, a Human Resource Management System (HRMS) was designed and developed for them. In order to evaluate machine learning, data sets were extracted from the system's evaluation and used for machine learning evaluation. The neural network algorithms have been selected because they are the most effective and advantageous for the program's needs. It was revealed that the neural network yielded the greatest results out of the four developed models that were employed in the performance rating software, this was owing to the nonlinear modelling technique used, as well as additional advantages that were not available when using other techniques. Results of the study can be used to streamline the assignment of specific roles to users within an organisation and to ensure that duties are carried out efficiently.

In contrast to the current study, which used Ensembles integrating four learning algorithms to build a better of them and found the best, a future study should use the best of any of the constituent learning algorithms individually to find the best results. In the current study, it was demonstrated that while Bootstrap aggregation was used because of its consistent allocation of equal weight, future studies should make use of Boosting because it involves incrementally building the training instances and

also because it is a technique in which a model selection algorithm is used to select the best model for each problem, as demonstrated in the current study, Boosting is a technique in which a model selection algorithm is used to select the best model for each problem, as demonstrated in the current study. It is also possible to employ the technique of stacking, which is an approach to combining the predictions of several different learning algorithms. Even though the focus of this study is on the four algorithms that indicate their performance, along with their advantages and disadvantages, as well as the reasoning behind their selection for use in the development of the performance appraisal system, future research should explore more predictive analytic algorithms to further explore the situation. However, the fundamental factors that will influence the success of AI adoption in human resource management are not covered in the scope of the studies provided above. Although AI's promise in HRM and the ease with which it appears to increase HRM performance over time are important factors, future studies will look at ways to improve AI in HRM.

References

Abdeldayem, M. M., and Aldulaimi, S. H. (2020). Trends and opportunities of artificial intelligence in human resource management: Aspirations for public sector in Bahrain. International Journal of Scientific and Technology Research, 9(1), 3867-3871.

Bhardwaj, G., Singh, S. V., and Kumar, V. (2020, January). An empirical study of artificial intelligence and its impact on human resource functions. In 2020 International Conference on Computation, Automation and Knowledge Management (ICCAKM) (pp. 47-51). IEEE.

Bilgic, E. (2020). Human Resources Information Systems: A Recent Literature Survey. Contemporary Global Issues in Human Resource Management.

Boxall, P. (2013). Mutuality in the management of human resources: assessing the quality of alignment in employment relationships. Human resource management journal, 23(1), 3-17.

Breiman, L. (1996). Bagging predictors. Machine learning, 24(2), 123-140.

Bühlmann, P. (2012). Bagging, boosting and ensemble methods. In Handbook of computational statistics (pp. 985-1022). Springer, Berlin, Heidelberg.

Dou, J., Yunus, A. P., Bui, D. T., Merghadi, A., Sahana, M., Zhu, M., Chen, C.W., Han, Z. and Pham, B.T., (2020). Improved landslide assessment using support vector machine with bagging, boosting, and stacking ensemble machine learning framework in a mountainous watershed, Japan. Landslides, 17(3), 641-658.

Duan, Q., Ajami, N. K., Gao, X., and Sorooshian, S. (2007). Multi-model ensemble hydrologic prediction using Bayesian model averaging. Advances in Water Resources, 30(5), 1371-1386.

Folino, F., Folino, G., Guarascio, M., Pisani, F. S., and Pontieri, L. (2021). On learning effective ensembles of deep neural networks for intrusion detection. Information Fusion, 72, 48-69.

Gong, M., Xie, Y., Pan, K., Feng, K., and Qin, A. K. (2020). A survey on differentially private machine learning. IEEE Computational Intelligence Magazine, 15(2), 49-64.

Gupta, V., Choudhary, K., Tavazza, F., Campbell, C., Liao, W. K., Choudhary, A., and Agrawal, A. (2021). Cross-property deep transfer learning framework for enhanced predictive analytics on small materials data. Nature communications, 12(1), 1-10.

Jääskeläinen, A., SiUanpää, V., Helander, N., Leskelä, R. L., Haavisto, I., Laasonen, V., and Torkki, P. (2020). Designing a maturity model for analyzing information and knowledge management in the public sector. VINE Journal of Information and Knowledge Management Systems.

Jawabreh, O., Jahmani, A., Khaleefah, Q., Alshatnawi, E., and Abdelrazaqe, H. (2020). Customer Expectation in Five Star Hotels in Aqaba Special Economic Zone Authority (ASEZA). International Journal of Innovation, Creativity and Change, 11(4) 417-438.

Karlaftis, M. G., and Vlahogianni, E. I. (2011). Statistical methods versus neural networks in transportation research: Differences, similarities and some insights. Transportation Research Part C: Emerging Technologies, 19(3), 387-399.

Kolbjørnsrud, V., Amico, R., and Thomas, R. J. (2016). How Artificial Intelligence Will Redefine Management. Harvard Business Review. Retrieved from https://hbr.org/2016/11/how-artificial-intelligence-will-redefine-management.

Laudon, K. C., and Laudon, J. P. (2015). Management information systems (p. 143). Upper Saddle River: Pearson

Lim, S., Wang, T. K., and Lee, S. Y. (2017). Shedding new light on strategic human resource management: The impact of human resource management practices and human resources on the perception of federal agency mission accomplishment. Public Personnel Management, 46(2), 91-117.

Maier, C., Laumer, S., Eckhardt, A., and Weitzel, T. (2013). Analyzing the impact of HRIS implementations on HR personnel's job satisfaction and turnover intention. The Journal of Strategic Information Systems, 22(3), 193-207.

Margherita, A., Sharifi, H., and Caforio, A. (2021). A conceptual framework of strategy, action and performance dimensions of organisational agility development. Technology Analysis & Strategic Management, 33(7), 829-842.

Melville, N. P. (2010). Information systems innovation for environmental sustainability. MIS quarterly, 34(1), 1-21.

Montgomery, D. C., Peck, E. A., and Vining, G. G. (2021). Introduction to linear regression analysis. John Wiley & Sons.

Moretti, F., Pizzuti, S., Panzieri, S., and Annunziato, M. (2015). Urban traffic flow forecasting through statistical and neural network bagging ensemble hybrid modeling. Neurocomputing, 167, 3-7.

Qamar, Y., Agrawal, R. K., Samad, T. A., and Jabbour, C. J. C. (2021). When technology meets people: the interplay of artificial intelligence and human resource management. Journal of Enterprise Information Management.

Saiz-Rubio, V., and Rovira-Más, F. (2020). From smart farming towards agriculture 5.0: A review on crop data management. Agronomy, 10(2), 207.

Schultze, U., and Leidner, D. E. (2002). Studying knowledge management in information systems research: discourses and theoretical assumptions. MIS quarterly, 26(3), 213-242.

Seni, G., and Elder, J. F. (2010). Ensemble methods in data mining: improving accuracy through combining predictions. Synthesis lectures on data mining and knowledge discovery, 2(1), 1-126.

Sony, M., and Naik, S. (2020). Industry 4.0 integration with socio-technical systems theory: A systematic review and proposed theoretical model. Technology in Society, 61, 101248.

Spooner, A., Chen, E., Sowmya, A., Sachdev, P., Kochan, N. A., Trollor, J., and Brodaty, H. (2020). A comparison of machine learning methods for survival analysis of high-dimensional clinical data for dementia prediction. Scientific reports, 10(1), 1-10.

Tambe, P., Cappelli, P., and Yakubovich, V. (2019). Artificial intelligence in human resources management: Challenges and a path forward. California Management Review, 61(4), 15-42.

Trejos, J., Villalobos-Arias, M. A., and Espinoza, J. L. (2016). Variable selection in multiple linear regression using a genetic algorithm. In Handbook of research on modern optimization algorithms and applications in engineering and economics (pp. 133-159). IGI Global.

Ullah, Z., Ahmad, N., Scholz, M., Ahmed, B., Ahmad, I., and Usman, M. (2021). Perceived accuracy of electronic performance appraisal systems: The case of a non-for-profit organization from an emerging economy. Sustainability, 13(4), 2109.

Usman, A. G., Işik, S., and Abba, S. I. (2020). A novel multi-model data-driven ensemble technique for the prediction of retention factor in HPLC method development. Chromatographia, 83, 933-945.

Vu, T. M., Probst, C., Nielsen, A., Bai, H., Buckley, C., Meier, P. S., and Purshouse, R. C. (2020). A software architecture for mechanism-based social systems modelling in agent-based simulation models. Journal of artificial societies and social simulation: JASSS, 23(3).

Wheeler, A. R., and Buckley, M. R. (2021). The Current State of HRM with Automation, Artificial Intelligence, and Machine Learning. In HR without People? Emerald Publishing Limited.

Yariyan, P., Janizadeh, S., Van Phong, T., Nguyen, H. D., Costache, R., Van Le, H., and Tiefenbacher, J. P. (2020). Improvement of best first decision trees using bagging and dagging ensembles for flood probability mapping. Water Resources Management, 34(9), 3037-3053.

Zhang, J., Liu, X., and Li, X. B. (2020). Predictive analytics with strategically missing data. INFORMS Journal on Computing, 32(4), 1143-1156.

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal
Regular Issue, Vol. 11 N. 2 (2022), 147-158
eISSN: 2255-2863
DOI: https://doi.org/10.14201/adcaij.27184

Optimization of Window Size for Calculating Semantic Coherence Within an Essay

Kshitiz Srivastava[a], Namrata Dhanda[b] and Anurag Shrivastava[c]

[a] Ph.D. scholar, Dr. APJ Abdul Kalaam technical university, Lucknow, Uttar Pradesh, India
[b] Professor, Department of computer science, Amity University, Lucknow, Uttar Pradesh, India
[c] Professor, Department of Computer Science & Engineering, Babu Banarasi Das Northern India Institute of Technology, Lucknow, Uttar Pradesh, India
kshitiz.sri@gmail.com, ndhanda@lko.amity.edu, anuragbbdlko@gmail.com

KEYWORDS

automated essay evaluation; semantic attributes; semantic coherence; semantic mining

ABSTRACT

Over the last fifty years, as the field of automated essay evaluation has progressed, several ways have been offered. The three aspects of style, substance, and semantics are the primary focus of automated essay evaluation. The style and content attributes have received the most attention, while the semantics attribute has received less attention. A smaller fraction of the essay (window) is chosen to measure semantics, and the essay is broken into smaller portions using this window. The goal of this work is to determine an acceptable window size for measuring semantic coherence between different parts of the essay with more precision.

1. Introduction

In the world of computer science, automated essay assessment is always seen as a challenge. It is both a natural language processing application and a statistics classification challenge. There are numerous automatic essay evaluation systems that have been designed and implemented, Psalmerosi, F. H. (2019). Some are already on the market, while others are still in the works. Ke, Z., and V. Ng (2019). Thousands of pupils in the United States are scored each year using systems like Project essay grader, which was developed in 1966 and is currently a product of Measurement Inc. Systems like C-rater, on the other hand, were unable to influence the marketplace with their performance, Y. Attali, (2011).

C-rater is now a standard feature of many other grading systems. With advancements in the field of natural language processing, the need for essay grades that are closer to human scores has grown.

All these systems work on attributes, these attributes denote the different features of the essays, Chandrasekaran, D. et al., (2020). Attributes are broadly classified into three categories: style, content, and semantics of the essay, Khatavkar, V., and Kulkarni, P. (2019). These attributes are subdivided into smaller attributes for example lexical sophistication, Readability measures, Lexical diversity, mechanics, content, etc. These sub-attributes may be a group of other smaller attributes for example lexical sophistication may include all or few of the following sub-attributes: number of characters, number of words, number of long words, number of short words, etc. Zupanc, K., and Bosnic, Z. (2014).

All of the proposed systems have prioritized style and content features, with semantic attributes receiving less attention, Azmi, A. M. et. al. (2019). Measuring the semantic of an essay becomes a huge difficulty due to differences in writing style and content length. A few features were proposed in 2014, and these qualities were integrated as an important component of SAGE Zupanc, K., and Bosnić, Z. (2014), allowing us to quantify the semantic coherence between the portions of the essay. These attributes were able to measure semantic coherence by splitting the essay into smaller sections using various reduction techniques, then placing these portions as points in a high-density semantic space and measuring attributes using various methodologies. In one experiment, a window of around 25% of the essay's entire size was taken, and a smaller corpus was identified by moving the window by 10 words Zupanc, K., and Bosnic, Z. (2016). In our work, we are trying to identify the appropriate window size that can increase the overall performance of the system. For the experiment, we have taken data from the Kaggle website, A. Mellor, (2011).

2. Ethics

The field of Automated Essay Evaluation is carried out in different problem areas, A. Mellor, (2011). In the following subsection, we have discussed these problem areas.

2.1. Automated Essay Evaluation Systems and their Approaches

Project Essay Grader was the first automated essay evaluation system, proposed by Elis B Page M. D. Shermis and J. Burstein (2003). It was developed in the mid-1960s. According to Page, the system was a better solution to manual grading. Because of the technology of that time his project was not much accepted by the community, as the operational cost was too high. With the advancement in technology, the internet, and text processing software, development in the field of automated essay evaluation systems took a great pace. Many systems and techniques were proposed and implemented. A few of the mentions systems are *Intelligent essay assessor* Srivastava K., Dhanda N. & Shrivastava A. (2020), *Intellimetric* M. T. Schultz (2013), *Project essay grader*, and *E-Rater* J. Burstein, J. Tetreault, and N. Madnani (2013).

One of the major obstacles in achieving success in this area is the non-availability of any open-source automated essay evaluation system. Most of the systems are either commercially available (PEG, E-rater, Intellimetric, etc.) or under development. The only system whose compiled code along with source code was publicly available was LightSide, developed by Mayfield and Rose Mayfield, E., and Rosé, C. (2010). It was designed to do various these, in which essay grading was one of the major features.

The quality of an essay can be measured by focusing on three basic attribute styles, content, and semantics. Style attribute focuses on the way the essay is written spelling, punctuations, grammar, etc. The content attribute is based on a comparison of the essay with the pre-graded essay. This attribute is used to compares two essays and finding the similarities between them. The semantic attribute is used to verify the correctness of the essay, Chandrasekaran, D., and Mago, V. (2020), D. Higgins, J. Burstein, D. Marcu, and C. Gentile (2004), Darwish, S. M., and Mohamed, S. K. (2019).

Numerous methodologies are used to extract attributes from the essay. Latent semantic analysis is one of the most popular methodologies used. Other methods that are used are pattern matching, sentence similarity networks, Generalized Latent semantic analysis, n-gram approaches, etc. For correctness and consistency semantic networks, ontologies, fuzzy logic, open information extraction, etc. are used Zupanc, K., and Bosnic, Z. (2016); Ferreira-Mello et al. (2019); T. K. Landauer et al.,(1998).

Most of the system uses Machine learning algorithms to create the prediction models like regression modeling, cosine similarity, rule-based expert system, etc, Romero, C., and Ventura, S. (2020). The idea is to use a pre-graded essay to create a prediction model, T. Kakkonen, et al., (2008). This prediction model is then used to grade the rest of the essays, Peng, C., et al. (2020). The performance of such a model is calculated by the comparing score obtained with the actual score, Gao, Y. et al., (2019); P. W. Foltz, et al. (2013); P. W. Foltz, et al. (2007). A brief comparison of a few of the available systems is given in Table 1 Srivastava K., Dhanda N. & Shrivastava A. (2020).

Table 1. A comparison of Some AEE systems

System	Attribute	Prediction Model	Methodology
Autoscore	Style and content	Machine Learning	Statistical
BETSY	Style and content	Bayesian Network	Statistical
Bookette	Style and content	Neural Networks	NLP
CRASE	Style and content	Machine Learning	NLP
E-rater	Style and content	Linear regression	NLP
IEA	Content	Machine Learning	LSA, NLP
Intellimetric	Style and content	Multiple Math metical model	NLP
Lightside	Content	Machine Learning	Statistical
Markit	Content	Linear regression	NLP, PMT
PEG	Style	Multiple linear regression	Statistical
PS-ME	Style	Linear regression	NLP
SAGE	Semantics	Random Forest	OIE, NLP
SAGrader	Semantics	Rule-based expert System	FL, SN
SEAR	Style and content	Linear regression	Statistical

2.2. Semantic Coherence

Coherence is defined as the flow of information from one part of the essay to other parts Mimno, D., Wallach, H., Talley, E., Leenders, M., and McCallum, A. (2011). A highly coherent essay has a low coherence movement whereas a low coherent essay has high movement. The system proposed till now either measures the coherence using supervised or unsupervised approaches, Darwish, S. M., and Mohamed, S. K. (2019).

In unsupervised approaches system usually measures the repetition of words or repetition of phrases assuming that a highly coherent essay has high repeating words and phrases Zupanc, K., and Bosnic, Z. (2016). According to Foltz, P. W., Laham, D., and Landauer, T. K. (1999). a highly coherent essay contains high semantically related words and essay. Hearst Hearst, M. A. (1997) has proposed that an essay can be subdivided into smaller parts and can be used in identifying the repetition of semantically related words and phrases. The most commonly used approach is latent semantic analysis.

In a supervised approach, pre-graded essays are used, D. Higgins,et al., (2004); M. D. Shermis and B. Hamner (2013). According to the centric theory Grosz, B. J., Joshi, A. K., and Weinstein, S. (1995) «The extent a discourse adheres to centering constraints, its coherence will increase and the inference load placed upon the hearer will decrease». This phenomenon helps in locating topic shifts in the essay, Goulart et al., (2018), Foltz, P. W. (2007).

2.3. Semantic Coherence Measurement

The first step towards semantic coherence measurement is to identify the attributes upon which we measure semantic coherence, Injadat, M. et. al. (2020), Janda, H. K. et. al. (2019). Around 72 different attributes (including all types of attributes: style, content, and semantics) are considered by different AES systems Zupanc, K., and Bosnić, Z. (2017), J. Burstein, (2010). We have considered four semantic attributes upon which we will be calculating semantic coherence, Muangkammuen, P., and Fukumoto, F. (2020), D. Higgins et al., (2004). These attributes are attribute-1 (average distance) [8], attribute-2 (Minimum distance), attribute-3 (Maximum distance), and attribute-4 (average nearest neighbor), Romero, C. and Ventura, S. (2020).

To calculate these attributes first data is cleaned by: lowering, stemming, stop-word removal and correcting spelling, Misuraca, M. et al., (2021). Then data obtained is divided into smaller datasets by choosing a window of certain length and moving it by certain words to obtain sequential overlapping parts, J. Burstein, et. al. (2010). For this example, our window consists of 25% of the overall size of data. This window is moved by 10 words to have the next overlapping part Zupanc, K., and Bosnić, Z. (2017).

Each dataset obtained is represented as point in semantic space by first calculating Tf-idf (statistical method). Tf-idf provides the weights to the words in a particular document, Janda, H. K. et al., (2019).

$$\text{tf-idf}(t, d) = \text{tf}(t, d) * \text{idf}(t)$$

This two-dimensional state is obtained with the help of principal component analysis [Zupanc, K. and Bosnic, Z. (2016)]. It is a process of obtaining the principal components and using obtained components to measure the change in the data. PCA may choose the first few principal components ignores the other principal components, Bhatt, R. et al. (2020). From these points, different attributes are measured, calculating the semantic relatedness of the essay Zupanc, K., and Bosnić, Z. (2017). Our experiment is to find the optimal window size that can be moved to obtained sequential overlapping parts, for improving the overall performance of the systems, Azmi et al., (2019).

3. Experiment and Analysis

To find optimal window size a series of the experiments were conducted over different datasets of essays. Each essay has experimented using different window sizes ranging from 1/2 to 1/10th size of

essay, Injadat, M., et al. (2020). Four attributes were selected for the calculation of the window size. These attributes are average distance, minimum distance, maximum distance, and average nearest neighbor Zupanc, K., and Bosnić, Z. (2017). Following are the effects of various window sizes over different attributes.

3.1. Attribute 1 (Average distance)

The average distance attribute helps in identifying semantic relatedness between the sequential parts of the essay. According to Foltz, P. W. et. al. (1999) high semantically related essays have smaller movements whereas less sematic related essays have greater movements between parts of essay. To measure average distance following formula is considered. It helps us understand how well an idea persists within the essay

Distance = distance + sqrt(dx*dx+dy*dy)
Average distance = 2.0*distance/(n*(n-1))

Where n = total numbers of points in semantic space.

For the experimental purpose, the value of the average distance calculated is normalized and then rounded off to two places to get normalized results. The sample of the results is as follows (Table 2).

Table 2. Average window size

Essay/Window	Size 2	Size 3	Size 4	Size 5	Size 6	Size 7	Size 8	Size 9	Size 10
Essay 1	0.19	0.16	0.15	0.14	0.13	0.12	0.12	0.11	0.11
Essay 2	0.2	0.17	0.16	0.14	0.13	0.13	0.12	0.11	0.11
Essay 3	0.18	0.15	0.14	0.13	0.12	0.12	0.11	0.1	0.1
Essay 4	0.28	0.24	0.22	0.2	0.18	0.17	0.16	0.15	0.15
Essay 5	0.29	0.25	0.23	0.21	0.19	0.18	0.17	0.16	0.16
Essay 6	0.26	0.23	0.21	0.19	0.17	0.16	0.15	0.14	0.14
Essay 7	0.23	0.2	0.18	0.16	0.15	0.13	0.12	0.12	0.12
Essay 8	0.42	0.36	0.33	0.29	0.26	0.26	0.23	0.23	0.23
Essay 9	0.45	0.38	0.34	0.29	0.28	0.24	0.23	0.23	0.23
Essay 10	0.38	0.34	0.3	0.28	0.25	0.24	0.22	0.22	0.22
Essay 11	0.21	0.18	0.17	0.16	0.15	0.15	0.14	0.14	0.14
Essay 12	0.25	0.22	0.2	0.18	0.17	0.16	0.15	0.15	0.15
Essay 13	0.19	0.17	0.15	0.14	0.13	0.12	0.11	0.11	0.11
Essay 14	0.18	0.16	0.14	0.12	0.11	0.11	0.1	0.1	0.1

The highlighted result in above table shows that with decreasing window size the value of the result becomes stable. The difference between the value of window size 2 and window size 3 is greater in comparison to window sizes 8, 9, or 10. The difference between the values of window size 8, 9, or 10 is either minimum or null. It can be concluded from this experiment that lower window sizes have stable results.

3.2. Attribute 2 (Minimum Distance)

The minimum distance between the neighboring points shows how well the idea is transferred from one sentence to the next sentence. Minimum distance measures the distance between all the points in the semantic spaces and finds out the minimum movement between the various parts of the essay. This distance helps in understand how well the idea flows between the various parts of the essay. The minimum distance is calculated using the following formula

```
                Distance = distance + sqrt(dx*dx+dy*dy)
if(min>dist):
                min = dist
```

The value of the minimum distance is normalized and rounded off to get a normalized value. Sample results are as follows (Table 3).

Table 3. Minimum Distance

Essay/Window	Size 2	Size 3	Size 4	Size 5	Size 6	Size 7	Size 8	Size 9	Size 10
Essay 1	0.05	0.01	0.01	0.02	0	0.02	0.01	0.01	0.01
Essay 2	0.03	0.01	0.01	0.01	0.01	0.01	0.01	0	0
Essay 3	0.02	0.01	0.01	0.01	0	0.01	0.01	0.01	0.01
Essay 4	0.01	0.01	0.01	0.02	0.01	0.01	0.01	0.01	0.01
Essay 5	0.02	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
Essay 6	0.09	0.04	0.05	0.04	0.04	0.01	0.03	0.03	0.03
Essay 7	0.06	0.02	0.02	0.01	0.02	0.01	0.01	0.03	0.03
Essay 8	0.02	0.03	0.02	0.04	0.03	0.03	0.03	0.03	0.03
Essay 9	0.05	0.07	0.03	0.03	0.02	0.03	0.03	0.01	0.01
Essay 10	0.06	0.03	0.01	0.03	0.02	0.02	0.01	0.01	0.01
Essay 11	0.06	0.02	0.01	0.03	0.02	0.03	0.02	0.02	0.02
Essay 12	0.03	0.02	0.01	0.01	0.01	0.01	0.01	0	0
Essay 13	0.08	0.05	0.05	0.03	0.03	0.03	0.03	0.03	0.03
Essay 14	0.01	0.04	0.01	0	0	0	0.01	0.01	0.01
Essay 15	0.02	0.01	0.01	0	0.01	0.01	0.02	0.02	0.02
Essay 16	0.07	0.03	0.04	0.01	0.01	0.02	0.01	0.01	0.01
Essay 17	0.03	0.02	0.01	0.01	0.01	0.01	0.01	0.01	0.01
Essay 18	0.05	0.07	0.03	0.05	0.06	0.04	0.04	0.04	0.04
Essay 19	0.02	0.01	0	0.01	0.01	0	0.01	0.01	0.01
Essay 20	0.06	0.02	0.02	0.01	0.01	0.01	0.01	0.01	0.01
Essay 21	0.06	0.02	0.01	0.01	0.01	0.01	0.01	0.01	0
Essay 22	0.02	0	0.01	0	0.01	0.01	0.01	0.01	0.01
Essay 23	0.02	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
Essay 24	0.03	0.02	0.01	0.01	0.01	0.01	0.01	0.01	0
Essay 25	0.04	0.02	0.01	0.01	0.01	0.01	0.01	0.01	0.01
Essay 26	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
Essay 27	0.04	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01

From the above table, it can be easily seen that the minimum distance works better with smaller window size (Highlighted cells). Because of window size and overlapping windows sometimes we obtain zero because the corpus obtained is highly identical or the value obtained is very small or negligible. The large variation is not attained as attained in the average distance because a smaller window size causes the result to be more accurate in terms of a fraction. It can be concluded that a lower window size works well with smaller window size.

3.3. Attribute 3 (Maximum Distance)

The maximum distance between the neighboring points shows the breadth of the discussed concept. Maximum distance measures the distance between all the points in the semantic spaces and finds out the maximum movement between the various parts of the essay. This distance helps in understand the breadth of the discussed concept. The maximum distance is calculated using the following formula.

```
Distance = distance + sqrt(dx*dx+dy*dy)
if(max<dist):
max=dist
```

The value of maximum distance is normalized and rounded off to get a normalized value. Sample results are as follows (Table 4).

Table 4. Maximum Distance

Essay/Window	Size 2	Size 3	Size 4	Size 5	Size 6	Size 7	Size 8	Size 9	Size 10
Essay 1	1.49	1.23	1.02	0.94	0.89	0.83	0.75	0.78	0.78
Essay 2	1.01	0.7	0.59	0.57	0.52	0.5	0.49	0.49	0.49
Essay 3	0.86	0.64	0.55	0.51	0.48	0.46	0.45	0.44	0.44
Essay 4	0.85	0.65	0.55	0.52	0.5	0.48	0.48	0.48	0.48
Essay 5	0.71	0.57	0.51	0.46	0.45	0.43	0.41	0.41	0.41
Essay 6	0.7	0.56	0.5	0.48	0.44	0.43	0.43	0.4	0.4
Essay 7	1.58	1.2	1.11	1.01	0.87	0.9	0.88	0.88	0.87
Essay 8	1.36	1.2	1.1	0.9	0.95	0.94	0.82	0.74	0.74
Essay 9	1.5	1.32	1.15	1.02	1.08	1.04	1.04	0.92	0.92
Essay 10	1.43	1.19	1.05	1	0.92	0.86	0.81	0.79	0.79
Essay 11	1.18	1.09	0.99	0.86	0.82	0.75	0.75	0.7	0.7
Essay 12	1.23	0.92	0.81	0.71	0.69	0.63	0.61	0.61	0.61
Essay 13	1.12	0.89	0.78	0.72	0.68	0.66	0.64	0.64	0.64
Essay 14	0.81	0.61	0.53	0.48	0.48	0.45	0.43	0.41	0.41
Essay 15	0.64	0.54	0.49	0.48	0.47	0.45	0.44	0.43	0.43
Essay 16	0.71	0.55	0.52	0.48	0.49	0.46	0.44	0.43	0.43
Essay 17	0.58	0.47	0.38	0.37	0.36	0.34	0.32	0.31	0.31
Essay 18	0.69	0.53	0.45	0.43	0.41	0.41	0.39	0.37	0.37
Essay 19	0.55	0.45	0.35	0.32	0.3	0.3	0.29	0.28	0.28
Essay 20	0.68	0.5	0.42	0.38	0.34	0.33	0.32	0.31	0.31
Essay 21	0.75	0.64	0.59	0.56	0.57	0.53	0.51	0.54	0.54
Essay 22	1.04	0.72	0.63	0.55	0.51	0.52	0.51	0.51	0.51
Essay 23	0.81	0.6	0.51	0.55	0.52	0.51	0.47	0.47	0.47
Essay 24	0.76	0.6	0.51	0.47	0.46	0.45	0.43	0.43	0.43
Essay 25	0.71	0.57	0.51	0.46	0.45	0.43	0.41	0.41	0.41
Essay 26	0.68	0.52	0.46	0.41	0.37	0.35	0.34	0.34	0.34
Essay 27	0.70	0.56	0.5	0.48	0.44	0.43	0.43	0.4	0.4

Similar to the experiment of minimum value, the maximum value also works better with smaller window sizes (Highlighted cells). The large variation is not obtained as obtained in the average distance because a smaller window size causes the result to be more accurate in terms of a fraction. It can be concluded that a lower window size also works well with maximum distance.

3.4. Attribute 4 (Average Nearest Neighbor)

Average nearest neighbor measures how fast an idea develops across an essay. Higher the value slower the idea developed across an idea and lowers the value faster the idea is developed across the essay. To calculate the average nearest neighbor firstly nearest neighbor of each point is calculated. Then the average of the value received is calculated. This result will tell us how fast the idea is developed across the essay [T. Kakkonen et. al. (2008)]. The following formula is used to calculate the average nearest neighbor, P. J. Clark and F. C. Evans, (1954).

$$\overline{d} = \frac{\sum_{i-1}^{N} d_i}{N}$$

Where,
di is nearest neighbor of every point
N is total number of features

The value of the Average nearest neighbor is normalized and rounded off to get a normalized value. Sample results are as follows (Table 5).

Table 5. Average Nearest Neighbor

Essay/Window	Size 2	Size 3	Size 4	Size 5	Size 6	Size 7	Size 8	Size 9	Size 10
Essay 1	3.86	3.48	2.08	1.94	1.72	1.05	0.07	0.07	0.07
Essay 2	1.51	1.18	0.95	0.87	0.7	0.53	0.49	0.64	0.57
Essay 3	0.68	0.76	0.72	0.78	0.65	0.55	0.64	0.58	0.81
Essay 4	0.98	0.88	0.95	0.86	0.84	0.64	0.53	0.57	0.72
Essay 5	0.67	0.59	0.56	0.5	0.64	0.5	0.54	0.5	0.61
Essay 6	0.78	0.64	0.58	0.54	0.58	0.63	0.65	0.7	0.44
Essay 7	0.75	0.57	0.57	0.56	0.59	0.46	0.44	0.34	0.65
Essay 8	0.52	0.6	0.46	0.5	0.37	0.4	0.43	0.42	0.38
Essay 9	0.55	0.5	0.47	0.45	0.55	0.65	0.71	0.61	0.65
Essay 10	0.51	0.47	0.52	0.45	0.51	0.5	0.39	0.43	0.37
Essay 11	0.30	0.27	0.27	0.27	0.24	0.22	0.18	0.21	0.21
Essay 12	0.32	0.28	0.28	0.26	0.28	0.25	0.27	0.26	0.39
Essay 13	0.36	0.35	0.34	0.34	0.36	0.31	0.26	0.26	0.36
Essay 14	5	3.67	2.8	1.68	1.98	1.27	2.06	2.06	0.14
Essay 15	3.25	2.61	2.11	3.57	1.25	1.54	0.33	0.21	0.21
Essay 16	2.27	1.42	1.35	1.43	1.45	0.93	0.52	0.79	0.74
Essay 17	2.79	2.71	1.57	0.69	0.63	1.56	0.81	0.18	0.18
Essay 18	0.28	0.26	0.35	0.33	0.28	0.33	0.33	0.4	0.48

(continued)

Table 5. Average Nearest Neighbor (continued)

Essay/Window	Size 2	Size 3	Size 4	Size 5	Size 6	Size 7	Size 8	Size 9	Size 10
Essay 19	0.34	0.26	0.22	0.19	0.23	0.21	0.2	0.24	0.26
Essay 20	0.33	0.29	0.25	0.27	0.23	0.23	0.3	0.32	0.38
Essay 21	0.28	0.27	0.24	0.23	0.23	0.24	0.3	0.26	0.28
Essay 22	0.35	0.25	0.22	0.24	0.2	0.22	0.17	0.2	0.18
Essay 23	0.33	0.25	0.25	0.23	0.24	0.21	0.24	0.21	0.22
Essay 24	0.29	0.21	0.23	0.2	0.26	0.29	0.31	0.29	0.37
Essay 25	0.29	0.24	0.23	0.2	0.21	0.18	0.28	0.3	0.32
Essay 26	2.57	2.14	2.57	1.8	1.35	1.21	0.88	0.56	0.56
Essay 27	3.70	4.12	3.23	2.44	1.26	0.43	0.43	0.01	0.01

The average nearest neighbor does not show any significant variation with changing window size (Highlighted cell). This means that no effect of varying window size has on an average nearest neighbor.

4. Conclusion and Future Scope

The findings of our experiment show that choosing window size for different qualities has a substantial impact. The value of Attribute 1 (average distance) becomes stable as the window size decreases. Similarly, with lower window sizes, the results for Attribute 2 (minimum distance) and Attribute 3 (maximum distance) are steady. However, when the window size is changed, the property 4 (average nearest neigbour) does not exhibit any significant fluctuation. The window size for four semantic qualities that quantify the coherence between the parts of the essay is proposed in this paper. For attribute 1, attribute 2, and attribute 3, the smaller window size displays the most accurate result. The result obtained is unaffected by the four window sizes used for attribute. To get a better outcome, use a window size that is less than 1/7th the size of the total essay.

In this experiment we have focused on the window size that will help to subdivide the essay into parts, these parts are represented as a point in semantic space from where we can measure different attributes. In our future work, we will focus on the movement of these windows and choosing some appropriate movement techniques to obtain better results for various attributes.

References

A. Mellor, (2011). «Essay Length, Lexical Diversity and Automatic Essay Scoring», Memoirs of the Osaka Institute of Technology, vol. 55, no. 2, pp. 1–14.

Azmi, A. M., Al-Jouie, M. F., and Hussain, M. (2019). AAEE–Automated evaluation of students' essays in Arabic language. Information Processing & Management, 56(5), 1736-1752.

Bhatt, R., Patel, M., Srivastava, G., and Mago, V. (2020). A Graph Based Approach to Automate Essay Evaluation. In 2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp. 4379-4385.

Chandrasekaran, D., and Mago, V. (2020). Evolution of Semantic Similarity--A Survey. arXiv preprint arXiv:2004.13820.

D. Higgins, J. Burstein, D. Marcu, and C. Gentile, (2004). «Evaluating Multiple Aspects of Coherence in Student Essays», in Proceedings of HLT-NAACL, Boston, MA.

Darwish, S. M., and Mohamed, S. K. (2019). Automated Essay Evaluation Based on Fusion of Fuzzy Ontology and Latent Semantic Analysis. In International Conference on Advanced Machine Learning Technologies and Applications, pp. 566-575.

Ferreira-Mello, R., André, M., Pinheiro, A., Costa, E., and Romero, C. (2019). Text mining in education. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery.

Foltz, P. W. (2007). Discourse coherence and LSA. Handbook of latent semantic analysis, 167-184.

Foltz, P. W., Laham, D., and Landauer, T. K. (1999). The intelligent essay assessor: Applications to educational technology. Interactive Multimedia Electronic Journal of Computer-Enhanced Learning, 1(2), 939-944.

Gao, Y., Davies, P. M., and Passonneau, R. J. (2018). Automated content analysis: A case study of computer science student summaries. In Proceedings of the thirteenth workshop on innovative use of NLP for building educational applications, pp. 264-272.

Goulart, H. X., Tosi, M. D., Gonçalves, D. S., Maia, R. F., and Wachs-Lopes, G. A. (2018). Hybrid model for word prediction using naive bayes and latent information. arXiv preprint arXiv:1803.00985.

Grosz, B. J., Joshi, A. K., and Weinstein, S. (1995). Centering: A framework for modelling the local coherence of discourse.

Hearst, M. A. (1997). Text Tiling: Segmenting text into multi-paragraph subtopic passages. Computational linguistics, 23(1), 33-64.

Injadat, M., Moubayed, A., Nassif, A. B., and Shami, A. (2020). Systematic ensemble model selection approach for educational data mining. Knowledge-Based Systems, 200, 105992.

Janda, H. K., Pawar, A., Du, S., and Mago, V. (2019). Syntactic, semantic and sentiment analysis: The joint effect on automated essay evaluation. IEEE Access, 7, 108486-108503.

J. Burstein, J. Tetreault, and S. Andreyev, (2010). «Using Entity-Based Features to Model Coherence in Student Essays», in Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the ACL, no. June. Los Angeles, California: Association for Computational Linguistics, pp. 681–684.

J. Burstein, J. Tetreault, and N. Madnani (2013), «The E-rater Automated Essay Scoring System», in Handbook of Automated Essay Evaluation: Current Applications and New Directions, M. D. Shermis and J. Burstein, Eds. New York Routledge, 2013, ch. 4, pp. 55–67.

Khatavkar, V., and Kulkarni, P. (2019). Trends in Document Analysis. In Data Management, Analytics and Innovation (pp. 249-262). Springer, Singapore.

Ke, Z., and Ng, V. (2019). Automated essay scoring: a survey of the state of the art. In Proceedings of the 28th International Joint Conference on Artificial Intelligence (pp. 6300-6308). AAAI Press.

M. D. Shermis and J. Burstein (2003) «Introduction», in Automated essay scoring: A cross-disciplinary perspective, M. D. Shermis, and J. Burstein, Eds. Manwah, NJ: Lawrence Erlbaum Associates, 2003, pp. 13-16.

M. D. Shermis and B. Hamner, (2013). «Contrasting State-of-the-Art Automated Scoring of Essays: Analysis», in Handbook of Automated Essay Evaluation: Current Applications and New Directions, M. D. Shermis and J. Burstein, Eds. New York: Routledge, ch. 19, pp. 313–346.

M. T. Schultz (2013) «The IntelliMetric Automated Essay Scoring Engine - A Review and an Application to Chinese Essay Scoring», in Handbook of Automated Essay Evaluation: Current Applications and New Directions, M. D. Shermis and J. C. Burstein, Eds. New York: Routledge, 2013, ch. 6, pp. 89–98.

Mayfield, E., and Rosé, C. (2010, June). An interactive tool for supporting error analysis for text mining. In Proceedings of the NAACL HLT 2010 Demonstration Session (pp. 25-28).

Mimno, D., Wallach, H., Talley, E., Leenders, M., and McCallum, A. (2011). Optimizing semantic coherence in topic models. In Proceedings of the 2011 conference on empirical methods in natural language processing (pp. 262-272).

Misuraca, M., Scepi, G., and Spano, M. (2021). Using Opinion Mining as an educational analytic: An integrated strategy for the analysis of students' feedback. Studies in Educational Evaluation, 68, 100979.

Muangkammuen, P., and Fukumoto, F. (2020). Multi-task Learning for Automated Essay Scoring with Sentiment Analysis. In Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing: Student Research Workshop, pp. 116-123.

P. J. Clark and F. C. Evans, (1954). «Distance to Nearest Neighbor as a Measure of Spatial Relationships in Populations», Ecology, vol. 35, no. 4, pp. 445–453.

P. W. Foltz, L. A. Streeter, K. E. Lochbaum, and T. K. Landauer (2013). «Implementation and Applications of the Intelligent Essay Assessor», in Handbook of Automated Essay Evaluation: Current Applications and New Directions, M. D. Shermis and J. Burstein, Eds. New York: Routledge, ch. 5, pp. 68–88.

P. W. Foltz (2007). «Discourse Coherence and LSA», in Handbook of Latent Semantic Analysis, T. K. Landauer, D. S. McNamara, S. Dennis, and W. Kintsch, Eds. Mahwah, New Jersey: Lawrence Erlbaum Associates, Inc., ch. 9, pp. 167–184.

Peng, C., Chen, Y., Kang, Z., Chen, C., and Cheng, Q. (2020). Robust principal component analysis: A factorization-based approach with linear complexity. Information Sciences, 513, 581-599.

Psalmerosi, F. H. (2019). Applying Text Mining and Machine Learning to Build Methods for Automated Grading (Master's thesis, University of Twente).

Romero, C., and Ventura, S. (2020). Educational data mining and learning analytics: An updated survey. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, v10.

Srivastava K., Dhanda N. & Shrivastava A. (2020), An Analysis of Automated Essay Grading Systems, International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-8 Issue-6, March 2020.

T. Kakkonen, N. Myller, E. Sutinen, and J. Timonen (2008). «Comparison of Dimension Reduction Methods for Automated Essay Grading», Educational Technology & Society, v. 11, pp. 275–288.

T. K. Landauer, P. W. Foltz, and D. Laham (1998). «An introduction to latent semantic analysis», Discourse Processes, vol. 25, pp. 259–284.

Y. Attali, (2011), «A Differential Word Use Measure for Content Analysis in Automated Essay Scoring», ETS Research Report Series, vol. 36.

Zupanc, K., and Bosnić, Z. (2017). Automated essay evaluation with semantic analysis. Knowledge-Based Systems, 120, 118-132.

Zupanc, K., and Bosnic, Z. (2014). Automated essay evaluation augmented with semantic coherence measures. In 2014 IEEE International Conference on Data Mining, IEEE, pp. 1133-1138.

Zupanc, K., and Bosnic, Z. (2016). Advances in the field of automated essay evaluation. Informatica, 39(4).

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal
Regular Issue, Vol. 11 N. 2 (2022), 159-178
eISSN: 2255-2863
DOI: https://doi.org/10.14201/adcaij.27271

CHOP: Maximum Coverage Optimization and Resolve Hole Healing Problem using Sleep and Wake-up Technique for WSN

Vipul Narayan and A.K. Daniel

Department of Computer Science and Engineering, Madan Mohan Malaviya University of Technology, Gorakhpur, U.P, India, 273008
vipulupsainian2470@gmail.com, danielak@rediffmail.com

KEYWORDS

sensor nodes; coverage area; network area; quality of services; coverage hole

ABSTRACT

The Sensor Nodes (SN) play an important role in various hazardous applications environments such as military surveillance, forests, battlefield, etc. The Wireless Sensor Network (WSN) comprised multiple numbers of sensor nodes which are used to perform sensing the physical conditions and subsequently transmitting data to the Base Station (BS). The nodes have limited batteries. The random distribution of nodes in the hazardous areas causes overlapping of nodes and coverage hole issues in the network. The Coverage Optimization and Resolve Hole Healing (CHOP) Protocol is proposed to optimize the network's overlapping and resolve the coverage hole problem. The working phases of the proposed protocol are network initialization, formation of the cluster, Selection of Cluster Head, and sleep and wake-up phase. The issues are optimized, and maximum coverage is achieved for a specific sensing range. Using statistics and probability theory, a link is established between the radius of the node and the coverage area. The protocol used the sleep and wake phase to select optimal nodes active to achieve maximum coverage. The proposed protocol outperformed and showed improvements in the network's performance and lifetime compared to LEACH, TEEN, SEP, and DEEC protocols.

1. Introduction

The sensor network domains such as industry, defense, research Wireless Sensor Network (WSN) is widely used nowadays. The sensor networks are randomly deployed in the monitoring area to gather the environmental data and then transmit the collected data to the Base Station (BS). In (Farhat et al., 2018), the authors discussed that the sensor nodes in the network work collectively and autonomously. The coverage is the important metrics for Quality of Service (QoS) in WSN to enhance network lifetime. In (Gupta and Jha, 2019), the authors discussed the limitation of the protocol in terms of the sensing range of the nodes. They have also addressed that coverage and connective are significant issues in deploying the nodes in the sensing area and proposed the Biogeography-based optimization (BBO) technique to optimize the problem. The proposed approach is based on an encoding technique with objective function to optimize the problem occurred in the network. In (Singh et al., 2020), authors discussed due to the impact on the network performance, the authors categorized the coverage problem. Based on the frequency monitored in the network area, there are two categories: sweep coverage and continuous coverage problem. In (Rai and Daruwala, 2016) discussed deterministic and probabilistic sensing models are based on the SN's sensing capacity. The sensing models are further classified into unidirectional and omnidirectional models. The unidirectional sensing model has a finite viewing angle; hence, it does not cover and detect the entire area. The omnidirectional nodes detect the whole area. In (Wang et al., 2015; Aliouane et al., 2016), discussed the mobility model based on Sensor Location Awareness (SLA), sensor sensing range, sensor deployment strategy, which is a very important factor to build a model for coverage protocols. The optimal resource utilization and a hierarchical-based architecture are the essential requirements to perform sensing tasks in the network. Node clustering is the technique where nodes are grouped into multiple classes with similar properties. The data is transferred from Cluster Members (CM) to the respective cluster head (CH), from where it is further finally transferred to the base station. The sensing range adjustment is a crucial strategy for efficient energy utilization in the network. The time between node sensed the data and finally successfully transmitted to BS is known as network lifetime.

The Coverage Optimization and Resolve Hole Healing (CHOP) Protocol is proposed, and the performance is compared for network lifetime and performance of the system. The protocol achieves enhanced network lifetime and transmits more packets to BS.

The rest of the papers are structured as follows: Related work is discussed in Section 2. The proposed Coverage Optimization and Resolve Hole Healing (CHOP) Protocol is discussed in Section 3. The simulation results and validation are deliberated in Section 4. The paper concludes with future directions in the last section.

2. Related Work

The WSN has emerged as a key study area in the domain of artificial intelligence and cloud computing other related fields. In (Nayak et al., 2021), the authors have tried to enhance the Quality-of-Service metrics in WSN. The complexity of the algorithm is unclear that must be computed for practical implementation of the algorithm. In (Narayan and Daniel, 2021), the authors, the deployment of SN throughout the network region is a random process. The CH gathers data and forwards it to the station node. In (Narayan, Daniel and Rai, 2020), the authors suggested the Energy Efficient Two-Tier Region protocol for the data transmission within the network. The capacity to conduct activities utilizing the available Residual Energy (RE) of node and CH nodes. The protocol boosts system efficiency and network stability in terms of lifetime.

In (Prusty, Sethi and Nayak, 2020), the authors focused on a network's route optimization protocol and addressed the issue related to non-uniform distribution of energy, which causes coverage gaps and poor network performance. The technique of movable nodes is used to fix these complex issues. In (Dwivedi, Sharma and Mehra, 2020), the authors devised an Energy Efficient two-stage routing protocol based on the higher residual node is selected as the CH for a specific round. This protocol increased system performance and node RE. In (Ramluckun and Bassoo, 2018), the authors discussed the significance of optimization algorithms in the identification of proper CH based on the route optimization strategy based following metrics on energy consumption, Cluster size, cluster distance, and a total number of clusters created. In (Wang et al., 2010), the authors introduce a protocol for area-aware coverage. The protocol employs SNs with sensing ranges and customizable transmission. The sleep and wake-up methodology is utilized to allocate the duty cycle to each node. In (Shi, Chen and Lin, 2015), the authors have utilized the α-k coverage approach is used to cover the target region with possibility by k nodes. The method maintains some SN active in order to complete the task in the network. The protocol optimizes energy usage and increases network longevity.

In (Narayan and Daniel, 2019), the authors devised a strategy to reduce the network's overlapping issue. For total coverage, the radius of the nodes and RE are chosen as parameters. The protocol mitigates the overlapping problem, and the overlapped SN is used to increase network longevity via the sleep and wake-up mechanism. In (Salam and Hossen, 2020), The authors devised the LEACH (Low Energy Adaptive Clustering Hierarchy) technique, which works best for homogeneous nodes. The restrictions of a protocol are routing, CH selection, data transfer, and aggregation. In (Behera, Samal and Mohapatra, 2018), The authors described the LEACH protocol feature in which every node has an equal chance of becoming CH. Because the protocol uses a single hop for data transfer duties in the network, it requires more energy. The effective use of energy remains a difficult challenge in the network. In (Al-Shalabi, Ababneh and Abdulraheem, 2021), the authors presented the energy hole removal method to solve the energy hole issue. The sleep and wake-up method conserves the power of SNs. The threshold value is used to determine the SN's available and RE for the data transmission operation.

In (Aliouat and Harous, 2012), the authors recommended two Threshold Sensitive Energy Efficient Sensor Network (TEEN) protocol variations (WBM-TEEN and WB-TEEN). Each SN connects with its neighboring SN in the cluster, reducing the number of data packets delivered and received in the network. WB-TEEN and WBM-TEEN outperformed the TEEN and LEACH protocols in terms of lifespan. In (Mittal, Singh and Sohi, 2017), the authors proposed an application-specific low power consuming routing protocol that employs (distance between RE and CHs, distance from BS) as CH selection parameters. When compared to LEACH protocol versions, the protocol outperforms them. In (Peng et al., 2015), The authors presented a scalable clustering technique (ESCS) and an energy-aware and for WSN. The method is based on the notion of the Barabasi-Albert (BA) model. The CH is chosen depending on the network architecture and the SN's energy. The ESCS simulation result improves the scalability period and network lifespan. In (El Handri and Idrissi, 2020), The Node Rank Algorithm (NRA) was proposed by the authors. The NRA selects CH based on the number of linkages within SN and the path's cost. The NRA algorithm has solved the problem of the conventional CH selection procedure, which caused unanticipated CH failure in the previous LEACH version. The NRA outperforms several variants of the LEACH protocols in terms of performance and network longevity. In (Cai et al., 2019), The authors suggested a sensor network routing technique. Data packet transfer requires a lot of energy in the network. To reduce data transmission in the network, the approach utilizes both clustering methods and multi-hop routing. In (Luo and Fu, 2017), The authors explored certain clustering algorithms that need the position of SN to be determined using the Global Positioning System (GPS).

However, owing to GPS's high energy consumption and expense, the GPS position is not constantly accessible. The authors proposed a hybrid clustering and routing system that use both qualitative binary data connection and quantitative location data.

In (Singh and Malik, 2017), The authors introduced a sensor network Stable Election Protocol (SEP). It has a two-level architecture and utilizes Normal Nodes (NN) and Advance Nodes (AN). The SEP protocol's CH election probability depends on Initial Energy. The protocol outperforms the LEACH protocol in terms of network stability. In (Nehra, Sharma and Tripathi, 2020), The authors developed the distributed Energy Efficient Clustering (DEEC) routing protocol. The technique allows for multilayer heterogeneity. In multilayer heterogeneity, the SN energy is distributed randomly in a specified energy gap. For the CH selection procedure, the DEEC protocol employs the probability ratio between the RE of each SN and the network estimated average energy. DEEC, when compared to LEACH and SEP, increases network longevity.

In (Chaturvedi and Daniel, 2017), the authors presented a sleep/wake-up methodology for network coverage issues. For trust computation, the protocol's CH selection employs distance, RE, and node as parameters. The protocol outperforms the Disjoint Set Cover (DSC) protocol. In (Khan et al., 2020), the authors introduced the connectivity and energy efficiency algorithm. The few SN are utilized to monitor the data, while the rest of the nodes are retained in sleep mode for future usage and increased network longevity. In (Harizan and Kuila, 2019), the authors recommend the sleep and wake-up strategy. Each node in the network is allotted a timeframe for performing the operation. In a decentralized method, each node freely decides its actions in the specified time frame. The technique implementation performance of the suggested strategy increases network longevity. In (Zimmerling, Mottola and Santini, 2020), the authors presented a method for monitoring network node density. To increase network performance, the reinforcement learning approach allows certain nodes to sleep while others remain active. In (Nithya and Jegadeeshwari, 2019), the authors proposed the Energy-Efficient Scheduling with Deep Reinforcement (E2S-DRL) method. The technique operates in stages to reduce network latency. In the first step, zonal-based clustering is employed for data gathering. The DRL method is utilized in the second phase to award a duty cycle to each node, and routing is accomplished in the third phase utilizing Optimization techniques.

The authors (Rajawat, Jain and Barhanpurkar, 2021) proposed a multi-level-based data fusion protocol for clustering, which is used to increase the coverage area and increase the reliability of the sensor network. The concept of the static sensor is used as a point of interest so that the given area is covered. The two key performance indicators are coverage ratio and network lifetime. The performance analysis showed that using minimum active nodes in the network increases network lifetime. The authors (Al-Fuhaidi et al., 2020) proposed a protocol for maximizing the coverage in the network and minimizing the number of nodes deployed in a given field of interest to overcome the issue of high energy consumption and cost. The proposed protocol used a Probabilistic Sensing Model (PSM) in contrast with the Harmony Search Algorithm (HSA) to balance the network coverage and deployment cost. The HSA works in the protocol to achieve maximum coverage with a minimum number of sensor nodes in the network. The probabilistic sensing model (PSM) is used to deploy the nodes in the network efficiently. The proposed protocol is simulated for a small area where the coverage is almost 100% and the deployment cost is reduced by nearly 50%. The comparative analysis with other protocols achieved the deployment cost reduced by a factor of 10%. In the future, the protocol will be enhanced to simulate a large field of interest. The authors (Nandagopal and Ramesh, 2020) proposed an Optimal Minimum Covering Spanning Tree (OMST) protocol to select the optimal path for data transmission in the WSN. The concept of polling points is proposed in the protocol for enhanced data gathering and reduced time for a data packet to reach the sync node. Low Energy Adaptive Clustering

Hierarchy (LEACH) is the two-level clustering protocol. The complexity of the generated tree in the deployment area is reduced using the Particle Gene Swarm (PGS) optimization technique in the protocol. Different performance metrics such as energy, delay, overhead, delivery ratio is used to simulate the performance enhancement over other existing protocols. Optimal path selection in the generated tree is used in the protocol to be overcome for high-performance WSN. The authors(Chen, Xu and Wang, 2019) proposed a novel WSN based energy-efficient coverage protocol based on a genetic algorithm. A two-dimensional simulation model is proposed to represent a sensing area using the minimum number of sensor nodes. The genetic algorithm incorporated with the k-covered technique to compute the minimum number of potential positions of sensor nodes so that all targets must be covered. The simulated result showed performance enhancement and achieved a high degree of covering the targets with minimum wastage of energy in the network. The use of the proposed protocol in the Dimension area is yet to be implemented and is open for research. The authors (So-In, Nguyen and Nguyen, 2019) presented a distributed deployment protocol to enhance WSN area-coverage called coverage hole-healing algorithm (CHHA). The working of the protocol is divided into two stages. In the first stage a novel approach called Delaunay triangulation topology proposed for detecting hole-boundary nodes. In the second stage a hole healing algorithm is used to track the movements of the nodes in the network to heal the coverage hole with minimum moving distance. The three parameters are used in the simulation are the number of SNs moved, the movement distance, and the overall energy consumption of movement where the protocol outperformed better results than existing algorithm. It is also suggested to enhance for global optimization of protocol using soft-computing techniques.

The authors (Khan and Nguang, 2021) proposed a dual-sensing algorithm, a modified version of a previously built technique for sensing and scheduling in road networks. In such a network, the targets can enter from either side of the road. The algorithm proceeds with scanning the road with a two-way scan wave. This approach reduces half of the detection time in the network compared to the existing technique. One drawback here is that it decreases network lifetime. The algorithm is compared to other schemes such as duty cycling and always-awake schemes. In addition, a comprehensive labeling scheme base hole healing algorithm is proposed in the road network.

The authors (Xu, Jiao and Tian, 2020) proposed an efficient optimization methodology by combining the node scheduling with the routing protocol. The new algorithm is proposed in order to avoid redundant coverage and find out the minimum number of nodes in the network. To overcome the issue of dead nodes, a wake-up scheme is also proposed to activate only appropriate nodes replaced with the dead nodes. Optimized energy consumption in data communication is achieved in the proposed protocol using a minimum coverage set incorporated with the new routing protocol. The CH selection task is performed by considering the residual energy of nodes, distance to the sink node, and the probability of the node. To balance the energy consumption in the networks, a few uneven clusters are also formed in the proposed protocol. The proposed protocol needs some changes in order to adapt to the unreliability of the WSN channel that may lead to data missing due to transmission failure. The proposed algorithm must be combined with efficient channel encoding or such kind of data retransmission schemes in real scenarios.

The authors (Verma and Sharma, 2018) proposed a decentralized localization algorithm for the detection and restoration of coverage holes in WSN. The proposed algorithm is called the Coverage Hole Detection and Restoration algorithm. Energy and time are the key parameters that are improved to enhance the overall performance of the network. Convex and non-convex holes are overcome in the proposed work. The concept of intersection point-based detection and restoration process of coverage hole is demonstrated in the work where a smaller number of patch nodes were used. The advantage of this algorithm is that it can be applied to any kind of area (polygon).

3. Coverage Optimization and Resolve Hole Healing (CHOP) Protocol

The Ψ * Ψ m^2 network area and the BS are situated in the middle of the network. The working phases of the proposed protocol are Initialization of the Network, Formation of the Cluster, Selection of Cluster Head, and sleep and wake-up phase. The issues are optimized, and maximum coverage is achieved for a specific sensing range. Using statistics and probability theory, a link is established between the radius of SN and the coverage area. The protocol used the sleep and wake phase to select the minimum number of nodes active to achieve maximum coverage. The CH node aggregates the data and sends it to the BS.

3.1 Initialization of the Network

The 150 sensor nodes are randomly distributed in the Ψ * Ψ m^2 network area for the simulation purpose. The simulation is performed for 6000 rounds of packet transmission. The position of BS is the middle in the network. The notation and description are shown in Table 1.

Table 1. Notation and description

S. No.	Notation	Definition
1	Ψ	Network Region
2	S_N	Nodes
3	λ	PD density
4	P_D	Poisson Distribution
5	lb	Lower limit [0, k-1]
6	hb	Upper limit $h \geq 0$
7	S_R	Sensing radius
8	C_P	Coverage Probability
9	C_R	Communication radius
10	Φ	Area of detection

The CHOP protocol assumptions.

- The SN is distributed randomly.
- The SN, BS, and CH are all in sync with one another.
- The position of BS is fixed and continuous supply power.
- The nodes have limited battery power.
- The noise factor and signal interference are disregarded.
- In the network region, we are considering 150 effective nodes.

3.2 Formation of the Cluster

The SN gathers data from the environment either individually or in interaction with other SN. Clustering is a method that divides SN into various, and data is transmitted to BS.

The following are some definitions:

Definition 1. Connectivity: The network region ($\Psi * \Psi$ m^2) is covered by S_N number of Sensor Nodes. The communication and sensing radius of S_N is C_R and S_R, respectively. Thus, the essential condition to maintain the connectivity is:

$$C_R \geq 2 \times S_R \tag{1}$$

Definition 2. Coverage: The Sensor Node S_N position is adjusted by a stationary Poisson point distribution. The Poisson Distribution (P_D) is:

$$P_D(r,\lambda) = e^{-\lambda}\frac{\lambda^r}{r!}, r = 1,2,\ldots,n \tag{2}$$

Where $\lambda N \times P$, i.e., the PD density. The value of N is large, that is, the number of Sensor Nodes (S_N) in the area of detection Φ and $S_N >> 10$. The value of P is small and is defined as coverage probability (C_P). The C_P is the ratio of the area of sensing to the area of detection Φ and $C_P << 0.1$.

$$C_p = \frac{\pi S_R^2}{\Phi} \tag{3}$$

$$\lambda = S_N \times C_P \tag{4}$$

$$\lambda = S_N \times \frac{\pi S_R^2}{\Phi} \tag{5}$$

Let «α» be the network coverage probability value. The neighboring nodes of S_N in the Φ can attain the value «α» in the interval of [r - lb, r + hb].

Substituting the λ in Eq. (2),

$$P_D(r,\lambda) = e^{-S_N \times \frac{\pi S_R^2}{\Phi}} \frac{\left(S_N \times \frac{\pi S_R^2}{\Phi}\right)^r}{r!}$$
$$P_D(r-lb \leq \alpha \leq r+hb) \tag{6}$$

$$\sum_{\alpha=r-lb}^{r+hb} e^{-S_N \times \frac{\pi S_R^2}{\Phi}} \frac{\left(S_N \times \frac{\pi S_R^2}{\Phi}\right)^\alpha}{r!} \tag{7}$$

Where $lb \geq 0 \wedge lb \leq r-1$, $hb \geq 0$

The partial derivative is calculated with respect to S_R.

$$\frac{\partial P_D}{\partial S_R} = \sum_{\alpha=r-lb}^{r+hb} S_N \times \frac{2\pi S_R}{\Phi} e^{-S_N \times \frac{\pi S_R^2}{\Phi}} \left[\frac{\left(S_N \times \frac{\pi S_R^2}{\Phi}\right)^{\alpha-1}}{(\alpha-1)!} - \frac{\left(S_N \times \frac{\pi S_R^2}{\Phi}\right)^{\alpha}}{\alpha!} \right] \tag{8}$$

To achieve maximum coverage at sensing radius S_R, set the partial derivative to zero.

$$\frac{\partial P_D}{\partial S_R} = 0 \tag{9}$$

Then,

$$\frac{\left(S_N \times \frac{\pi S_R^2}{\Phi}\right)^{r-lb-1}}{(r-lb-1)!} - \frac{\left(S_N \times \frac{\pi S_R^2}{\Phi}\right)^{r+hb}}{(r+hb)!} = 0 \tag{10}$$

$$\frac{\left(S_N \times \frac{\pi S_R^2}{\Phi}\right)^{r-lb-1}}{(r-lb-1)!} = \frac{\left(S_N \times \frac{\pi S_R^2}{\Phi}\right)^{r+hb}}{(r+hb)!} \tag{11}$$

$$\frac{(r+hb)!}{(r-lb-1)!} = \frac{\left(S_N \times \frac{\pi S_R^2}{\Phi}\right)^{r+hb}}{\left(S_N \times \frac{\pi S_R^2}{\Phi}\right)^{r-lb-1}} \tag{12}$$

$$\frac{(r+hb)!}{(r-lb-1)!} = \left(S_N \times \frac{\pi S_R^2}{\Phi}\right)^{hb+lb+1} \tag{13}$$

$$S_R^2 = \frac{\Phi}{S_N \pi}\left[\frac{(r+hb)!}{(r-lb-1)!}\right]^{\frac{1}{hb+lb+1}} \tag{14}$$

$$S_R = \sqrt{\frac{\Phi}{S_N \pi}\left[\frac{(r+hb)!}{(r-lb-1)!}\right]^{\frac{1}{hb+lb+1}}} \tag{15}$$

When the S_N is confirmed, then S_R is taken from the above Eq. (15) in the interval [*r-lb*, r+*hb*] to achieve a higher coverage probability for the network.

A set of SN, $\Omega = \{S_1, S_2, \ldots, S_N\}$ covering the network area (Ψ) where $S_N(i)$ is an individual sensor that belongs to Ω and $i \in [1, 2, 3\ldots, 150]$. The 150 SN are distributed randomly in the 100 * 100 m^2 network area. The target points $P = \{P_1, P_2, \ldots, P_M\}$is to be covered. The D (S_i, P_j) denotes the distance between the ($S_N(i)$, P(j)). The coverage function $F(P)$ is defined as:

$$F(P) = \sum_{i=1}^{N} f\left(D(S_i, P)\right) \tag{16}$$

$$f\left(D(S_i, P)\right) = \begin{cases} 1, if\left(D(S_i, P)\right) \le S_R \\ 0, otherwise \end{cases} \tag{17}$$

Eq. (17) represents the binary value of the coverage function, where 1 indicates that P is covered by $S_N(i)$. At the same time, P can be covered by multiple $S_N(i)$. If P is covered by k number of $S_N(i)$ then F(P) = k. The coverage of the network with respect to the deployed $S_N(i)$ is computed as:

$$F(\psi) = \min_{\forall P \in \Psi} f\left(D(S_i, P)\right) \tag{18}$$

Thus, Eq. (18) is also known as a function of network coverage. The F(Ψ) is described as the least value of f(D(S_i, P)) for all the possible values of P in Ψ.

3.3 Selection of Cluster Head

The selection of CH is chosen among the active Sensor Nodes. The CH selection depends on two parameters, i.e., Distance and Energy. The CH aggregates all data and transmit it to BS. The CH selection by using Eq. (19).

$$C_H(i) = c \times \frac{D_{max} - D_{BS}\left(S_N(j)\right)}{D_{max} - D_{min}} + (1-c) \times \frac{R_E\left(S_N(j)\right)}{I_E} \tag{19}$$

Where c is constant, D_{max} is Maximum Distance, D_{min} is Minimum Distance, D_{BS} is Distance to BS, R_E is Residual Energy, I_E is Initial Energy.

Algorithm 1. Cluster Head Selection

Initialization: S_N = Sensor Nodes, C_N = Clusters Number, BS = Base Station, R_E = Residual Energy, I_E=InitialEnergy,D_{max}=MaximumDistance,D_{min}=MinimumDistance,D_{BS}=DistancetoBS,c=constant, C_H = Cluster Head, C_M = Cluster Member, S_R = Sensing radius, Φ = Detection area, C_P = Coverage Probability, λ = Poisson Distribution Density T = Threshold.

1: Initialize parameters S_R, Φ, C_P, S_N.
2: Calculate $C_P = \frac{\pi S_R^2}{\Phi}$
3: Calculate λ = $S_N \times C_P$
4: Calculate S_R for max C_P.
5: **for** each C_H(i) 1: C_H do
6: **for** each S_N(j) 1:S_N do
7: R_E = Calculate Residual Energy (S_N(j))
8: D_{max} = Calculate Distance (S_N(j))
9: End **for**
10: End **for**
11: **for** each S_N(i) 1:S_N do
12: Calculate $C_H(i) = c \times \frac{D_{max} - D_{BS}\left(S_N(j)\right)}{D_{max} - D_{min}} + (1-c) \times \frac{R_E\left(S_N(j)\right)}{I_E}$
13: T←rand (0,1)
14: **if** (C_H(i) > T)
15: C_H = CH_Node (S_N(i))
16: **else**
17: C_M = CM_Node (S_N(i))
18: End **if**
19: End **for**

3.4 Sleep and Wake-up

The Sensor Nodes are preserved in sleep mode to conserve the energy of the network. The network area is initially covered by wake-up nodes, and after a period of time, when the wake-up nodes' energy is completely exhausted, the sensor nodes in the sleep mode are turned on. There are four steps in the sleep and wake-up scheduling technique. In the first step, redundant sensor nodes are switched to sleep mode. In the second step, the common or overlapping area is minimized. In the third step, the radius of each node is increased up to the threshold value. In the fourth step, the sleeping nodes are turned on when the wake-up nodes' energy is completely exhausted.

Step 1: The redundant nodes are switched to sleep mode. The following condition identifies the redundant nodes. Let us consider three intersecting sensor nodes S_{N1}, S_{N2}, S_{N3}. The node S_{N3} is redundant if the value of the threshold, i.e., 30% of the radius of S_{N1} is greater than or equal to the distance between (S_{N1}, S_{N2}) and (S_{N1}, S_{N3}) as shown in Figure 1.

Step 2: The common area is minimized. The common area is calculated as:

$$A_C = S_{R1} + S_{R2} - D_{(SlN1,S_{N2})} \tag{16}$$

Where common area is denoted as A_C, S_{R1} is the radius of S_{N1}, S_{R2} is the radius of S_{N2}, and D (S_{N1}, S_{N2}) is the distance between sensors S_{N1} and S_{N2}. The Nodes lying within the S_R of other SN are identified and switched to sleep mode. For example, if S_{R1} is greater than or equal to the D (S_{N1}, S_{N2}), then node S_{N2} lies within the sensing radius of node S_{N1}. Thus, node S_{N2} is switched to sleep mode.

Step 3: Each node's radius is increased until it reaches a threshold value of 25% of its initial radius, at which point it intersects the border of a neighbor node.

Step 4: When the energy of the wake-up nodes is totally exhausted, the sleeping nodes are turned on.

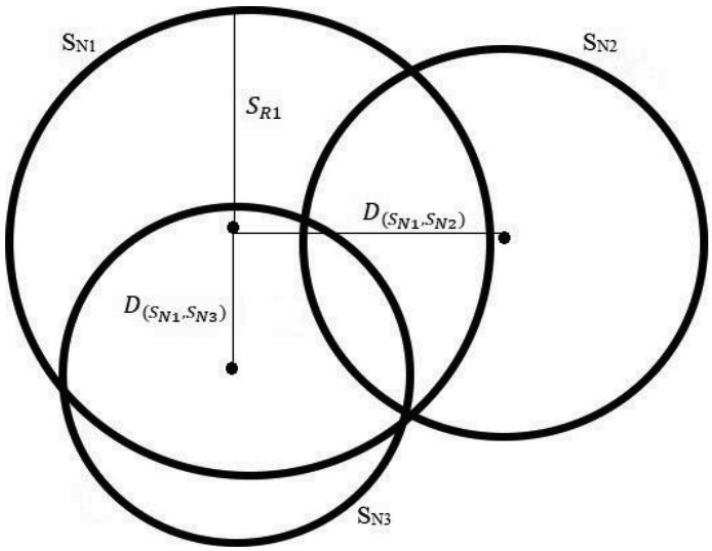

Figure 1. Redundant Sensor Nodes

Algorithm 2. Sleep and Wake-up Scheduling

Initialization: S_N = Nodes, I_N = Intersect Nodes, $S_{R(i)}$ = Radius, $S_{NR(i)}$ = Nodes with Radius $S_{R(i)}$, $D_{PQ(i)}$ = Distance between sensor P and Q L_T = Lifetime, C_1 = Common length, N_N = Nearest Neighbor Node.

for node 1: S_N do
  if ($D_{PQ(i)} \leq (S_{R(i)} * 0.3)$)
   Sleep ($S_{NR(i)}$)
  End **if**
End **for**
for in 1: S_N do
  $S_{R(i)} = S_{R(i)} - \min (C_1, C_2, C_3$---------$C_1)$
End **for**
for Node 1: S_N do
  if ($D_{PQ(i)} \leq S_{R(i)}$)
   Sleep ($S_{NR(i)}$)
  End **if**
End **for**
for Node 1: S_N do
  $S_{R(i)} = S_{R(i)} + S_{NR(i)} \times 0.25$
End **for**
for every non –intersect node
  $S_{R(i)} = S_{R(i)} + 1$;
End **for**
for Node 1: S_N do
  if ($L_T == 0$)
   Awake (N_N)
  End **if**
End **for**

4. Simulation Performance and Validations

Simulation is performed in MATLAB software. Table 2 value (Narayan and Daniel, 2021) is used for the proposed protocol simulation parameter. The simulation for a network lifetime of 150 nodes is performed for a 100*100 m^2 network area. The lifetime is evaluated for successfully 6000 rounds of packet transmission. The proposed protocol is contrasted with LEACH, DEEC, TEEN, and SEP protocols in terms of network performance and packet successfully transmitted to BS. The CHOP improves the system performance for the validation of research work (Mohamed, Khalil and Hammad, 2018).

4.1. LEACH and CHOP-LEACH Network Lifetime

The simulation result in Figure 2 shows that the CHOP-LEACH protocol achieves a better stability period and improves network performance compared with the LEACH protocol. The first packet drop indicates that the network will be unstable for LEACH protocol at 1499 rounds and the CHOP-LEACH at 2031 rounds. The LEACH protocol lifetime is 3099 rounds, whereas CHOP-LEACH is 3956 rounds.

Table 2. Simulation Value (Narayan and Daniel, 2021)

Symbol		Values
Ψ	Network Region	100 m^2
$\mathbf{S_R}$	Sensing radius of Node	Four units
$\mathbf{E_{TX}}$	Energy for Transmission	50nj//bit
$\mathbf{E_{DA}}$	Energy for Aggregation	5//nj/bit
P	CH Selection Probability	0.1
$\mathbf{E_{amp}}$	Dissipation Energy	10 pJ//bit/m^2
$\mathbf{E_{fs}}$	Amplifier power consumption ($d<d_0$)	10 pJ//bit//m^2
$\mathbf{E_{rx}}$	Amplifier energy consumption($d\geq d_0$)	0.0013pJ//bit/m^4
$\mathbf{E_0}$	Initial energy per node	0.50 Joule

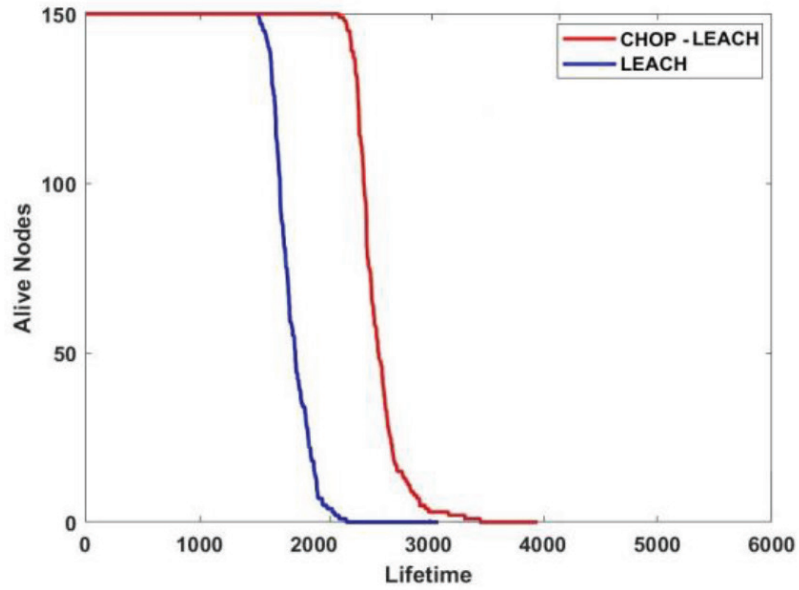

Figure 2. LEACH Vs. CHOP-LEACH

4.1.1. LEACH Vs CHOP-LEACH Packet transmitted to BS

The CHOP-LEACH utilizes optimal energy consumption inside the network, enhances node residual energy, and transmits more packets to BS. The 2.6×10^4 packets are transmitted by LEACH protocol, whereas 3.4×10^4 packets are transmitted by CHOP-LEACH to BS are shown in Figure 3.

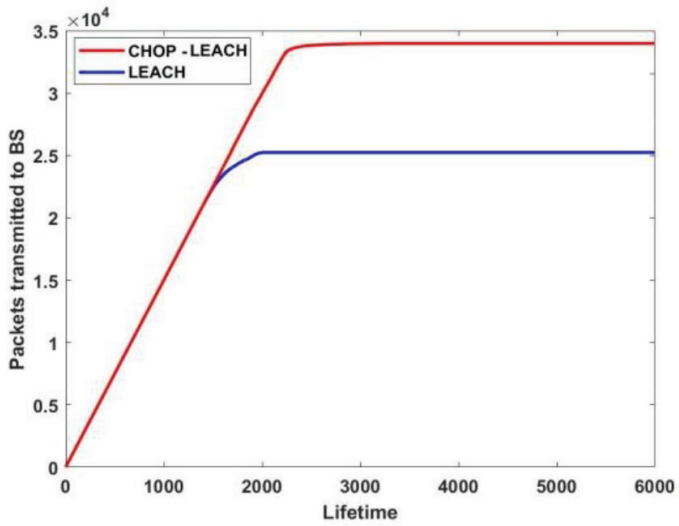

Figure 3. LEACH Vs. CHOP-LEACH

4.2. TEEN and CHOP-TEEN Network Lifetime

The simulation performance in Figure 4 shows CHOP-TEEN protocol achieves a better stability period and improves network performance compared with the TEEN protocol. The first packet drop indicates that the network will be unstable for TEEN protocol at 2121 rounds and the CHOP-TEEN at 2311 rounds. The TEEN protocol lifetime is 4699 rounds, whereas CHOP-TEEN is 5073 rounds.

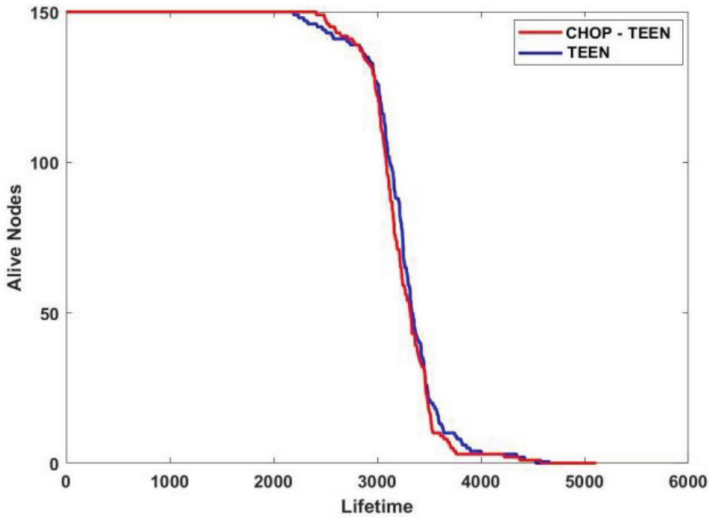

Figure 4. TEEN Vs. CHOP-TEEN

4.2.1. TEEN Vs CHOP-TEEN Packet transmitted to BS

The CHOP-TEEN utilizes optimal energy consumption inside the network, enhances node residual energy, and transmits more packets to BS. The 3.1×10^4 packets are transmitted by TEEN protocol, whereas 3.4×10^4 packets are transmitted by CHOP-TEEN to BS is shown in the Figure 5.

4.3. SEP and CHOP-SEP Network Lifetime

The simulation result in Figure 6 shows that the CHOP-SEP protocol achieves a better stability period and improves network performance compared with the SEP protocol. The first packet drop indicates that the network will be unstable for SEP protocol at 1420 rounds and the CHOP-SEP at 2110 rounds. The SEP protocol lifetime is 4051 rounds, whereas CHOP-SEP is 5251 rounds.

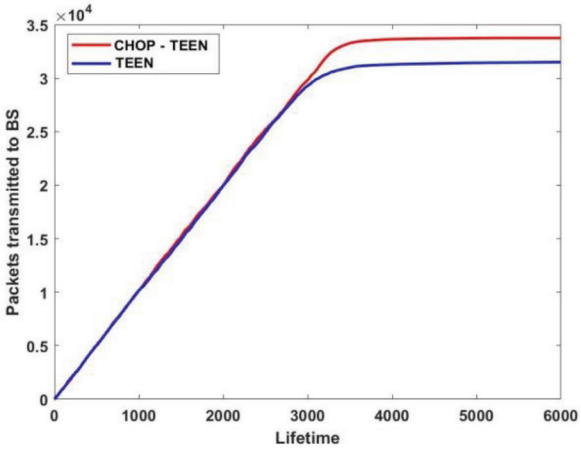

Figure 5. TEEN Vs CHOP-TEEN

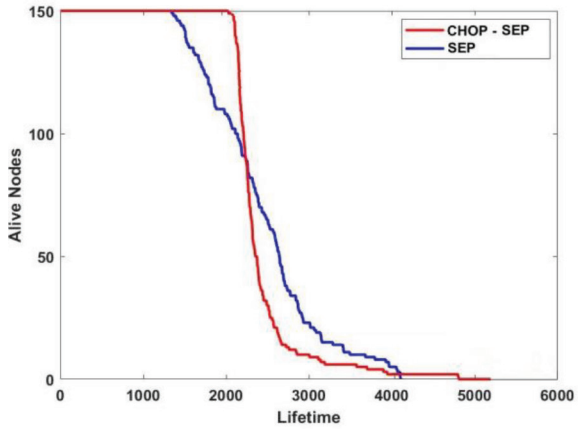

Figure 6. SEP Vs. CHOP-SEP

4.3.1. SEP Vs CHOP-SEP Packet transmitted to BS

The CHOP-SEP utilizes optimal energy consumption inside the network, enhances node residual energy, and transmits more packets to BS. The 3.2×10^4 packets are transmitted by SEP protocol, whereas 3.9×10^4 packets are transmitted by CHOP-SEP to BS is shown in Figure 7.

4.4. DEEC and CHOP-DEEC Network Lifetime

The simulation result in Figure 8 shows that the CHOP-DEEC protocol achieves a better stability period and improves network performance compared with the DEEC protocol. The first packet drop indicates that the network will be unstable for the DEEC protocol at 2101 rounds and the CHOP-DEEC at 2231 rounds. The DEEC protocol lifetime is 4701 rounds, whereas CHOP-DEEC is 5050 rounds.

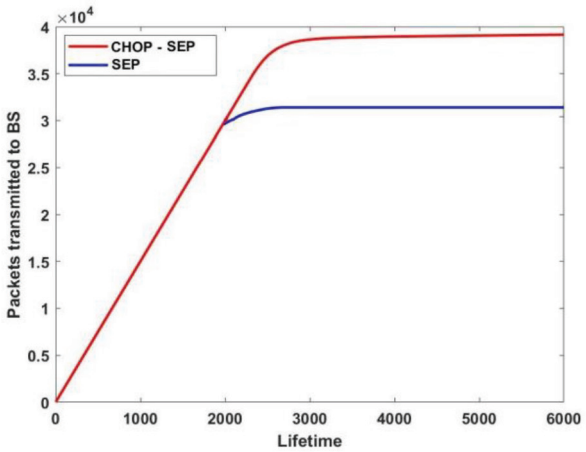

Figure 7. SEP Vs CHOP-SEP

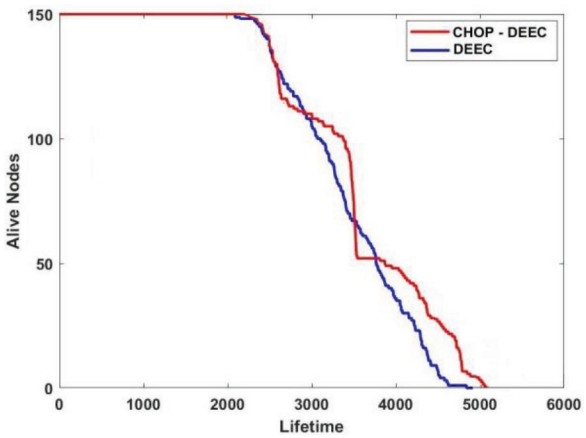

Figure 8. DEEC Vs. CHOP-DEEC

4.5. DEEC Vs CHOP-DEEC Packet transmitted to BS

The CHOP-DEEC utilizes optimal energy consumption inside the network, enhances node residual energy, and transmits more packets to BS. The 1.4×10^5 packets are transmitted by DEEC protocol, whereas 2.2×10^5 packets are transmitted by CHOP-DEEC to BS is shown in Figure 9.

The performance of CHOP Protocol for the First Packet Drop (FPD), Last Packet Drop (LPD), and the number of packets successfully transmitted to station node is shown in Figure 10. Table 3 shows that the simulation performance of CHOP protocol in terms of FPD, LDP, and packet transmitted to BS.

The overlapping of nodes and coverage hole issues is minimized. The proposed protocol CH selection is based on minim BS distance and maximum node RE. The sleep and wake-up concept is used to keep the minimum no of nodes to be active and adjust the load distribution across the network.

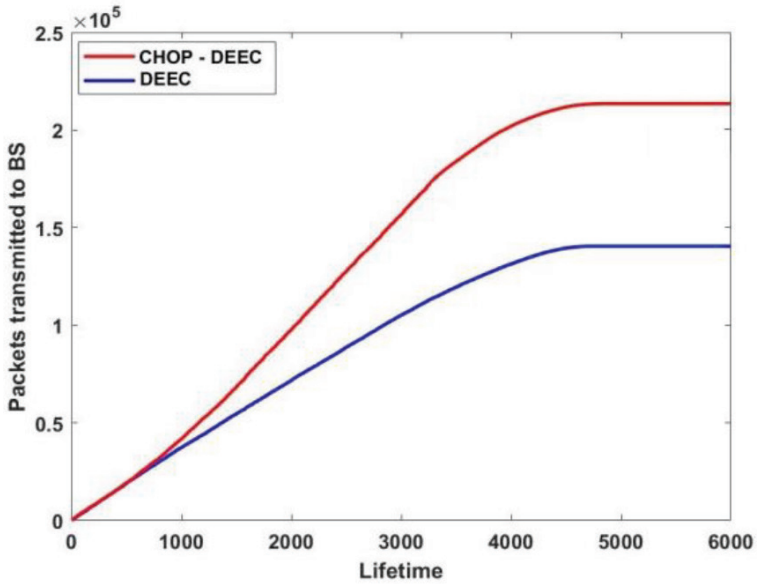

Figure 9. DEEC Vs CHOP-DEEC

5. Conclusion and Future Work

In this paper, we have focused on two fundamental issues that are overlapping of nodes and the coverage hole problem. The CHOP protocol is proposed to overcome the issues. The sleep and wake-up technique is used to cover the network with minimum active nodes and adjust the load distribution across the network. The Poisson distribution function is applied for maximum network coverage in the network.The CHOP protocol is contrasted with traditional routing protocols such as LEACH, TEEN,

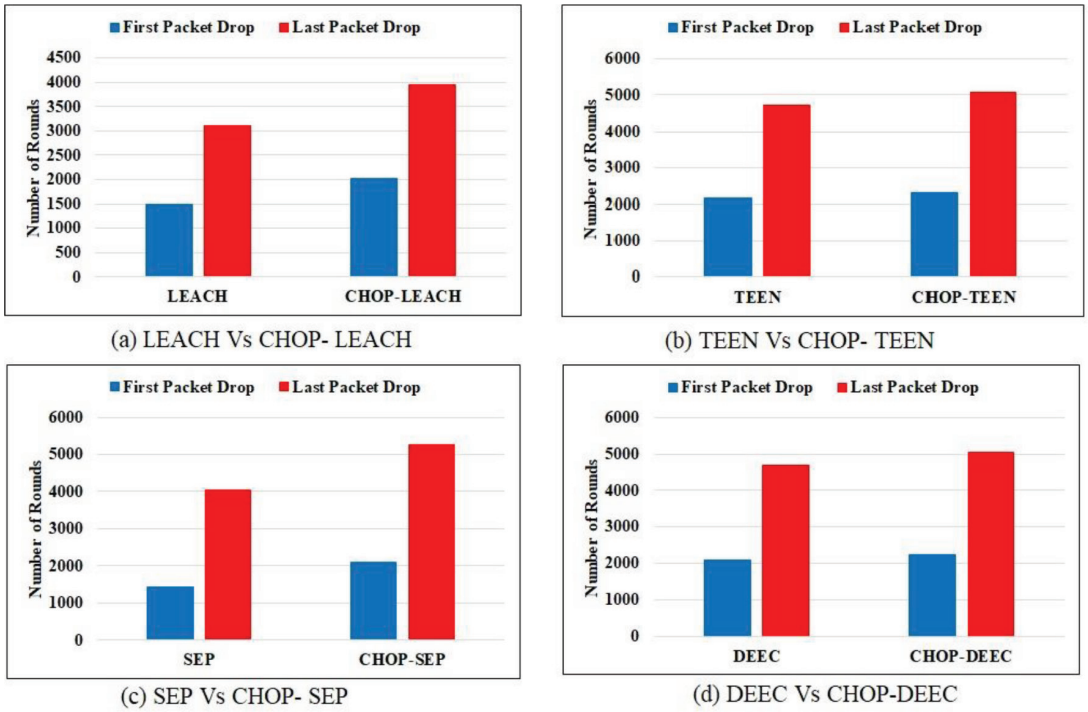

Figure 10. Simulation Performance for the first packet and the last packet drop for 150 nodes

Table 3. Simulation Performance Table

Routing Protocol	LEACH	CHOP-LEACH	TEEN	CHOP-TEEN	SEP	CHOP-SEP	DEEC	CHOP-DEEC
FPD	1499	2031	2121	2311	1420	2110	2101	2231
LPD	3099	3956	4699	5073	4051	5251	4701	5050
Packet to BS	2.6×10^4	3.4×10^4	3.1×10^4	3.4×10^4	3.2×10^4	3.9×10^4	1.4×10^5	2.2×10^5

SEP, DEEC and an enhanced lifetime of the network, better stability period, and more data packets successfully to the base station. In the future fuzzy logic, the technique will use for the selection of cluster head. The approach maximizes the coverage area using the optimal number of nodes and efficient energy utilization in the network. The identification and prevention of energy holes are important research problems, will deal with this problem in the future. We will perform an experimental test for the mobile and static sensor networks. In the future, we will also plan to develop techniques for recovering holes with limited mobility of nodes.

References

Al-Fuhaidi, B. et al. (2020) «An Efficient Deployment Model for Maximizing Coverage of Heterogeneous Wireless Sensor Network Based on Harmony Search Algorithm», *Journal of Sensors*, 2020.

Al-Shalabi, M., Ababneh, J. and Abdulraheem, W. (2021) «A Novel Adjacent Sensors-Based Mechanism to Increase Performance of Wireless Sensor Networks», *International Journal of Antennas and Propagation*, 2021.

Aliouane, L. and Benchaïba, M. (2016) «Efficient boundary detection of coverage hole in WSNs», in *2016 International Symposium on Networks, Computers and Communications (ISNCC)*, pp. 1–6.

Aliouat, Z. and Harous, S. (2012) «An efficient clustering protocol increasing wireless sensor networks life time», in *2012 International Conference on Innovations in Information Technology (IIT)*, pp. 194–199.

Behera, T. M., Samal, U. C. and Mohapatra, S. K. (2018) «Energy-efficient modified LEACH protocol for IoT application», *IET Wireless Sensor Systems*, 8(5), pp. 223–228.

Cai, S. et al. (2019) «Data collection in underwater sensor networks based on mobile edge computing», *IEEE Access*, 7, pp. 65357–65367.

Chaturvedi, P. and Daniel, A. K. (2017) «A novel sleep/wake protocol for target coverage based on trust evaluation for a clustered wireless sensor network», *International Journal of Mobile Network Design and Innovation*, 7(3–4), pp. 199–209.

Chen, Y., Xu, X. and Wang, Y. (2019) «Wireless sensor network energy efficient coverage method based on intelligent optimization algorithm», *Discrete & Continuous Dynamical Systems-S*, 12(4\&5), p. 887.

Dwivedi, A. K., Sharma, A. K. and Mehra, P. S. (2020) «Energy efficient sensor node deployment scheme for two stage routing protocol of wireless sensor networks assisted iot», *ECTI Transactions on Electrical Engineering, Electronics, and Communications*, 18(2), pp. 158–169.

Farhat, A. et al. (2018) «On the coverage effects in wireless sensor networks based prognostic and health management», *International Journal of Sensor Networks*, 28(2), pp. 125–138.

Gupta, G. P. and Jha, S. (2019) «Biogeography-based optimization scheme for solving the coverage and connected node placement problem for wireless sensor networks», *Wireless Networks*, 25(6), pp. 3167–3177.

El Handri, K. and Idrissi, A. (2020) «Parallelization of Topk Algorithm Through a New Hybrid Recommendation System for Big Data in Spark Cloud Computing Framework», *IEEE Systems Journal*.

Harizan, S. and Kuila, P. (2019) «Coverage and connectivity aware energy efficient scheduling in target based wireless sensor networks: An improved genetic algorithm based approach», *Wireless Networks*, 25(4), pp. 1995–2011.

Khan, F. and Nguang, S. K. (2021) «Dual sensing scheduling algorithm for WSN based road network surveillance», *AIMS Electronics and Electrical Engineering*, 5(1), pp. 38–54.

Khan, M. N. et al. (2020) «Improving energy efficiency with content-based adaptive and dynamic scheduling in wireless sensor networks», *IEEE Access*, 8, pp. 176495–176520.

Luo, J. and Fu, L. (2017) «A smartphone indoor localization algorithm based on WLAN location fingerprinting with feature extraction and clustering», *Sensors*, 17(6), p. 1339.

Mittal, N., Singh, U. and Sohi, B. S. (2017) «Harmony Search Algorithm Based Threshold-sensitive Energy-Efficient Clustering Protocols for WSNs.», *Adhoc & Sensor Wireless Networks*, 36.

Mohamed, M. A., Khalil, A. T. and Hammad, A. (2018) «An energy-efficient wireless sensor network routing protocol powered by ambient energy harvesting», *International Journal of Wireless and Mobile Computing*, 14(4), pp. 378–386. doi: 10.1504/IJWMC.2018.093861.

Nandagopal, C. and Ramesh, S. M. (2020) «An Efficient Data Gathering Technique Using Optimal Minimum Coverage Spanning Tree Algorithm in WSN», *Journal of Circuits, Systems and Computers*, 29(14), p. 2050225.

Narayan, V. and Daniel, A. K. (2019) «Novel protocol for detection and optimization of overlapping coverage in wireless sensor networks», *Int. J. Eng. Adv. Technol*, 8.

Narayan, V. and Daniel, A. K. (2021) «A Novel Approach for Cluster Head Selection using Trust Function in WSN», *Scalable Computing: Practice and Experience*, 22(1), pp. 1–13.

Narayan, V., Daniel, A. K. and Rai, A. K. (2020) «Energy Efficient Two Tier Cluster Based Protocol for Wireless Sensor Network», in *2020 International Conference on Electrical and Electronics Engineering (ICE3)*, pp. 574–579.

Nayak, P. et al. (2021) «Routing in wireless sensor networks using machine learning techniques: Challenges and opportunities», *Measurement*, 178, p. 108974.

Nehra, V., Sharma, A. K. and Tripathi, R. K. (2020) «I-DEEC: improved DEEC for blanket coverage in heterogeneous wireless sensor networks», *Journal of Ambient Intelligence and Humanized Computing*, 11(9), pp. 3687–3698.

Nithya, R. and Jegadeeshwari, P. (2019) «Design of Novel Based Sensing Model for Coverage Area Using Evolutionary Algorithm».

Peng, H. et al. (2015) «Energy-efficient and scalable clustering scheme for wireless sensor networks», in *2015 International Conference on Wireless Communications \& Signal Processing (WCSP)*, pp. 1–6.

Prusty, A. R., Sethi, S. and Nayak, A. K. (2020) «Energy aware optimized routing protocols for wireless ad hoc sensor network», in *Sensor technology: concepts, methodologies, tools, and applications*. IGI Global, pp. 1494–1521.

Rai, N. and Daruwala, R. D. (2016) «Effect of probabilistic sensing models in a deterministically deployed wireless sensor network», in *2016 IEEE Region 10 Conference (TENCON)*, pp. 1352–1355.

Rajawat, A. S., Jain, S. and Barhanpurkar, K. (2021) «Fusion protocol for improving coverage and connectivity WSNs», *IET Wireless Sensor Systems*.

Ramluckun, N. and Bassoo, V. (2018) «Energy-efficient chain-cluster based intelligent routing technique for Wireless Sensor Networks», *Applied Computing and Informatics*.

Salam, T. and Hossen, M. S. (2020) «Performance analysis on homogeneous LEACH and EAMMH protocols in wireless sensor network», *Wireless Personal Communications*, pp. 1–34.

Shi, K., Chen, H. and Lin, Y. (2015) «Probabilistic coverage based sensor scheduling for target tracking sensor networks», *Information sciences*, 292, pp. 95–110.

Singh, S. et al. (2020) «Proficient QoS-based target coverage problem in wireless sensor networks», *IEEE Access*, 8, pp. 74315–74325.

Singh, S. and Malik, A. (2017) «hetSEP: Heterogeneous SEP protocol for increasing lifetime in WSNs», *Journal of Information and Optimization Sciences*, 38(5), pp. 721–743.

So-In, C., Nguyen, T. G. and Nguyen, N. G. (2019) «An efficient coverage hole-healing algorithm for area-coverage improvements in mobile sensor networks», *Peer-to-Peer Networking and Applications*, 12(3), pp. 541–552.

Verma, M. and Sharma, S. (2018) «A greedy approach for coverage hole detection and restoration in wireless sensor networks», *Wireless Personal Communications*, 101(1), pp. 75–86.

Wang, C. et al. (2015) «A mobile data gathering framework for wireless rechargeable sensor networks with vehicle movement costs and capacity constraints», *IEEE Transactions on Computers*, 65(8), pp. 2411–2427.

Wang, Xiaodong et al. (2010) «Dynamic duty cycle control for end-to-end delay guarantees in wireless sensor networks», in *2010 IEEE 18th International Workshop on Quality of Service (IWQoS)*, pp. 1–9.

Xu, Y., Jiao, W. and Tian, M. (2020) «Energy-efficient connected-coverage scheme in wireless sensor networks», *Sensors*, 20(21), p. 6127.

Zimmerling, M., Mottola, L. and Santini, S. (2020) «Synchronous transmissions in low-power wireless: A survey of communication protocols and network services», *ACM Computing Surveys (CSUR)*, 53(6), pp. 1–39.

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal
Regular Issue, Vol. 11 N. 2 (2022), 179-189
eISSN: 2255-2863
DOI: https://doi.org/10.14201/adcaij.27533

Tıme-Windowed Vehicle Routing Problem: Tabu Search Algorıthm Approach

Hasibe Berfu Demir[a], Ebru Pekel Özmen[b] and Sakir Esnaf[c]

[a] Department of Industrial Engineering, Beykent University, Istanbul, Turkey
[b] Department of Industrial Engineering, Samsun University, Samsun, Turkey
[c] Department of Industrial Engineering, Istanbul-Cerrahpasa University, Istanbul, Turkey
pkl.ebru@gmail.com

KEYWORDS

vehicle routing problem; tabu search optimization; time window

ABSTRACT

Vehicle routing problem (VRP); it is defined as the problem of planning the best distribution or collection routes of the vehicles assigned to serve the scattered centers from one or more warehouses in order to meet the demands of the customers. Vehicle routing problem has been a kind of problem in which various studies have been done in recent years. Many vehicle routing problems include scheduling visits to customers who are available during certain time windows. These problems are known as vehicle routing problems with time windows (VRPTWs). In this study, a tabu search optimization is proposed for the solution of time window vehicle routing problem (VRPTWs). The results were compared with the current situation and the results were interpreted.

1. Introduction

Intense competition, products with short life curve, and increasing expectations of customers in today's global market have forced producers to invest in and give due consideration to distribution systems. This situation, with changes in communication and transportation technologies, has brought about the continuous development of logistic management (Bramel and Simchi-Levi, 1997). Logistic management is a crucial component of supply chain management. Solutions for daily encountered issues, regarding distribution, collection, or transportation of various goods, are searched for within the scope of logistic management. The main objective in logistic management is to minimize costs, and to fulfil needs expeditiously. The diversity and complexity of this field is reflected via the abundance of study areas, various methods and softwares (Crainic and Laporte, 1998).

Vehicle routing problems are placed in the center of distribution management. Every day, thousands of enterprises and organizations are to take care of the collection and delivery of various products, or

transportation of people from one place to another. As the constraints and purposes encountered in practice are quite flexible and distinctive, conditions within each business are different (Cordeau et al., 2007). To withstand the competition, decision makers in this field are to further rely on huge amounts of data, complex mathematical models and optimization techniques, as much as powerful computer and communication technologies.

2. Literature Review

Braysy and Gendreau (2002) analyze researches on Tabu Search methods for vehicle routing problem with time windows (VRPTW). VRPTW can be defined as the issue of designing minimum-cost routes for a vehicle fleet, from the main depot to spots geographically spread. Routes should be designed in such a manner that each spot will be visited only once, strictly by one vehicle, in a certain time period; and all routes begin and end at the depot. Moreover, the total number of demands of all the spots in a certain route should not exceed the vehicle's capacity. Apart from the designation of each method's basic characteristics, also presented and analyzed in the study are experimental results for Solomon's test problems. In their study, Ho and Haugland (2004) discuss vehicle routing problem with time windows and split delivery (VRPTWSD). They recommend a solving method based on tabu search for the problem which is taken without imposing constraints on split delivery options. Furthermore, analysis results of the practice for the discussed problem, carried out with 100 customers, are expressed clearly. In their study, Moccia and others (2012) explain a gradual tabu search method for generalized vehicle routing problems with time windows. The objective in this study is to provide a general tool that can be successfully applied to a wide range of specific problems. The algorithm is based on a tabu search heuristic method previously developed through the changing of neighborhood structure. In the study, experimental results are presented, and effectiveness of the approach is shown. Nguyen and others (2013) recommend a tabu search meta-intuitive method for multi-site vehicle routing problem with time windows. With the strategy which establishes control over two types of neighborhood relation which are equaled to two decision sets in certain stages of the problem, necessary tools for solving the problem are provided. Extensive numerical experiments, and comparisons with the literature, in the study, indicate that the suggested tabu search yields very high quality of solutions and improves those having been published recently. In their study, Gmira and others (2021) recommend a solution approach for vehicle routing problem with time windows in which travel speed is in relation with road sections in road network. This solution approach includes a tabu search intuitive method that focuses on shortest paths between two customers at different times of the day. The biggest contribution of this study is the development of techniques to evaluate in fixed time the applicability and approximate cost of the solution; and this way, the solution approach is enabled to take care of problematic examples possessing 200 nodes and 580 hops, in very acceptable calculation periods. The performance of the algorithm is evaluated through comparison with a full method in a series of comparative examples. Results indicate that high quality solutions are produced.

3. Methodology

3.1. Problem Definition

Vehicle routing problem can be defined as the distribution of product with different quantities to *n* number of customers by vehicles initially placed in the depot, and the collection of product from

customers. Designation of an optimum route to be used by vehicle-side while attempting to provide a group of users with service, presents a VRP problem. The purpose of the problem is to minimize the total cost of transportation. Solving the classical VRP problem includes determining a series of paths which start and finish at the depot, and which provide the constraint of giving service to each customer once. Classical vehicle routing problem (VRP) aims to find suitable routes with minimum costs (designating the shortest route, minimizing the number of vehicles, etc.). Some formulations also offer the shortening of maximum travel time (Belfiore et al., 2008). Vehicle routing problems are expressed as optimization problems occuring in the course of product distribution, the final stage of supply chain. And the approach taken into consideration to acquire the best solution, is the generating of constraints suitable for problem varieties. Some of the problem types discussed in VRP. Vehicle routing problem with time windows (VRPTW) is an improved version of classical vehicle routing problem with the addition of [a,b] time period constraint (defined as time window) to each node. In a time window, (ai) states the earliest start time of service, and (bi) states the latest start time of service.

3.2. Problem constraints

The problem is modeled under the following assumptions:

- Routes must start and end at the warehouse within the time window of the warehouse.
- Each customer should be served with a single vehicle at a time.
- The total demand of customers on the same route should not exceed the vehicle capacity.
- Customers should serve within the time window. Early arriving vehicles must wait for the customer's ready time.

Sets

$C\{1,2,\ldots,n\}$: Customer set

$N\{0,1,2,\ldots,n\}$: Node set (0. node represent warehouse.)

$D\{0,1,\ldots V\}$: Days set

Decision Variables

t_i : arrival time to customer i

w_i : waiting time at customer i

s_i : service time of customer i

x_{ijd} : a binary variable whether the vehicle travels from point i to j on day d (1 if the vehicle is going from i customer to j customer on day d, 0 otherwise)

Parameters

d_{ij} : distance between customer i and customer j

Q : vehicle capacity

e_j : start time of time window of customer j

l_j : expiry time of time windowof customer j

l_0 : end time of the time window of the warehouse (maximum travel time of the vehicle)

m_i : demand quantity of customer i

$$\sum_{i=0}^{n}\sum_{j=0}^{n}\sum_{d=0}^{n} d_{ij} * x_{ijd} \tag{1}$$

$$\sum_{d=1}^{V}\sum_{i=0,i\neq j}^{n} x_{ijd} = 1 \tag{2}$$

$$\sum_{d=1}^{V}\sum_{i=1}^{n} x_{i0d} = \sum_{d=1}^{V}\sum_{j=1}^{n} x_{0j} = V \tag{3}$$

$$\sum_{i=1}^{n}\sum_{j=0,j\neq i}^{n} m_i * x_{ijd} \leq Q, \forall d \in V \tag{4}$$

$$\sum_{i=0}^{n}\sum_{j=0,j\neq i}^{n} x_{ijd}(d_i + s_i + w_i) \leq l_o, \forall d \in V \tag{5}$$

$$w_i = max\{e_i - t_i, 0\} \tag{6}$$

$$e_j \leq t_j + w_j \leq l_j, \forall\, d \in V \tag{7}$$

$$x_{ijd} \in \{0,1\} \tag{8}$$

$$t_j \geq 0 \tag{9}$$

$$w_i \geq 0 \tag{10}$$

Equation 1 is the objective function and aims to minimize the sum of distances. Equation 2 ensures that vehicle is assigned to each customer. Equation 3 allows that the number of vehicles leaving and entering the warehouse is equal to one for each day. In Equation 4, the total of customer demands on the route assigned for each day should not exceed the vehicle capacity. Equation 5 states that all time spent along the route should not exceed the vehicle's maximum travel time. Equation 6 shows the waiting time calculation if the vehicle arrives before the customer is ready. Equation 7 is the time window constraint and Equation 8-10 specifies the value sets for variables.

3.3. Tabu Search Algorithm

Tabu search (TA) is a metaheuristic algorithm based on problem solving and includes the concept of 'memory'. The basic approach is to ban or punish repetition in the next cycle to prevent the step leading to the final solution from making circular motions. Thus, the Tabu Search algorithm guides the

regional-heuristic research to search for solutions that are beyond the regional optimum by examining new solutions. Tabu search algorithms can be used for the problems of scheduling, investment planning, communication, design, routing, etc. (Glover and Laguna, 1998). The basic principle of TA is to create a tabu list to prevent a previous move from reversing rather than repeating it. Eq. (3.1) can be used to explain the TA:

$$Minc\ (x),\ x\ \in\ X \tag{3.1}$$

Here each x element represents a motion, and the entire set of motions is denoted by X. The x vectors are used as the TA memory structure. Thus, depending on the memory value kept in the vector, some movements will be considered as taboo in the search for solutions, and some will focus more on others. Each movement in the vector X represents the choice of a neighbor of the current solution.

Some concepts and steps of TA method are explained in detail below (Jayaswal, 2000):

Creating a Solution: Initial solution pairs were determined randomly.

Movement Mechanism: The changes made in the solution provide a new solution. There is a stopover from the current point to the visit of all points. This event constitutes the neighbors of the current solution.

Search Neighbor: The most crucial point provided by the neighbor search concept is that it provides the best motion selection. Neighborhood is chosen by the best value chosen in each iteration. Besides, let the neighborhood of x to be $N(x)$, it could be expressed as $N^*(x) \subset N(x)$.

Aspiration criteria: This criterion is the key factor in defining the degree of applicability of the method and allows for the action that provides better results (Salhi, 2002). If $f(xtabu) < f(xbest)$, then current movement will be invalid.

Memory: It memorizes prohibited actions and adds them to the tabu list.

Taboo List: Tabu list adds solutions that have been visited before to avoid duplication. The size of the tabu list is very important.

Stopping Criteria: Reaching maximum iteration number or reaching the desired solution, etc. situations provide stop condition for tabu search.

The steps of the TA algorithm are simply shown below.

Step 1: Creating the initial solution
Step 2: Creating a candidate solution list
Step 3: Evaluate the solutions
Step 4: Choose the best solution
Step 5: Stop criteria satisfied? If non-satisfied, go to Step 2
Step 6: Save the last solution

4. Computational Results

4.1. Data Sets

The data set is obtained from http://vrp.galgos.inf.puc-rio.br/index.php/en/ called as Benchmark: Set A. The pseudo code of the proposed TA model was given in Javascript in detail. The processor model of the computer used is Intel® XEON® CPU E5-2660 V2. It also has 2.2 GHz processor base

Algorithm // pseudocode

```
function TabuAlgo()
{
   try
   {
      UnVisitedNodeCount=GetCountUnvisitedNode();/*Returning Unvisited node counts. Its 38 for begining*/
            while(ToNumber(UnVisitedNodeCount)!=0)/*This loop will continue until Count 0*/
          {
            for(i=1;i<39;i++)
            {
               ChechVisitedStation(InputsChechVisit, OutputsVisit); /*Checking node visited status. If node was
viseted return Y*/
               VisitedFlag=OutputsVisit.GetProperty(«VisitFlag»);
               if(VisitedFlag==N){
                  InputsCoorditanes.SetProperty(«Node», i);
                  GetNodeCoorditanes(InputsCoorditanes, OutputsCoorditanes);/*Returning Coordinate Values for
nodes*/
                  GetDistance(InputsDistance, OutputsDistance);/*Calculated distance*/
                  DistanceValue = OutputsDistance.GetProperty(«DistanceValue»);
                  if(ToNumber(BestDistance)>ToNumber(DistanceValue)){
                        BestDistance=DistanceValue;
                        BestSolution=i;
                        Node=i;/*Best Solution*/
                        RootX1=x2;
                        RootY1=y2;
                     }
               }
            }
            GetTimeAndOrderFromNode(InputsValidation,OutputsValidation);/*Calculating Time and Order
values for this root node*/
            TotalTime=ToNumber(Time)+ToNumber(TotalTime);
            TotalOrder=ToNumber(Order)+ToNumber(TotalOrder);
            if(ToNumber(ValidTime)>ToNumber(TotalTime) && ToNumber(ValidOrder)>ToNumber(TotalOrder))
            {
               AddBestPathNode(Node, OutputsBestPath);/*Added Path and flagged as visited*/
               Station=Station+1; /*Rised Station values for same day*/
               x1=RootX1;/*New root Node X1 updated*/
               y1=RootY1;/*New root Node Y1 updated*/
               DistanceValue=0;
               BestDistance=0;
               InputsList.SetProperty(«Node», Node);
               UpdateVisitedList(InputsList, OutputsList);    /*Update visited Node list*/
               UnVisitedNodeCount=ToNumber(UnVisitedNodeCount)-1;
            }
            else {/*Return Store and begining values also Day Values rised*/
               x1=39;
               y1=19;
               TotalTime=0;
               Time=0;
               TotalOrder=0;
               Order=0;
               Station=1;
               Day=Day+1;

            }
         }
   }
```

Figure 1. Pseudocode of the proposed algorithm

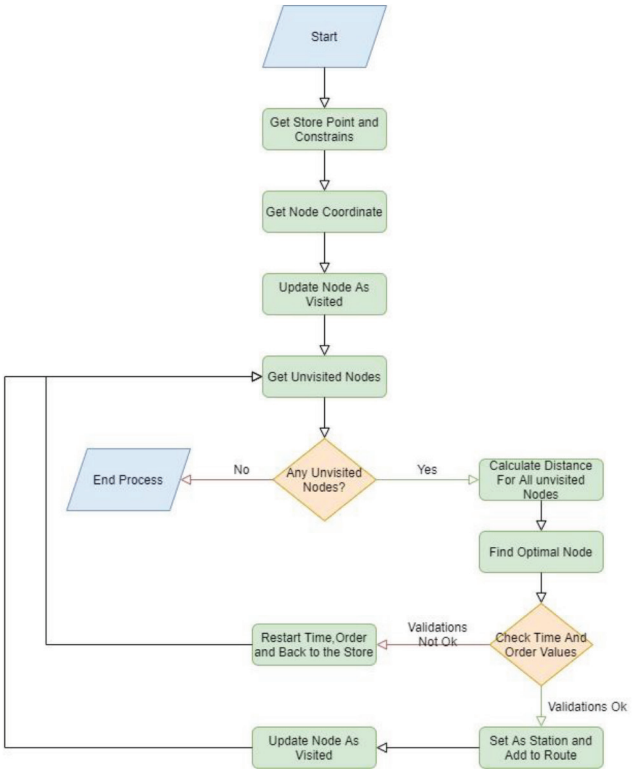

Figure 2. Process Flow Chart

frequency. The pseudo code of the TA which is employed to solve the routing problem is given in Figure 1. At the same, the progress of the algorithm is given as flow chart in Figure 2. Some performance criteria are used to analyze mathematical models. In this study, average solution time was chosen as a performance criterion.

Customer coordinates and demands are presented in Table 1 and Table 2.

Table 1. Coordinates of Customer

	X	**Y**		**X**	**Y**
1	39	19	**21**	55	43
2	79	19	**22**	83	29
3	41	79	**23**	93	49
4	25	31	**24**	87	23
5	63	93	**25**	31	23
6	33	5	**26**	19	97

	X	Y		X	Y
7	69	17	**27**	41	9
8	57	73	**28**	83	61
9	53	75	**29**	9	7
10	1	1	**30**	13	13
11	79	73	**31**	43	37
12	59	5	**32**	13	61
13	1	37	**33**	71	51
14	41	31	**34**	45	93
15	23	73	**35**	93	55
16	37	27	**36**	5	97
17	85	93	**37**	81	11
18	93	13	**38**	7	53
19	85	45	**39**	7	41
20	49	91			

Table 2. Customer Demand

Warehouse	0	**20**	7
1	18	**21**	11
2	16	**22**	11
3	22	**23**	1
4	24	**24**	22
5	3	**25**	16
6	19	**26**	15
7	6	**27**	7
8	6	**28**	5
9	6	**29**	22
10	12	**30**	9
11	18	**31**	10
12	16	**32**	11
13	72	**33**	9
14	7	**34**	3
15	16	**35**	7
16	23	**36**	15
17	4	**37**	10
18	22	**38**	2
19	23		

The calculation of the cost was made as the sum of the distances on the route assigned to the vehicle. Euclidean distance is used in calculating the distance as shown in Eq.11.

$$d(A, B) = \sqrt{\quad} \tag{11}$$

The results of the algorithm has summarized in the tables below. Table 3 shows on which day the vehicle will go to which points. For example, on day 1, the vehicle for customer number 16 should leave from the warehouse. The demand for this visit is seen as 16 (See Table 4). Again, on the 2nd day, a demand for 22 units must be delivered to the customer number 25 from the warehouse. The costs for all warehouses are shown in Table 5.

Table 3. Customers to be visited day by day

Assignments	**1st day**	**2nd day**	**3rd day**	**4th day**	**5th day**	**6th day**
I	16	25	27	21	2	20
II	14	4	6	33	37	34
III	31	39	30	19	18	5
IV		13	29	23	24	26
V		38	10	35	22	36
VI		32	12	28	8	
VII		15	7	11	9	

Table 4. Demands of the customers

Demand Section (unit)	**1st day**	**2nd day**	**3rd day**	**4th day**	**5th day**	**6th day**
I	16	22	16	7	18	23
II	72	22	3	11	15	9
III	9	2	22	22	4	24
IV		16	5	11	1	16
V		10	6	3	11	7
VI		10	18	7	6	
VII		7	19	12	6	
VIII				23	16	
Capacity Utilization (%)	**97**	**89**	**89**	**96**	**77**	**79**

Table 5. Total cost by day (distances)

Warehouse	1st day	2nd day	3rd day	4th day	5th day	6th day
I	8,246	8,944	10,198	28,844	40	72,691
II	5,656	10	8,944	17,888	8,246	4,4721
III	6,324	20,591	21,540	15,231	12,165	18
IV		7,211	7,2111	8,944	11,661	44,181
V		17,088	10	6	7,211	14
VI		10	58,137	11,661	51,107	
VII		15,620	15,620	12,649	4,472	
VIII				20,880	12,649	
Total Cost	20,22762	89,45514	131,6523	122,1004	147,5137	153,3447

The assigned route for first day is visualized in Figure 3. As explained in the definition of the problem, each route starts with the warehouse and ends with the warehouse. Considering that the maximum capacity of the vehicles is 100 units, it can be said that the vehicle capacity is 97% on the 1st day. The total cost covering all days was found as 664.29 units of road, end of the analysis for the vehicle routing problem. The total solution time of the problem was calculated as 57 seconds.

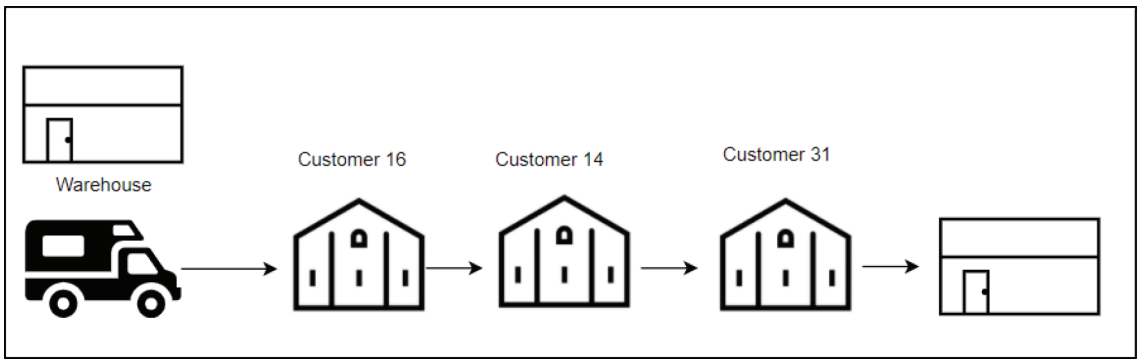

Figure 3. Assigned routes on day 1

5. Results and Discussions

This study proposed a tabu search heuristic for the VRPTWs. The heuristic minimizes total distance traveled using tabu search algorithm. The application of tabu search algorithm, one of the metaheuristic methods frequently encountered in the international literature, to VRPTWs has not been adequately studied in the literature. In order to eliminate this deficiency, information was given about the time windowed vehicle routing problem in the study. In addition, the algorithm obtained the best known solution on every instance of a set of 38 instances of the VRPTWs.

In future work, we plan to solve our tabu search algorithm to the VRPTW problem with Phyton. Additionally, we programe to compare our method with other metaheuristic algorithms such as genetic

algorithm. This study is assumed that the results of this study will provide various benefits to the practice and the literature.

6. References

Belfiore, P. Tsugunobu, H. and Yoshizaki, Y. (2008). *Scatter search for Vehicle Routing Problem with Time Windows and Split Deliveries. Vehicle Routing Problem*, Book edited by: Tonci Caric and Hrvoje Gold, ISBN 978-953-7619-09-1, pp. 142.

Bramel, J. and Simchi-Levi, D. (1997). *The Logic of Logistics: Theory, Algorithms and Applications for Logistics Management.*

Braysy, O. and Gendreau, M. (2002). Tabu Search Heuristics for The Vehicle Routing Problem with Time Windows, *Sociedad de Estadística e Investigación Operativa,* 10 (2), 211-237.

Cordeau, J.F., Laporte, G., Savelsbergh, M.W.P. and Vigo, D. (2007). Vehicle Routing Transportation, *Handbooks in Operations Research and Management Science*, 14, 367–428.

Crainic, T.G. and Laporte, G. (1998). Fleet Management and Logistics, Springer.

Glover, F., Laguna, M. (1998). Tabu Search. In: Du, DZ., Pardalos, P.M. (eds) Handbook of Combinatorial Optimization. Springer, Boston, MA. https://doi.org/10.1007/978-1-4613-0303-9_33

Gmira, M., Gendreau, M., Lodi, A. and Potvin, J.Y. (2021). Tabu Search for The Time Dependent Vehicle Routing Problem with Time Windows on A Road Network, European Journal of Operational Research, 1(1), 129-140.

Ho, S. and Haugland, D. (2004). A Tabu Search Heuristic for The Vehicle Routing p-Problem with Time Windows and Split Deliveries, Computers & Operations Research, 32, 1947–1964.

Jayaswal, S. (2000). A Comparative Study of Tabu Search and Simulated Annealing for Travelling Salesman Problem.

Moccia, L., Cordeau, J.F. and Laporte, G. (2012). An Incremental Tabu Search Heuristic for The Generalized Vehicle Routing Problem with Time Windows, *Journal of the Operational Research Society,* 63(2), 232-244.

Nguyen, P.K., Crainic, T.G. and Toulouse, M. (2013). Tabu Search for The Time Multi-zone Multi-trip Vehicle Routing Problem with Time Windows, *European Journal of Operational Research,* 231, 43-56.

Salhi, S. (2002). Defining Tabu List Size and Aspiration Criterion Within Tabu Search Methods, *Computers & Operations Research*, 29(1), 67-68.

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal
Regular Issue, Vol. 11 N. 2 (2022), 191-206
eISSN: 2255-2863
DOI: https://doi.org/10.14201/adcaij.27393

Deep Learning Approach to Technician Routing and Scheduling Problem

Engin Pekel

Hitit University, Turkey
enginpekel@hitit.edu.tr

KEYWORDS

Adam algorithm; deep learning; optimization; technician routing and scheduling

ABSTRACT

This paper proposes a hybrid algorithm including the Adam algorithm and body change operator (BCO). Feasible solutions to technician routing and scheduling problems (TRSP) are investigated by performing deep learning based on the Adam algorithm and the hybridization of Adam-BCO. TRSP is a problem where all tasks are routed, and technicians are scheduled. In the deep learning method based on the Adam algorithm and Adam-BCO algorithm, the weights of the network are updated, and these weights are evaluated as Greedy approach, and routing and scheduling are performed. The performance of the Adam-BCO algorithm is experimentally compared with the Adam and BCO algorithm by solving the TRSP on the instances developed from the literature. The numerical results evidence that Adam-BCO offers faster and better solutions considering Adam and BCO algorithm. The average solution time increases from 0.14 minutes to 4.03 minutes, but in return, Gap decreases from 9.99% to 5.71%. The hybridization of both algorithms through deep learning provides an effective and feasible solution, as evidenced by the results.

1. Introduction

Optimized systems are one of the keys to success in today's competitive world. This situation is encountered in almost every field. The technician routing and scheduling problem (TRSP) is one of these areas. TRSP can be considered a special case of vehicle routing problems (Pekel and Kara, 2019). TRSP does not only consist of assigning technicians to teams but also of assigning teams to tasks and creating routes. In addition, the selected technician group must meet a certain level of qualification (Pekel, 2020; Zamorano and Stolletz, 2017). In many technician planning problems, routing complexity is crucial. Routing is carried out so that the expert technician group leaves and returns to

the central warehouse at the end of the day. Travelling times between each node are computed as Euclidean distance (Khalfay et al., 2017).

The TRSP includes a pre-determined set of technicians who have various talents and a set of tasks requested by the customer at various locations. To perform tasks consisting of customer requests, the set of technicians is divided into teams, and teams travelling in different locations are assigned the tasks. When teams are assigned tasks, the constraint of compatibility between tasks requiring different skills and teams of technicians with different skills is considered. Time windows are defined for the tasks requested by the customer, and requests must be fulfilled during these time intervals. Besides, technicians perform tasks for a specific period, and overtime occurs when this time interval is exceeded. Once and for all, the service times of the tasks, travel times, and the skill requirements required for each task are known in advance.

Algorithms have been developed to provide many exact solutions to TRSP so far in the literature. Charris et al. (2019) implemented a decision support system to optimise technicians' routes in Colombia. In their work, technicians performed daily tasks received from customers. Mathlouthi et al. (2018) maximized the maintenance and repair of electronic process equipment by subtracting total profit from operating costs. Anoshkina and Meisel (2019) tackled the issue of combined teams of workers and routes while considering costs. Chen et al. (2016) presented the technician orientation model that creates experience-based learning, aiming to minimize the time taken to fulfil the final task. The authors used the heuristic method of travel from record to record (RTR) in solving the model. Kovacs et al. (2012) consider technicians with several skills at several levels grouped into teams to fulfil routing decisions and upkeep tasks and establish the Service TRSP. Pekel (2020) suggested an improved PSO (IPSO) algorithm to solve TRSP using a specific dataset. Graf (2020) considered a multi-period vehicle and technician routing and scheduling problem and proposed a combination of large neighbourhood and local search heuristics and a decomposition approach to efficiently generate competitive solutions under restricted computational resources. The authors' numerical results showed that the method is efficient and effective, especially under tight time restrictions. Çakırgil et al. (2020) dealt with multi-skilled workforce scheduling and routing problem in field service operations motivated by a daily, real-life problem faced by electricity distribution companies. They proposed a two-stage matheuristic to obtain a good approximation of the Pareto frontier since the computational effort considerably increases in real-life problem instances. Mathlouthi et al. (2021) dealt with technician routing and scheduling problems motivated by an application to repair and maintain electronic transactions equipment. The problem exhibits many special features, such as multiple time windows for service, a spare parts inventory taken by each technician, and tasks that may require a special part to be performed. The authors used a methodology based on Tabu search, coupled with an adaptive memory.

Deep learning first entered our lives mainly in the form of forecasting and clustering, but nowadays, it is beginning to be used effectively in combinational optimization problems. Fu et al. (2020) investigated vehicular energy networks' stability and efficiency to optimize the routing and dynamic storage allocation of renewable energy by integrating a time-expanded topology graph and a deep learning method. Authors applied deep learning to improve the prediction accuracy of the traffic pattern. Wang and Sun (2020) proposed a multi-agent deep reinforcement learning method to develop a bus route that is dynamic and has flexible holding control strategies. The authors used a headway-based reward function to train their proposed method. Hussain et al. (2021) researched the progress of machine learning in vehicle networks for intelligent route decisions. Lee et al. (2020) proposed a route and

charging station algorithm based on a model-free deep reinforcement learning method to deal with the uncertainty of traffic conditions by minimizing the total travel time. James et al. (2019) proposed a novel deep reinforcement learning-based neural combinatorial optimization strategy. The authors used a deep reinforcement learning mechanism with an unsupervised auxiliary network to train the model's parameters. Koh et al. (2020) proposed a deep reinforcement learning method to construct a real-time intelligent vehicle routing and navigation system. Also, the authors used intelligent agents to facilitate intelligent vehicle navigation. Hernández-Jiménez et al. (2019) explored a deep learning approach to the routing problem in vehicular delay-tolerant networks by performing a routing architecture and deep neural networks.

The paper's main contributions are as follows: a combinational optimization problem is solved by performing deep learning in TRSP for the very first time. Also, the Adam algorithm is integrated with the BCO method and obtains decent and fast solutions.

The remainder of the article is organised as follows: Section 2 describes the model of TRSP. Section 3 presents the method that consists of the Adam algorithm, Body change operator (BCO) and the Adam-BCO method. Section 4 offers the numerical results of the algorithms for Pekel's benchmark instances. Finally, Section 5 outlines all the findings and draws conclusions from them.

2. Mathematical Model

This paper does not propose an algorithm to provide an exact solution, however, the mathematical model of the problem is presented. I represents one central depot and tasks. TRSP consists of $A \subseteq (i,j)\ i,j \in I$ arcs, K teams, D days, $\overline{D}$ days allowed to visit a task, I tasks, M technicians, L proficiency levels, and Q skills. A generated technician team visits from node i to node j, considering c_{ij} visiting cost and p_i service time of node i. Visiting operation must be completed within the earliest a_{id} and the latest b_{id} bid, considering daily work hours $[e,f]$. Started works cannot be interrupted. The generated technician team has δ number of technicians, and this paper chooses $\delta = 2$. The mathematical model accepts different values for δ if requested. When the number of technicians was more than 2, there was a slight increase in the time the model required to obtain the solution, and more straightforward solutions were obtained. Each task requires $v_{id} : \{0\ or\ 1\}$ proficiency, and each technician has proficiency level $g_{mg} : \{0\ or\ 1\}$. The model allows ω^{cost} cost of waiting time and ot^{cost} cost of overtime. However, this paper does not allow ω^{cost} and ot^{cost}. All sets and parameters are described above. Table 1 shows the decision variables of the model.

Table 1. Notation

Decision variables	
x_{ijkd}	1 if team k fulfills task i and visits task j on day d, 0 otherwise
y_{ikd}	1 if team k carries out task i on day d, 0 otherwise
z_{mkd}	1 if technician m works for team k on day d, 0 otherwise
S_{ikd}	Beginning time of the task i carried on team k and day d
ω_i	Staying time of task i
ot_{kd}	Overtime of team k on day d

$$\min z = \sum_{(i,j)\in A}\sum_{k\in K}\sum_{d\in D} c_{ij}x_{ijkd} + \omega^{cost}\sum_{i\in I}\omega_i + ot^{cost}\sum_{k\in K}\sum_{d\in D} ot_{kd} \quad 12\backslash* \text{ MERGEFORMAT ()}$$

$$\sum_{k\in K}\sum_{d\in D} y_{ikd} = 1\,\forall i \in I' \quad 34\backslash* \text{ MERGEFORMAT ()}$$

$$\sum_{j:(i,j)\in A_d} x_{ijkd} = y_{ikd}\,\forall i \in I', \forall k \in K, \forall d \in D \quad 56\backslash* \text{ MERGEFORMAT ()}$$

$$\sum_{j:(o,j)\in A_d}\sum_{k\in K} x_{ojkd} \geq 1\ \forall d \in D \quad 78\backslash* \text{ MERGEFORMAT ()}$$

$$\sum_{i:(i,o)\in A_d}\sum_{k\in K} x_{iokd} \geq 1\ \forall d \in D \quad 910\backslash* \text{ MERGEFORMAT ()}$$

$$\sum_{i:(i,h)\in A_d} x_{ihkd} - \sum_{j:(h,j)\in A_d} x_{hjkd} = 0\ \forall h \in I', \forall k \in K, \forall d \in D \quad 1112\backslash* \text{ MERGEFORMAT ()}$$

$$x_{ijkd}\,(S_{ikd} + C_{ij} + P_i - S_{jkd}) \leq 0\ \forall i, j : (i,j) \in A_d,\ \forall k \in K,\ \forall d \in D \quad 1314\backslash* \text{ MERGEFORMAT ()}$$

$$y_{ikd}(a_{id} - S_{ikd}) \leq 0\ \forall i \in I',\ \forall k \in K,\ \forall d \in D \quad 1516\backslash* \text{ MERGEFORMAT ()}$$

$$y_{ikd}(S_{ikd} - b_{id} - \omega_i) \leq 0\ \forall i \in I',\ \forall k \in K,\ \forall d \in D \quad 1718\backslash* \text{ MERGEFORMAT ()}$$

$$x_{ojkd}(S_{jkd} - e) \geq 0\ \forall j \in I',\ \forall k \in K,\ \forall d \in D \quad 1920\backslash* \text{ MERGEFORMAT ()}$$

$$x_{iokd}(S_{jkd} + c_{io} + p_i - f - ot_{kd}) \leq 0\ \forall i \in I',\ \forall k \in K,\ \forall d \in D \quad 2122\backslash* \text{ MERGEFORMAT ()}$$

$$\sum_{k\in K} z_{mkd} \leq 1\ \forall m \in M, \forall d \in D \quad 2324\backslash* \text{ MERGEFORMAT ()}$$

$$\sum_{m\in M} z_{mkd} = \delta\ \forall k \in K, \forall d \in D \quad 2526\backslash* \text{ MERGEFORMAT ()}$$

$$v_{iql}\,y_{ikd} \leq \sum_{m\in M} g_{mql} z_{mkd}\ \forall i \in I', \forall q \in Q, \forall l \in L, \forall k \in K, \forall d \in D \quad 2728\backslash* \text{ MERGEFORMAT ()}$$

$$0 \leq \omega_i \leq \omega^{max} \quad 2930\backslash* \text{ MERGEFORMAT ()}$$

$$0 \leq ot_{kd} \leq ot^{max} \quad 3132\backslash* \text{ MERGEFORMAT ()}$$

$$S_{ikd} \geq 0\ \forall i \in I',\ \forall k \in K,\ \forall d \in D \quad 3334\backslash* \text{ MERGEFORMAT ()}$$

$$x_{ijkd},\ y_{ikd},\ z_{mkd} \in \{0,1\}\ \forall (i,j) \in A,\ \forall m \in M,\ \forall k \in K,\ \forall d \in D \quad 3536\backslash* \text{ MERGEFORMAT ()}$$

Equation (1) is the model's objective function and minimizes the total travelling cost. ω^{cost} and ot^{cost} are equal to zero since this paper does not allow waiting time and overtime. Equations (2) and (3) guarantee that each task is visited in the planned time windows. Each generated team completes its routed tasks starting and ending from the same central depot in equations (4) and (5). At least one team must be created for each day. Equation (6) provides the flow of tasks in a team and day. Equation (7) both avoids sub-tours and calculates the starting times of tasks. Equations (7) - (9) enable the tasks to start and end according to their time windows. Equations (7) - (11) are non-linear limitations. However, constraints are linearized by a big M formulation (7) - (11). The transformation of the 7th and 8th equations into linear format with Big M is shown in equations (19) and (20). The remaining equations (9) – (11) were made linear in the same way.

$$S_{ikd} + c_{ij} + P_i - S_{jkd} \leq BigM * (1 - x_{ijkd}) \quad 3738\backslash* \text{ MERGEFORMAT ()}$$

$$a_{id} - S_{ikd} \leq BigM * (1 - y_{ikd}) \quad 3940\backslash* \text{ MERGEFORMAT ()}$$

Equations (10) and (11) enable the tasks to start and finish according to daily working hours. When starting the route for the first task from the depot, the starting time is assumed to be greater than or equal to $\max\{e,a_{id}\}$. Equation (12) impedes the inclusion of a technician in more than one team per day, and Equation (13) ensures that the number of technicians in the teams is equal to the predetermined value. Equation (14) guarantees that the skills of the chosen technicians satisfy the mastery essentials for the routed tasks. Equations (15) and (16) provide lower and upper bounds to waiting time and overtime, respectively. Equation (17) ensures that the starting time is a positive variable, and constraint (18) defines the specified variables as binary.

Figure 1 shows the illustration of the mathematical model for TRSP.

3. Methodology

The methodology section describes the three different methods applied in this study. These methods are the Adam algorithm, the BCO method, and the Adam-BCO hybrid algorithm. Also, the network structure of the algorithms applied in this section is provided.

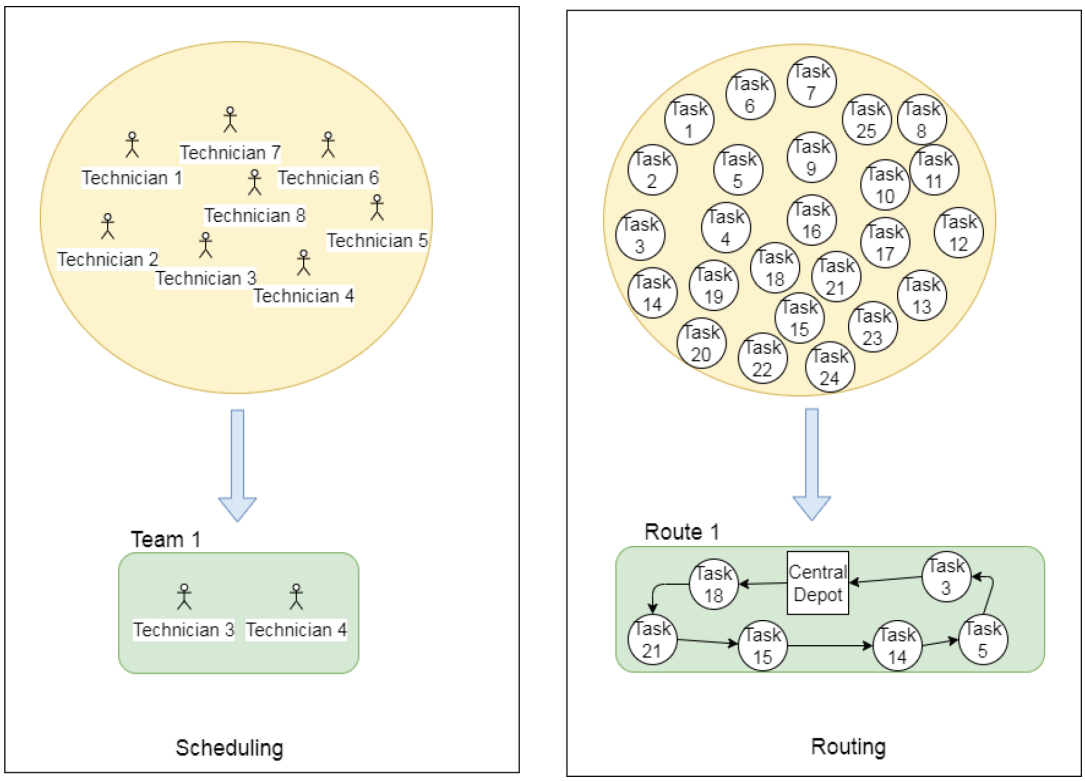

Figure 1. An illustration of the mathematical model

There are 26 hidden neurons in the first hidden layer. The main reason for choosing 26 neurons is that there are 25 customers and one central depot in the discussed TRSP. Since there are 26 nodes, 26 neuron and input entries are made. Two hidden layers are used in the network structure. Applications have been made to use more than two hidden layers, but practical results have not been obtained. The connections between the inputs and hidden neurons in the first layer are considered arc connections. Adam and Adam-BCO hybrid algorithms train arc connections, taking the maximum weight value. The maximum weight value passes through the arc connections on the second and first hidden layers, and the maximum weight is selected as the Greedy rule. If the obtained route meets the feasibility conditions, deep learning provides a new routing and the technician groups' scheduling.

Figure 2 shows the network structure of deep learning implemented in TRSP. For example, let's assume that there is an assignment from task 8 to complete task 1. Let's express this with X [8, 1] = 1. It is checked that this assignment is feasible, and according to a function value created on the basis of the main algorithm, X [8, 1] assignment value can be 0. Thus, new solution spaces are tried.

3.1 Adam Algorithm

The ADAM algorithm can be defined as the realization of an efficient stochastic optimization method with only the first-order derivative of the selected function and with minimal memory requirement. The name Adam is derived from adaptive moment estimation and is designed to combine the advantages of the two methods, AdaGrad (Duchi et al., 2011) and RMSProp (Tieleman and Hinton, 2012). The algorithm calculates adaptive learning rates from the outputs of the first and second moments of the gradients, considering the parameters it contains in its structure. The Adam algorithm is an algorithm that includes the advantageous aspects of the RMSProp and momentum methods. It caches the momentum changes as well as the learning rates of each of the parameters; it combines RMSProp and momentum. This provides higher performance in terms of speed.

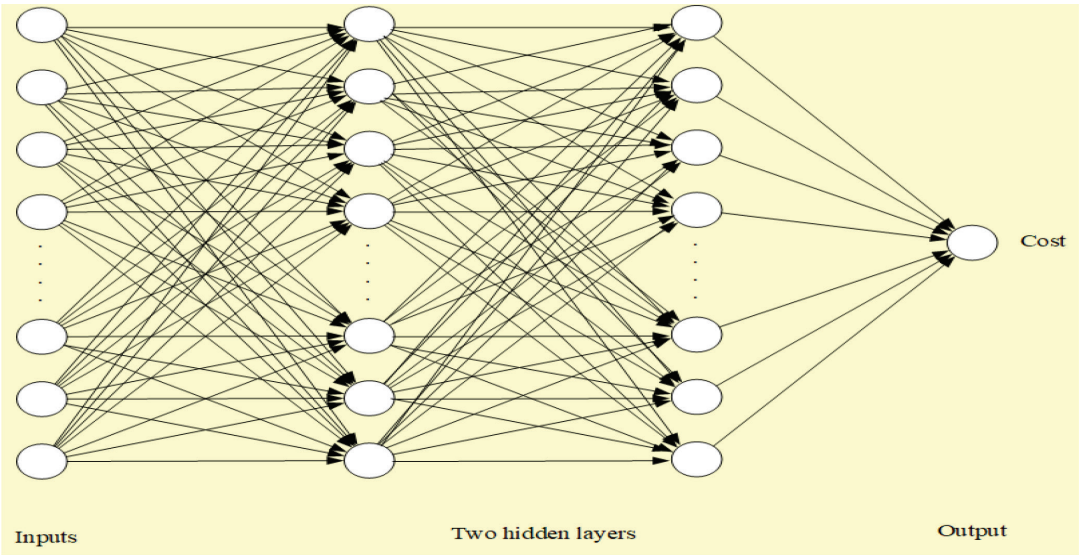

Figure 2. Deep learning network structure implemented in TRSP

The obtained learning rates are used to reach the bias-corrected version of the exponential moving averages (Kingma and Ba, 2014). With bias-corrected values, the weights or connection values of the network used for estimation are updated, and it is indented to obtain convergence over the determined iterations or convergence limit. Algorithm 1 shows the mechanism of the Adam algorithm used in TRSP. For the Adam algorithm to work, alpha and two different beta parameters must be determined.

Algorithm 1. Pseudo-code Adam algorithm

Require: α: Step size, β_1 and $\beta_2 \in$: decay rates for the moment estimates
Require: $f(\theta)$: Weight of deep-learning with parameters θ
Require: θ_0: Initial weights of deep-learning and set to 0
$m_0 \leftarrow 0$
$\vartheta_0 \leftarrow 0$
$t \leftarrow 0$ (Iteration (150))
while *currentiteration* $< t$ do
$t \leftarrow t + 1$
$g(\theta_t) = tanh\ (\theta_t^{max})$: Greedy rule
get the values of gradients $(g(\theta_t))$
update first-moment estimate (m_t)
update second-moment estimate (ϑ_t)
compute bias-corrected first-moment estimate $(\hat{m}_t)$
compute bias-corrected second-moment estimate $(\hat{\vartheta}_t)$
update weights (θ_t)
end while
return θ_t

Here, different combinations should be examined while determining their parameters. Within the scope of this study, research was conducted on different parameter values by trial-and-error method and chosen $\alpha = 0.0001$, β_1=0.8, and $\beta_2 = 0.99$. In this paper, inputs refer to actions. Visiting from a node to a different node is considered an action. There are 26 hidden neurons in the first and second hidden layers. The first and second-moment estimate values must be updated for the algorithm to update the net weights. The first-moment estimate is updated with equation $m_t = \beta_1 * m_{t-1} + (1-\beta_1) * g_t$ and the second-moment estimate is updated with equation $\vartheta_t = \beta_2 * \vartheta_{t-1} + (1-\beta_2) * g_t^2$. g_t is the gradient of the *tanh* function and computed by $g_t \leftarrow \nabla_\theta\ tanh_t\ (\theta_{t-1})$. Bias is corrected first, and second-moment estimates are computed by $\hat{m}_t = \frac{m_t}{(1-\beta_1^t)}$ and $\hat{\vartheta}_t = \frac{\vartheta_t}{(1-\beta_1^t)}$, respectively. After updating all the necessary parameters, the weights of the net on the first hidden layer are calculated by $\frac{\alpha * \hat{m}_t}{(\sqrt{\hat{\vartheta}_t} + \varepsilon)}$. Here, $\varepsilon = 10^{-8}$.

As a result of the calculations, the values of the networks between actions and neurons in the first hidden layer, are obtained. The maximum θ_t value coming to any hidden neuron in the first hidden layer from the twenty-six inputs, is taken. All the steps described earlier are repeated by taking the θ_t^{max} value. The network weights (θ_t) are updated, from the hidden neurons in the second hidden layer to the output neuron. Updated weights allow new routing to be created on the basis of the Greedy rule. While using a Greedy rule-based action selection mechanism, the routing and technician selections to be created should be feasible. To this end, environment simulation is applied while choosing an action.

Figure 3 shows the operation mechanism of the Adam algorithm used to find possible routings and scheduling. Feasible routing and scheduling are done with random data generation. The data created here reveals actions (from one node to the other). (26×26) actions are expressed as the weight value of the

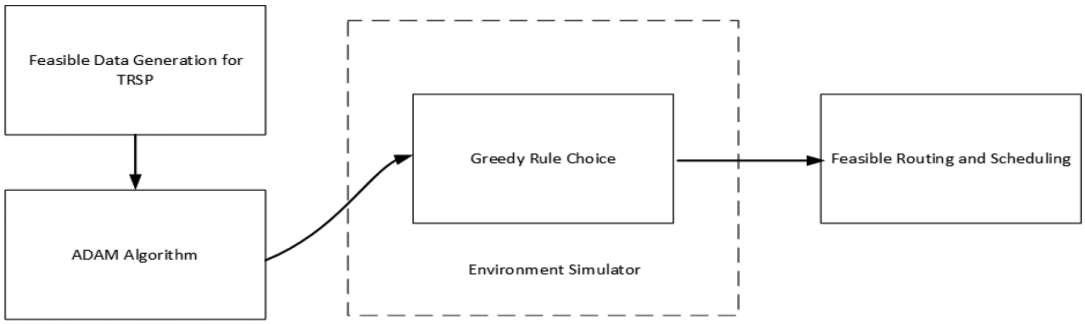

Figure 3. Operation mechanism of Adam algorithm in TRSP

network. The weights of the network are updated with the Adam algorithm. Later the update process, the routing, and scheduling process is completed by simulating the environment to verify feasibility.

3.2 Body Change Operator

The genetic algorithm is a stochastic solution generation algorithm inspired by the evolutionary process of living things. The development and characteristics of living things are found on chromosomes. The genetic algorithm uses a simple chromosome-like data structure to follow the characteristics of the solution through this structure. The replacement of these structures is provided by mechanisms such as crossing. The variety of problems in which genetic algorithms are applied is quite wide, and genetic algorithms are often seen as the optimizers of the obtained solutions (Whitley, 1994).

The body change operator (BCO) is a solution improvement operator based on a genetic algorithm (Pekel and Kara, 2019). The part where it differs from the genetic algorithm is the absence of mutation. Cross-over operation is performed by crossing sub-routing in a feasible routing, but there is no cross-over ratio. In short, this operator, which makes an improvement process for each solution, works. Algorithm 2 shows the mechanism of the body change operator used in TRSP.

Algorithm 2. Pseudocode of BCO

Initialize: Each chromosome (n: Population size (300))
    $ch_p \leftarrow 0$ (Chromosome position)
    $ch_c \leftarrow 0$ (Chromosome cost)
    $Gch_p \leftarrow 0$ (Chromosome global position)
    $Gch_c \leftarrow 0$ (Chromosome global cost)
    $t \leftarrow 0$ (iteration (1))
    while *currentiteration*$<t$ do
        perform: Body change operator
        compute: Cost function
        decide: $ch_c < Gch_c$
            $Gch_p \leftarrow ch_p$
            $Gch_c \leftarrow ch_c$
    end while
    return

The body change operator is an operator that can make very effective improvements, but it must be effectively directed to search different solution spaces. As seen in Algorithm 2, a single iteration is run, and the obtained solutions are shared in the results section. The integration of the Adam algorithm with the body change operator enables the search for different solution spaces and offers the Adam algorithm different learning pathways.

Figure 4 shows the operation mechanism of the BCO algorithm. One feasible solution includes multiple routes, where technicians scheduled each route. The choice is made between any two of the routes in the feasible solution, as shown in Figure 3. Then, when exchanging between nodes, it is checked whether there is an improvement. If there is an improvement, a new route is created, but the route obtained here must be feasible. To guarantee feasibility, control is made with an environment simulator.

3.3 Adam-BCO

The Adam algorithm's inability to prevent local traps and the BCO's inability to examine different solution spaces, made the integration of both algorithms necessary. Integrating the Adam algorithm with the body change operator will further eliminate the weaknesses of both algorithms.

Actions are first produced as described in the Adam algorithm, to start the learning process with the proposed hybrid Adam-BCO algorithm. Algorithm 3 shows the mechanism of the proposed Adam-BCO algorithm.

Algorithm 3. Pseudocode of the Adam-BCO algorithm

Require: α: Step size, β_1 and $\beta_2 \in$: decay rates for the moment estimates
Require:$f(\theta)$: Weight of deep-learning with parameters θ
Require: θ_0: Initial weights of deep-learning and set to 0
$m_0 \leftarrow 0$
$\vartheta_0 \leftarrow 0$
$t \leftarrow 0$ (Iteration (150))
while *currentiteration* $<t$ **do**
$t \leftarrow t+1$
If t % 5$\leftarrow$0
perform: BCO algorithm to Gch : $\{Gch_p, Gch_c\}$
end if
while *currentiteration* <50 do
perform: BCO algorithm to ch : $\{ch_p, ch_c\}$
end while
$g(\theta_t) = tanh\ (\theta_t^{max})$:Greedy rule
get the values of gradients ($g(\theta_t)$)
update first-moment estimate (m_t)
update second-moment estimate (ϑ_t)
compute bias-corrected first moment estimate ($\hat{m}_t$)
compute bias-corrected second moment estimate ($\hat{\vartheta}_t$)
update weights (θ_t)
end while
return θ_t

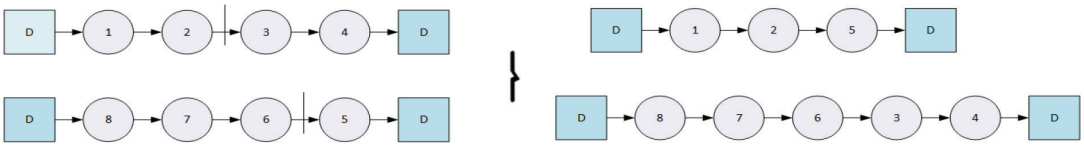

Figure 4. Operation mechanism of BCO algorithm

In the earlier chapters, randomly generated initial solutions create actions, and at the same time, actions refer to inputs. The parameters required by the Adam algorithm are specified, and the BCO algorithm improves the best solution in every five iterations. Thus, weight updates in the network are calculated according to the Adam algorithm. In addition to using the best solution in every five iterations, the BCO algorithm tries to improve the Greedy rule solutions obtained during the first 50 iterations. Regardless of whether the improvement is provided, the network weights are updated according to the mechanism of the Adam algorithm. Through the hybridization of both algorithms, it is ensured that more varied solution spaces are searched, and the Adam algorithm is not trapped in local solutions.

Figure 5 shows the Adam algorithm's operation mechanism for finding possible routings and scheduling. For the Adam-BCO hybrid algorithm to perform the learning action, feasible routing and scheduling are input as action. The best solution obtained in every five iterations is improved, and the weight is updated. Also, the solutions that are selected and created with the Greedy rule are improved in up to fifty iterations. Later, weights are updated, and the learning process that lasts 150 iterations is completed.

4. Numerical Results

The Adam algorithm, the BCO method, and Adam-BCO hybrid algorithm are implemented in Python using a laptop with INTEL I5-3360M, a 2.80 GHz processor, and a 4 GB memory. A time windows technician dataset illustrates the performances of the three algorithms. The dataset includes 29 instances that consist of travel times, allowed days, time windows, proficiencies, 25 tasks, and one central depot. Table 2 illustrates the notation used in Table 3.

Table 3 illustrates the comparison table of Adam, BCO, and Adam-BCO algorithms. The results in Table 3 show why the Adam method should be implemented with the BCO method. The starting and ending intervals of the daily working hours of the C, R, RC test sets are given as [0, 1236], [0, 230], [0, 240]. Working times, service times and time constraints of the three datasets differ. As a result of these differences, problem sets produce solutions at different solution values and times. The fact that the three primary datasets have different characteristics makes it possible to perform a better evaluation of the efficiency of the algorithms or hybrid methods to be applied.

Considering the average of the best solution values of the BCO algorithm and the Adam algorithm-based deep learning, it is seen that they are very close to each other. Adam algorithm-based deep learning produces feasible solutions with a 9.99% Gap and the BCO algorithm with an 8.82% Gap. It is crucial that the Adam algorithm-based deep learning obtain feasible solutions for TRSP with the specified Gap. Even methods that give exact and heuristic solutions do not guarantee optimal results and achieve higher solution times. Here, the average time for feasible solutions obtained with Adam algorithm-based deep learning is 0.14 minutes.

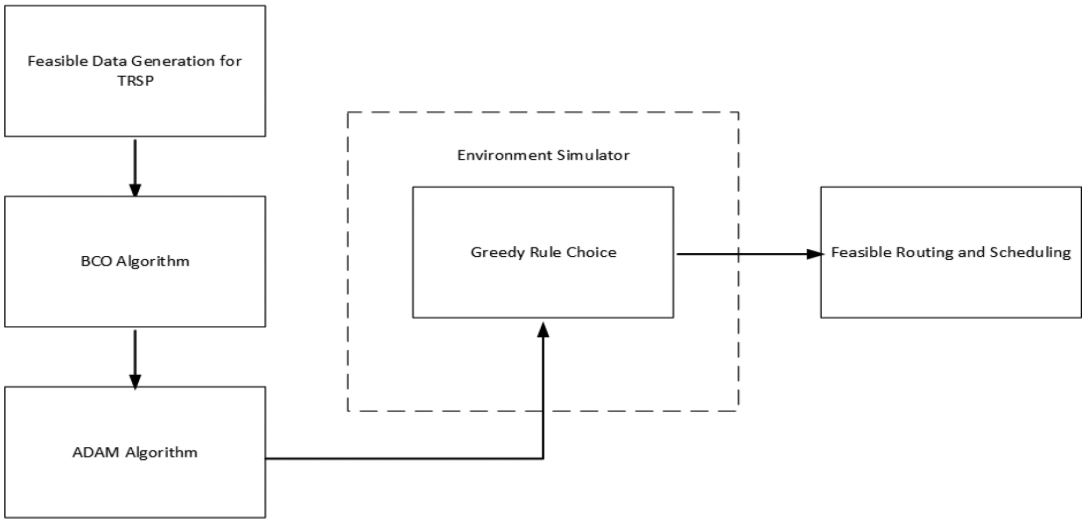

Figure 5. Operation mechanism of Adam-BCO algorithm in TRSP

Table 2. The notation is used in Table 3

Instance	Instance name
Best Cost	Best solution cost acquired by running algorithm ten times
BKS	Best known solution
NBKS	Number of best results acquired by the corresponding algorithm (bold)
CPU Time	A computational processing unit (Minute)
Gap	Percentage gap of the solution cost acquired by the related algorithm
Ins	Instance number

The aim is to integrate the Adam algorithm-based deep learning method with the BCO algorithm and obtain feasible solutions. With the randomness in the structure of the initial solutions required to reach different search spaces in the TRSP, the average initial solution values may differ, although not significantly. Some of the obtained initial solution values may be decent, and provide better solutions, nonetheless, in some cases, this does not give the possibility of searching for other solution spaces.

The Adam algorithm provides the average best cost of 443.62 units, and the average best found of 474.82 units. The BCO algorithm provides the average best cost of 441.83 units, and the average best found is 465.31 units. The Adam algorithm has been integrated with the BCO algorithm, and Adam-BCO based deep learning algorithm has been applied. Performing the hybrid algorithm provides the average best solution value as 430.47 units, and the average best found as 461.58 units. The average best value offers a difference of 5.71% Gap from the best solutions. The average solution time increases from 0.14 minutes to 4.03 minutes, but in return, Gap decreases from 9.99% to 5.71%. The performing hybridization of both algorithms based on deep learning provides effective feasible solutions considering the results.

Table 3. The instance results of the three algorithms

Dataset		DL Algorithm	Adam Algorithm					BCO Algorithm					Adam-BCO Algorithm					
Ins	**BKS**	**Initial Solution**	**Best Cost**	**Usage of Depot**	**Gap (%)**	**CPU Time (min)**	**Average Best Found**	**Best Cost**	**Usage of Depot**	**Gap (%)**	**CPU Time (min)**	**Average Best Found**	**Initial Solution**	**Best Cost**	**Usage of Depot**	**Gap (%)**	**CPU Time (min)**	**Average Best Found**
C-101	**230.10**	297.18	259.70	5	12.86	0.06	273.69	258.70	5	12.43	1.71	275.45	303.25	231.20	5	0.44	5.38	247.06
C-102	**227.60**	284.96	259.80	5	14.15	0.06	272.80	233.10	5	2.42	2.52	251.26	292.48	234.60	5	3.08	5.65	245.01
C-103	**227.60**	288.83	243.40	5	6.94	0.06	275.74	233.30	5	2.50	4.84	246.63	276.19	**227.60**	5	0.00	5.80	242.17
C-104	**225.70**	260.05	229.80	5	1.81	0.06	256.34	229.90	5	1.86	8.23	243.10	266.32	**225.70**	5	0.00	5.63	233.68
C-105	**230.10**	291.90	281.60	6	22.38	0.07	287.90	239.20	5	3.95	18.75	251.95	282.86	239.20	5	3.95	6.14	248.46
C-106	**230.10**	302.48	257.30	5	11.82	0.08	293.75	248.60	5	8.04	2.02	271.99	307.49	**230.10**	5	0.00	6.49	255.04
C-107	**278.80**	333.73	296.90	5	6.49	0.09	316.53	292.70	5	4.99	2.02	316.82	346.03	280.50	5	0.61	4.38	332.16
C-108	**278.80**	350.94	301.30	6	8.07	0.07	339.62	280.50	5	0.61	0.31	336.51	330.77	280.00	5	0.43	1.67	318.66
C-109	**277.30**	344.17	316.90	6	14.28	0.09	331.11	317.30	6	14.42	0.25	337.58	330.24	284.90	5	2.74	0.56	312.46
R-101	**440.20**	518.67	442.40	5	0.40	0.09	458.67	447.60	5	1.68	3.16	458.60	495.80	445.80	5	1.27	7.37	459.49
R-102	**421.00**	525.38	457.30	5	8.62	0.09	500.30	436.80	5	3.75	12.88	464.22	495.82	435.70	5	3.49	7.63	452.60
R-103	**406.30**	490.43	418.40	5	2.98	0.07	457.78	424.30	5	4.43	2.90	433.97	482.81	412.70	5	1.58	6.47	432.62
R-104	**404.30**	490.31	439.60	5	8.73	0.08	459.46	416.20	5	2.94	9.28	426.24	494.56	406.30	5	0.50	6.64	436.01
R-105	**490.80**	620.66	516.90	6	5.32	0.10	559.81	520.70	6	6.09	3.14	556.05	604.15	511.20	6	4.16	5.39	542.15
R-106	**476.40**	584.63	498.90	5	4.72	0.10	534.63	487.00	5	2.23	4.93	537.00	636.48	480.30	5	0.82	6.63	530.28
R-107	**476.40**	594.49	508.60	5	6.76	0.11	562.09	513.80	5	7.85	2.01	528.06	590.67	499.30	5	4.81	6.42	550.67
R-108	**470.60**	592.45	501.60	5	6.59	0.10	557.24	514.40	5	9.31	2.24	550.87	598.11	498.30	5	5.89	6.74	555.62
R-109	**480.60**	595.73	546.60	5	13.73	0.12	565.15	552.00	5	14.86	1.70	566.05	592.55	543.40	5	13.07	2.86	567.95
R-110	**480.60**	600.72	552.70	5	15.00	0.09	579.41	551.70	5	14.79	0.54	557.40	611.11	556.40	5	15.77	0.24	569.96
R-111	**478.30**	620.88	494.20	5	3.32	0.11	542.88	500.80	5	4.70	6.83	519.25	595.49	515.40	5	7.76	7.53	525.90
R-112	**395.30**	486.35	417.40	5	5.59	0.10	436.14	413.80	5	4.68	0.55	435.49	479.65	**395.30**	5	0.00	6.15	421.69
RC-101	**507.27**	656.77	592.14	5	16.73	0.12	616.76	586.62	5	15.64	0.21	614.80	653.96	581.02	5	14.54	1.15	621.65

Table 3. The instance results of the three algorithms

Dataset		DL Algorithm	Adam Algorithm					BCO Algorithm					Adam-BCO Algorithm					
Ins	**BKS**	**Initial Solution**	**Best Cost**	**Usage of Depot**	**Gap (%)**	**CPU Time (min)**	**Average Best Found**	**Best Cost**	**Usage of Depot**	**Gap (%)**	**CPU Time (min)**	**Average Best Found**	**Initial Solution**	**Best Cost**	**Usage of Depot**	**Gap (%)**	**CPU Time (min)**	**Average Best Found**
RC-102	**475.14**	662.99	552.28	6	16.24	0.23	592.69	586.89	6	26.83	0.22	607.84	667.68	572.99	6	20.59	0.26	601.08
RC-103	**488.58**	686.75	577.35	6	18.17	0.09	614.98	589.17	6	20.59	0.35	620.85	687.55	575.56	6	17.80	0.40	628.11
RC-104	**482.42**	732.71	546.92	5	13.37	0.13	586.51	560.66	5	16.22	0.33	594.65	681.12	561.44	5	16.38	0.17	589.70
RC-105	**545.50**	754.82	609.77	6	11.78	1.20	648.63	631.31	6	15.73	1.16	661.15	753.55	583.21	6	6.91	1.21	639.56
RC-106	**510.33**	720.23	599.04	6	17.38	0.09	610.72	571.47	6	11.18	0.25	599.53	741.71	559.43	6	9.62	0.63	604.15
RC-107	**509.89**	720.61	552.03	6	8.27	0.10	618.26	579.32	6	13.62	0.35	616.16	712.83	540.97	5	6.10	0.62	612.46
RC-108	**553.35**	692.88	594.18	6	7.38	0.23	620.25	595.11	6	7.55	1.15	614.51	694.75	575.03	6	3.92	0.72	609.51
Avg.	403.43	520.78	443.62	5.35	9.99	0.14	474.82	441.83	5.28	8.82	3.27	465.31	517.45	430.47	5.21	5.73	4.03	461.58
NBKS			**0**					**0**						**3**				

5. Conclusions

This paper proposed an Adam-BCO hybrid algorithm to solve TRSP, which is a combinatorial optimization problem. The TRSP consists of assigning technicians to teams, assigning teams to tasks, creating routes, and the selected technician group that must meet certain level of qualification. The TRSP included a pre-determined set of technicians who have various talents and a set of tasks requested by the customer at various locations. To perform tasks consisting of customer requests, the set of technicians are divided into teams, and teams travelling in different locations are assigned to the tasks. The numerical results provide that the performance of the proposed Adam-BCO algorithm is experimentally compared with the separate performance of the Adam and BCO algorithms for the solution of the TRSP on Pekel's modified instances. The conclusions of this article are presented as follows:

- The Adam algorithm was caught on local traps, and the BCO method failed to examine different solution spaces. The two algorithms have been integrated to eliminate each other's weaknesses, and the possibility of searching for new solution spaces has emerged. The changes are carried out to provide that the Adam algorithm-based deep learning method provides fast and ideal feasible solutions for TRSP.
- There are 26 hidden neurons in the first hidden layer. The main reason for choosing 26 neurons is that there are 25 customers and one central depot in the discussed TRSP. Since there are 26 nodes, 26 neuron and input entries are made. The number of hidden neurons would increase if the number of nodes addressed in the problem, the number of customers and depots increases.
- The Adam algorithm-based deep learning method can produce solutions close to optimal results in some data sets using the advantage of gradients' first and second moments, considering the parameters included in its structure.
- Although the Adam algorithm based deep learning method has a 9.99% Gap, it provides a swift solution using the advantage of the first and second moments of gradients, considering the parameters included in its structure.
- As a result of the randomness in the structure of the initial solutions required to solve the problem, the average initial solution values may differ, although not significantly. Some of the obtained initial solution values may be decent and provide better solutions, nonetheless, in some cases, this does not enable the search for other solution spaces.
- The deep learning method based on the Adam-BCO algorithm offers the best feasible solutions with a 5.73% Gap using the powerful sides of the two algorithms. While the Adam algorithm provides a faster approach to solution spaces, the BCO method explores those that could not be investigated before.
- The average best solution value obtained by the Adam-BCO algorithm-based deep learning method promises to reach better solutions in the future. With the combination of the strengths of both algorithms, more effective results have emerged. This situation shows the importance of integrating and researching other algorithms.

In future studies, better results can be obtained with the use of different learning algorithms in combinatorial problems such as TRSP. Also, a more effective solution space search can be performed by integrating heuristic methods and deep learning algorithms.

References

Anoshkina, Y., and Meisel, F. (2019). Technician Teaming and Routing with Service-, Cost-and Fairness-Objectives. Computers & Industrial Engineering.

Çakırgil, S., Yücel, E., and Kuyzu, G. (2020). An integrated solution approach for multi-objective, multi-skill workforce scheduling and routing problems, Computers & Operations Research 118: 104908.

Charris, E. L. S., Montoya-Torres, J. R., and Guerrero-Rueda, W. (2019). A decision support system for technician routing with time windows. Academia RevistaLatinoamericana de Administración.

Chen, X., Thomas, B. W., and Hewitt, M. (2016). The technician routing problem with experience-based service times. Omega, 61, 49-61.

Duchi, J., Hazan, E., and Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization. The Journal of Machine Learning Research, 12:2121–2159.

Fu, T., Wang, C., and Cheng, N. (2020). Deep-learning-based joint optimization of renewable energy storage and routing in vehicular energy network. IEEE Internet of Things Journal, 7(7), 6229-6241.

Graf, B. (2020). Adaptive large variable neighborhood search for a multiperiod vehicle and technician routing problem. Networks, 76(2), 256-272.

Hernández-Jiménez, R., Cárdenas, C., and Rodríguez, D. M. (2019). Towards the Optimal Solution for the Routing Problem in Vehicular Delay Tolerant Networks: A Deep Learning Approach. IEEE Latin America Transactions, 17(12), 2028-2036.

Hussain, D., Khan, M. A., Abbas, S., Naqvi, R. A., Mushtaq, M. F., Rehman, A., and Nadeem, A. (2021). Enabling Smart Cities with Cognition Based Intelligent Route Decision in Vehicles Empowered with Deep Extreme Learning Machine. CMC-computers materials & continua, 66(1), 141-156.

James, J. Q., Yu, W., and Gu, J. (2019). Online vehicle routing with neural combinatorial optimization and deep reinforcement learning. IEEE Transactions on Intelligent Transportation Systems, 20(10), 3806-3817.

Khalfay, A., Crispin, A., and Crockett, K. (2017, September). A review of technician and task scheduling problems, datasets and solution approaches. 2017 Intelligent Systems Conference. pp. 288-296.

Kingma, D. P., and Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Koh, S., Zhou, B., Fang, H., Yang, P., Yang, Z., Yang, Q., and Ji, Z. (2020). Real-time deep reinforcement learning based vehicle navigation. Applied Soft Computing, 96, 106694.

Kovacs, A. A., Parragh, S. N., Doerner, K. F., andHartl, R. F. (2012). Adaptive large neighborhood search for service technician routing and scheduling problems. Journal of scheduling, 15(5), 579-600.

Lee, K. B., A Ahmed, M., Kang, D. K., and Kim, Y. C. (2020). Deep Reinforcement Learning Based Optimal Route and Charging Station Selection. Energies, 13(23), 6255.

Mathlouthi, I., Gendreau, M., and Potvin, J. Y. (2021). A metaheuristic based on Tabu search for solving a technician routing and scheduling problem. Computers & Operations Research, 125, 105079.

Mathlouthi, I., Gendreau, M., and Potvin, J. Y. (2018). Mixed integer linear programming for a multi-attribute technician routing and scheduling problem. INFOR: Information Systems and Operational Research, 56(1), 33-49.

Pekel, E. (2020). Solving technician routing and scheduling problem using improved particle swarm optimization. Soft Computing, 24(24), 19007-19015.

Pekel, E., and Kara, S. S. (2019). Solving fuzzy capacitated location routing problem using hybrid variable neighborhood search and evolutionary local search. Applied Soft Computing, 83, 105665.

Tieleman, T. and Hinton, G. (2012). Lecture 6.5 - RMSProp, Coursera: Neural Networks for Machine Learning. Technical report.

Wang, J., and Sun, L. (2020). Dynamic holding control to avoid bus bunching: A multi-agent deep reinforcement learning framework. Transportation Research Part C: Emerging Technologies, 116, 102661.

Whitley, D. (1994). A genetic algorithm tutorial. Statistics and computing, 4(2), 65-85.

Zamorano, E., and Stolletz, R. (2017). Branch-and-price approaches for the multiperiod technician routing and scheduling problem. European Journal of Operational Research, 257(1), 55-68.

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal
Regular Issue, Vol. 11 N. 2 (2022), 207-235
eISSN: 2255-2863
DOI: https://doi.org/10.14201/adcaij.27430

Efficient Content Based Video Retrieval System by Applying AlexNet on Key Frames

Altaf Hussain[a,b], Mehtab Ahmad[a], Tariq Hussain[a,c*,] and Ijaz Ullah[d]

[a] Institute of Computer Science & IT (ICS/IT), The University of Agriculture Peshawar, Pakistan
[b] Department of Accounting & Information Systems, College of Business and Economics, Qatar University, Doha, Qatar
[c] School of Computer Science and Information Engineering, Zhejiang Gongshang University Hangzhou, China
[d] University of Rennes 1, France
altafkfm74@gmail.com, mehtab@aup.edu.pk, uom.tariq@gmail.com, ijaz_flair@yahoo.com
*Correspondence: uom.tariq@gmail.com

KEYWORDS

CNN; K-Mean, CBVR; color histogram; accuracy; loss; BoW; AlexNet

ABSTRACT

The video retrieval system refers to the task of retrieving the most relevant video collection, given a user query. By applying some feature extraction models the contents of the video can be extracted. With the exponential increase in video data in online and offline databases as well as a huge implementation of multiple applications in health, military, social media, and art, the Content-Based Video Retrieval (CBVR) system has emerged. The CBVR system takes the inner contents of the video frame and analyses features of each frame, through which similar videos are retrieved from the database. However, searching and retrieving the same clips from huge video collection is a hard job because of the presence of complex properties of visual data. Video clips have many frames and every frame has multiple properties that have many visual properties like color, shape, and texture.

In this research, an efficient content-based video retrieval system using the AlexNet model of Convolutional Neural Network (CNN) on the keyframes system has been proposed. Firstly, select the keyframes from the video. Secondly, the color histogram is then calculated. Then the features of the color histogram are compared and analyzed for CBVR. The proposed system is based on the AlexNet model of CNN and color histogram, and extracted features from the frames are together to store in the feature vector. From MATLAB simulation results, the proposed method has been evaluated on benchmark dataset UCF101 which has 13320 videos from 101 action categories. The experiments of our system give a better performance as compared to the other state-of-the-art techniques. In contrast to the existing work, the proposed video retrieval system has shown a dramatic and outstanding performance by using accuracy and loss as performance evaluation parameters.

1. Introduction

The worldwide connectivity grows day by day, it is important to index and retrieve the video data competently to combat the information explosion. The common system of video indexing and retrieval is usually achieved by manual annotations. Allowing more data to be generated and collected, a growing portion of that data will be real-time information, according to IDC (International Data Council). By 2025, nearly 30 percent of the so-called «global data sphere» will be real-time information. Currently, digital data has reached 281 exabytes. Furthermore, according to the IDC report, the digital data is to be 44 times larger in 2020 than in 2009. Currently, most of the data is unstructured such as images, video, text, and music. In information retrieval, content-wise video retrieval is the basic problem (Iqbal et al., 2018). The related video retrieval denotes the work of retrieving the same video gathering, by the user input query to the system. Multimedia resources get wide benefits from vigorous video retrieval schemes such as news channel analysis of videos, broadcasting of desired videos, analysis of marketable videos, modern museum-like digital, and action of surveillance in videos. In the previous years, the collections of videos small and video retrieve were placed manual annotation. The current high growth of digital data is increasing day by day, due to this increase the advanced technology will be supported in multimedia schemes. Manual annotation has been no more reliable in the retrieval of videos. As a result, it makes a great request for automatic video retrieval systems. User input query and retrieve video from the database using a specific model query. The submitted query information is annotating in a new advanced method and this information saves inefficient way. Due to the digital universe of different cultures and multimedia interaction, content-based image and video retrieval have a wide area of inventions. While the digital data increasing year by year.it is a very big problem to annotate the digital data content with keywords (Thanh et al., 2014).

For graphical content extraction, the Convolutional Neural Network (CNN) is used and CNN is artificial intelligence. For object recognition and human action recognition Convolutional neural network is very powerful and outstanding. Neuron includes convolutional Neural Networks and these are which have weights learnable and preferences. With nonlinearity in start one by one neuron load with digital values and calculate with dot product and follow till the processed completion. Convolutional Neural Network design provide a correct statement on behalf of stating inputs digital data and

call the network to compile these properties to the design. A reduce the number of parameters to the next function makes that more efficient to implement the network (Iqbal et al., 2018). In the current era of information technology, digital data is played a vital role in law enforcement, office work, and entertainment, and everywhere in life. The internet and offline systems are big sources of videos. On the internet and offline databases have big data and all users access these data through an index. All the mention resources use indexing retrieval. Somehow in Google using content-based image retrieval but does not have video retrieval. Formulation of the query annotation is generally the automation of content instead of indexing annotation. Content analysis is very difficult to work in the video but in the image, so not hard. In a single image, annotation is very easy through handcrafting. Nowadays the video is generally limited to simple content-based features for researches.

Human action recognition is a very hard task in research, and video data is available everywhere. In the present videos and cameras, human action recognition is a very difficult task. And in the data set, there are a lot of human actions on which we train machines. Featured based video retrieval is a very difficult job. A big reason is a high variation in videos in which the meaning full idea can occur in multiple conditions shot setting, lightning. i.e. a man in the video riding on a bicycle has a variation like multiple points, resolution, advents, type of bicycle, and movement of the camera. Almost all the researchers aimed at these challenges. Hence some features will be needed to match two videos (Jones and Shao, 2013).

1.1 Motivation and Background

The primary use of the system is to retrieve similar videos that are present in the database, such as retrieving the whole movie from its trailer. And this system can also use for database management, just search a video in the database and store it in categories wise. But once the system is implemented it can be used for diversified applications just by changing the algorithms. For example, if we change the algorithm to object detection. There are some other applications of the content-based video retrieval system. Searching videos in a large database is a very useful application in this environment because through the text they may retrieve fake names videos. From news, the archive retrieves news on demand. This research can also use for security purposes to detect a specific event. In education searching lecture related data. In the shopping mall, we can use for product information. Searching the desired video in a big database by text may have a fake name or wrong name so you will be retrieved undesired video. Hence content-based video retrieval can use in any digital environment.

Resourceful video retrieval has been applied earlier on the videos but to the best of our knowledge, it was neither efficient nor accurate. It has some limitations and the first one is that it was taking a long time i.e., it has the highest time complexity at the time of process when video data was being called by their contents. The existing work uses some approaches for the sake of improvements of the CBVR system. At the early stage, the video query was obtained. Then the feature extraction was taken into account. Then the query of the target file of the video was forwarded for a similar matching of the contents based on keyframes and some identical terms. On the other hand, the video collection and then the extraction were taken into account. These all were then checked into their created database and that database was linked directly with the similarity matching and at the end, retrieved videos were to be obtained as requested by the earlier video at the time of searching into the database. Unlike the existing work, the proposed work possesses the advantage of using an efficient video extraction system. The main difference between the existing work and the proposed work is that the proposed work takes less time at the time of processing the video retrieval and gives an efficient and robust result which is remarkably an outstanding performance as compared to the existing work. Additionally, the proposed

work performs well by using multiple feature extraction and getting or merging the positive features of these CBVR models.

1.2 Problem Statement

Text-based video retrieval systems mostly retrieve incorrect video because people namely caption the videos with a fake title. When users are searching for the desired video they are not found the desired video due to the fake. Manual annotation becomes very hard and expensive. Manual annotations also take more time to search for desired videos. If the videos are organized in a well-known manner the text-based retrieval will be a better option. But without annotation the retrieval of videos is inefficient. The previous research on this area was not used a good framework and algorithm. If someone used CNN they were not used good handcrafted methods for features extraction and benchmark datasets. And used RGB2GRAY for feature extraction and the videos are most colorful. For this purpose, the grayscale cannot achieve good performance on color videos. In light of the above-mentioned problem with the CBVR in the existing work, the proposed work will be using the best features of the CNN model for accurate extraction of the videos based on contents and mainly the graphical representation of the videos. The proposed CNN model has the best accuracy in contrast with the existing used models.

1.3 Research Contributions

1. To develop a Content-Based Video Retrieval system using an AlexNet model on keyframes.
2. To evaluate the proposed method based on AlexNet in terms of accuracy and loss with FC6, FC7, and FC8 layers of the CNN model.

2. Literature Review

In this section, the related works of the proposed research work have been discussed along with different perspectives from the author's point of view. The majority of the researchers have worked on contents based video retrieval scheme for the sake of improvements in contents feature extraction in video files. Along with these some open research issues and challenges have been also discussed and their solution. Research gaps have been revealed from the majority of the related works from which the performance evaluations have been carried out by using simulation-based techniques. In short, this section gives a thorough review of the related work with the contrast of the proposed work to find the research gap and also to find out the level of percentage of accuracy of the existing works.

Zhang et al. (2019) authors suppose that to solve the problem of large-scale video retrieval by an image query. They were given a top k image video query. After that, they combined the CNN for short and BoVW for short modules to construct a system for video frame information extraction and representation. For a large scale, video retrieval needed they proposed a visual weight inverted index and algorithm for the improvement of performance and accuracy of the process. And the suggested system got an improvement in the state of the approaches with better accuracy. They found the top k image for query and retrieved the top most related videos from the database. And experimentally they achieved good result from the state of the art methods with respect to accuracy. Bolettieri et al. (2019) the authors suggested a vision system for video retrieval. It contained a content-based analysis and module

of retrieval. They were contained searching through the keywords. and on the base of object search and content matching search. Euclidean matching was the most approach. And the suggested method is based on deep learning for the analysis of contents. Actually, they covered all the textual and visual descriptors extracted from the videos into textual combinations. They were representing the version of vision in the image retrieval system used to search for videos.

Chen et al. (2017) proposed and investigate the problem of personalized keyframe recommendations; they overcome the above problem. They designed a new keyframe recommender that was instantaneously model textual and visual features in an integrated structure. They posted comments on previously personalized base review frames, in an integrated multi-modal space they were able to encode deferent user benefits, and they would thus select keyframes in a modified method, which, according to his knowledge, it was the first time work in the research field of video content analysis. On various measures and experimental results show that has method accomplishes better than its challengers. Algorithms of clustering were having a very great point of research; they were suggested a real scheme for clustering by added important assets of video to collect the same frames into clusters. In the last, whole clusters' the center cluster was selected as static video summarization. Has work recognized a high relevant frame from the cluster of frames automatically compared to previous work, they test and the train has worked and compare with some traditional clustering techniques. The experimental result that has the proposed method has better accuracy and performance. They proposed VRHDPS based on HDPS, there was no need to mention the clusters, as the algorithm of clusters. HDPS was trusted on the centroid cluster frame and was considered by a great value than their adjacent and by a high difference were very low density to the centroid frame. In HDPS some structures of video summarization have not been measured. Thus, they proposed the VRHDPS clustering algorithm, which was more reliable for video summarization.

Ouadrhiri et al. (2017) suggested using spatial-temporal types to describe videos. Bounded Coordinate of Motion Histogram was acquainting to describe and match video subsequences in a reduced calculation interval. Furthermore, the suggested method was adaptive: presented a training procedure. And on a database of 1707 movie clips the cost was expressed in comparison measurement has its accuracy improved by more than double (approximately 38%) in association with extended fast dynamic time distorting method. the challenge was, to make the related signatures built on gesture and, and other was a very fast similarity measure must, therefore, be used to matched video subsequences. A novel solution was proposed in this research: for classification and matching, videos subsequences in a compact calculation time Bounded Coordinate of Motion Histogram was introduced. Sedighi and Fridrich, (2017) suggested the possibility behind the CNN to used present structure to improve the structure of kernels linear pixel analysts in feature-based coding. In CNN the additional objective function in the form of some scalar classifier performance condition by the missing function. For the optimization, the job provided a method for histograms within a CNN that could be powerful gradient descend systems. Thus, the first step, for the Caffe CNN suite they were to deploy a histogram layer. As the structure blocks of this layer to get an appropriate backflow of gradients over the layer and to enable learning of the parameters behind this layer using mean shifted Gaussian functions. In PSRM, each SRM kernel was trained on fifty-five arbitrary two-dimensional kernels and they are rotated and mirrored forms. The result of a large number of projections was the higher the detection rate of this feature was set, high the computational costs came. This problem could provide a powerful feature set impracticable for applications with restricted time and computational power. The proposed study also hints at the opportunity to extract more information in the last layers of CNNs to get the way for better

network projects. Asha et al. (2018) used multiple features for a realistic content-based video retrieval system (CBVR) to retrieve videos expertly. The proposed method converts videos into scenes, using keyframes that were extracted and the algorithm of histogram-based scene change detection. Straight forward rules were used for keyframes multiple feature extractions. They used the Euclidean distance formula to measure the values in feature vector and query. The CBVR systems were compared with the proposed system with a single feature. After the experiment shows that from the single feature system the multiple feature system performs better.

Iqbal et al. (2018) used the digital image processing method (Eigenface, the histogram of gradients, active appearance model, and Haar features) on Query Process Model that retrieved a list of videos from a database. And clustering method (k mean, SVM, and K-Nearest Neighbor) and consequence (testing and training) are with their confusion matrix (specificity and Sensitivity). As compared to convolutional Neural Network result with other clustering techniques the CNN was outstanding. According to inefficient hardware, they did not achieve a better result. For more research in this area of objection recognition, the complex convolution neural network will be used. And dropout architecture used unlimited hardware resources (TPU or GPU) in Deep Belief Network (DBN) with multiple hidden layers are used. Sikos (2018) excogitated that low-level feature extraction does not correspond to the concept. Events and persons are shown in videos, As well as to decrease the semantic Gap, the shown concepts and their spatial relations were described in a computational form used formal descriptions from designed data resources. Events and actions information was described as inefficient rule-based mechanisms. For the computational spatiotemporal annotation of complex video scenes that were suitable structured with audio and textual description, the annotation of videos was manually or panoramically done, and the presented research can be used in scene interpretation, and content-based video retrieval, and video understanding. Song et al. (2018) proposed a framework named Self Supervised Video Hashing (SSVH). That was able in hash fashion manner to capture the time base nature of the videos in the end to end learning. Two main problems specifically they addressed: 1) how to design converter and reconverted system to generate for video a binary code; and 2) how for accurate video retrieval train the binary codes with the ability. FCVID and YFC are two data sets on which the results were tested. And show that has SSVH approach has suggestively outclassed the state of the art systems and achieve the presently best result on the task.

Tarigan et al. (2018) invented an implementation of CBVR using SURF. In query give an image, the system search videos in database and retrieve similar videos with query image that matched with video frames. The objective of the authors was to measure performance. Through precision and recall, the performance of the proposed system was measured. Two sets of samples not in frame and in-frame were used for performance testing. Also, they limit five categories of simple only: pets, kitchen, body parts, fruits, and eating utensils. The test shows the program gives the performance on 37.5% precision and on recall 57.75% average value for not in frame tested, while the in-frame test gives precision 59% and for recall 51%. There is no relation between not in the frame and in the frame and speed Complexity depends upon the length of the video and query image. Lingam and Reddy (2019) described an RPCA framework for keyframe extraction. In unstructured user videos, they focused on the interesting application of extracting keyframes from the videos. By the observation, the proposed framework was motivated that RPCA enters data into (1) the features of the dataset a low-position segment that uncovers the methodical data over the framework, and (2), in the equivalent dataset a lot of inadequate parts every one of which contains unmistakable data. The two data types were consolidated into a solitary '1-standard based non-arched advancement issue to recover the needed number of keyframes. Besides, the answer to the advancement issue they were plans another iterative calculation. The proposed RPCA

based system doesn't require shot(s) division, semantic comprehension, or acknowledgment of the key film. At long last, on the client video, the tests were performed. A correlation of the outcomes acquired by has a strategy with related best in class and the ground truth calculations obviously show the attainability of the proposed RPCA based structure. As appeared to assess the multifaceted nature they surveyed the normal preparing time per outline. As indicated by these trials, to process a solitary edge has HIP put together method took 1.469 seconds with respect to average, with 0.233 seconds per outline for the RPCA decay of the info signal into low position and meager segments, and advancement issue fathomed then the on normal 1.236 seconds per outline. The referenced number was reliant on the computational intensity of the hidden equipment. Intel Core E7500 2.93 GHz stages were utilized in his work. Diminished the normal handling time per outline by an element, and the picture size similar to the size of 80x60, which they were utilized in has tried. For example, 1frame/sec utilized a pre-testing rate and diminished to 0.0612 sec/outline the normal time per a single frame.

3. The Proposed Methodology

In this section, the research methodology of the proposed work has been discussed. The methodology of the research work consists of proposed work which is the core contribution. Along with these the related methods and techniques have been discussed that how the proposed work has been carried out by using simulation. The performance evaluation parameters have been proposed for the research work by which the evaluation can be made with the state of the art solutions. This section also gave a thorough idea about the step by step methodology for the proposed work. In which the first step is the proposed framework that is the core of this research.

3.1 Proposed Research Framework

Video processing like indexing, browsing, and retrieval is a procedure of marking videos and sorting out them in an actual way for quick browsing and recovery. Computerization of video indexing can altogether lessen preparing cost while eradicating dull effort. The conventional features utilized as a part of the vast majority of the current video retrieval frameworks are the features, for example, color, texture, shape, movement, face, sound, etc. Clearly more the number of features used to present the information, the better the retrieval precision. In any case, as the feature vector measurement increments with an expanding number of features, there is an exchange off between the retrieval accuracy and complexity. So it is fundamental to have insignificant features present in the videos, slightly. In this research we talk about Video key frames, indexing, extract CNN features through deep learning used for video indexing, browsing, and retrieval.

Our proposed CBVR (Content-Based Video Retrieval) system, architecture shows a novel procedure for similar information retrieval. Our proposed framework includes multiple modules or steps. In the first step, we read the video and convert it into multiple frames, in the next step we are extracted handcrafted features from all frames and select key-frames through unsupervised learning, clustering technique. After selection of keyframes, we extract CNN (ConvNets) features using deep learning from key-frames and save that features in a single feature vector, the above process is used for query video as well as for videos database, at the end we find the difference between query video and database through distance measurement equation (distance matrix) to calculate the distance. And retrieve most similar videos to the query video from the database. The overall architecture is shown in Figure 1.

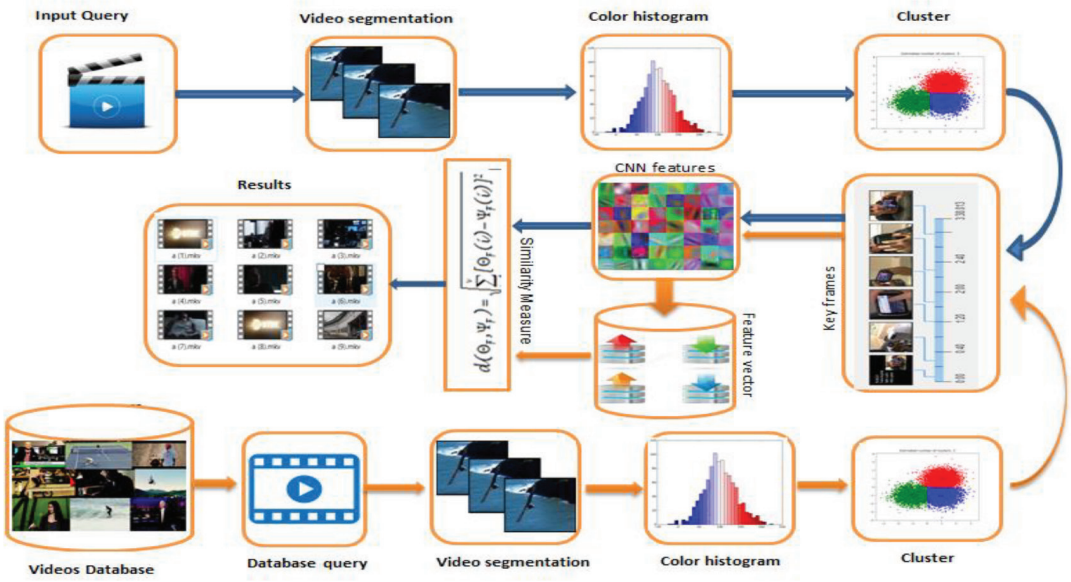

Figure 1. The Proposed Framework

3.1.1 Video Reading

The proposed search engine reads a video through reading video(), function, and find the total duration as well as frame rating of the given video, which is helpful to read video frame by frame through a depicting coding I any environment.

3.1.2 Framing

Every video has a sequence of images, which concatenate with each other in video form. We see a video in television or in any other device it is actually a sequence of the frame, the same sequence of frames makes a shot, and more than one-shots make a scene and some scene is together to make a video.

3.1.3 Color Histogram

In the field of digital image processing, a color histogram is one of the best feature representation techniques that represent all colors in every pixel of an image. The color histogram represents the fixed length of the color range in each and every pixel of the image. Through this technique, we represent or extract the color information from images or from a sequence of the frame and store the given information in a vector.

3.1.4 Hand Crafted Features

The set of handcrafted features contains features concerned with color and edges, to present special information and motion for temporal information. The shot regional properties will describe by color features, such as the average or median color of a specific region. The edge features think about the local

dissemination just as directionality of edges. At long last, the movement features produce local histograms of the direction of development. A total list of the handcrafted features realized in the CBVR framework is exhibited underneath. All handcrafted features utilize the quadratic Euclidean matrix for the correlation of feature vectors. The distances are adapted into a comparability score utilizing a direct change for which the most extreme is resolved exactly. The aftereffects of the single component modules are joined to a solitary lucid outcome set by recording a weighted average over the similarity scores (Rossetto et al., 2015).

3.1.5 High-Level Features

These types of features are extracted through deep learning, utilizing a specific model for feature extraction. There are two types' features, one is special and the other is temporal for motion in video shots. Such deep learning approaches are used to efficiently extract the features and other characteristics, which are performed than another conventional method.

3.1.6 Key Frame Extraction

The color histogram features have been used as a part of the key-frames selection step, however, the video movement has not been considered as a feature. In the key-frames extraction, we must calculate the color histogram of each frame and save it, after calculating the histogram, we apply the clustering technique and then find the centroid of each cluster and find the distance from the cluster of a center and select the nearest frame to the center of the cluster and select that frame as a keyframe, we follow this process and select the one frame from each cluster as a keyframe. We use the k mean clustering algorithm for the selection of keyframe, the main advantage of the k means clustering algorithm is that the method is efficient for the proposed work to implement and evaluate the performance of the suggested method is suitable. The steps followed in the k mean clustering algorithm are;

*Initialize***:** randomly select k of the n information focuses as the mean.

Assignment step: Relate every datum point to the nearest mean.

Update step: For each mean m and every datum point o related to m swap m and o and process the aggregate cost of the design (that is, the normal divergence of o to every one of the information directs related toward m). Select the mean o with the most reduced cost of the configuration.

Repeat substituting stages 2 and 3 until there is no adjustment in the assignments.

The methodology took after for extraction of key edges from a video comprises of the following stages;

Algorithm 1. Pseudo code for K-Mean Clustering

Step1:eachcasingofinfovideoextricateCH
featurevector.Hf=inputimage;
Step2:Acquirefewbunchesutilizingk-meanclusteringalgorithm,
thefeaturevectorsextricated∈
step1.K-numberofclusters∈ video;
Step3:Discoveronefeaturevectortheprinciplegroupwhichis
closertherelatingclustercenter.Cframe=(kframe,I);
Step4:Announcethecomparingvector'sframeaskey-frameofthe
firstcluster.Kf=cframe;
Step5:Applystage3∧stage4onremainingclusters∧
concentratekey-frameskf=(1,2,3,kn);End;

Figure 2 represents the diagrammatical illustration of key-frame extraction.

Figure 2 applying clustering method and then find the centroid of each cluster and find the distance from the cluster of a center and select the nearest point to the center of the cluster and select that point as a centroid, we follow this process and select the one point from each cluster as a centroid. We use the k mean clustering algorithm for the selection of centroid point, the main advantage of the k means clustering algorithm is that the method is efficient for the proposed work to implement and evaluate the performance of the suggested method is suitable.

3.1.7 Convolutional Neural Network (CNN)

Convolutional Neural Network is the branch of deep learning which takes the image in input and assigns most importance to several features/objects in the image and capable to distinguish one from the other. In convert, the required processing is lower as compared to other methods. After the training, they have the ability to learn the characteristics.

For computer visual contents detection the convolutional neural network becomes more dominant and CNN is the branch, a class of Ai. The design of CNN is designed to take automatically and adaptively learn three-dimensional orders of features through backpropagation by using multiple building blocks, pooling layers convolution layers, and fully connected layers. Through the mention layers, input data are converting to output propagation. 2D-CNN described pooling and convolution operations. For three-dimensional, these operations were also performed.

3.1.7.1 AlexNet Model Architecture

This design contained around 650,000 neurons and 60 million parameters. AlexNet design comprises fourteen layers, of which seven are convolution layers, four are Max-Pooling layers and three are fully connected layers. In every convolution layer, there are product channels utilized. The third, fourth, and fifth layers which are associated straightforwardly are called convolutional layers. The fifth convolutional

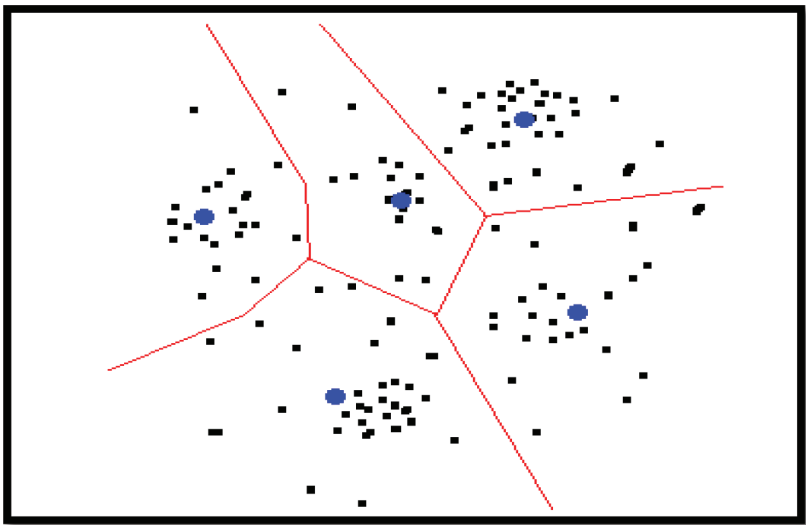

Figure 2. Key-frame extraction

layer is connected to the maximum tricking layer and the last three completely associated layers are utilized for the point of the arrangement. Figures 3 and 4 shows the AlexNet engineering.

The proposed AlexNet model consists of fourteen layers, of which seven are convolution layers, four are Max-Pooling layers and three are fully connected layers. The summary of the proposed Alex net model is shown in Table 1.

In Table 1 shown seven convolutions, four Max-Pooling, three fully connected layers, and Relu as activation. The input image of size (50, 50, 3) is used in which the height of the image is 50, the width of the image is also 50 and channels are three. In all seven convolution layers, the size of the kernel window is kept (3,3), the stride is kept 1 and padding is the same.in four max-pooling layers kernel window size is kept (2,2) and stride of 1. Batch normalization is also used throughout each layer in

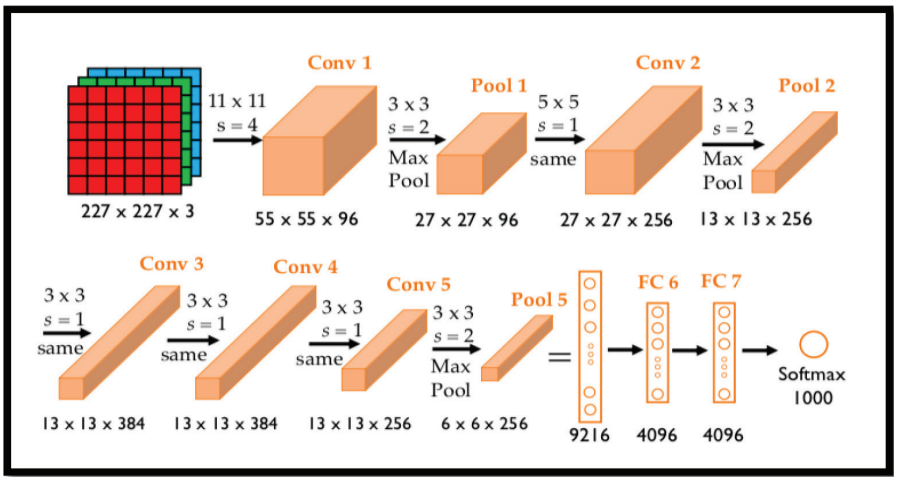

Figure 3. CNN architecture (Han et al. (2015))

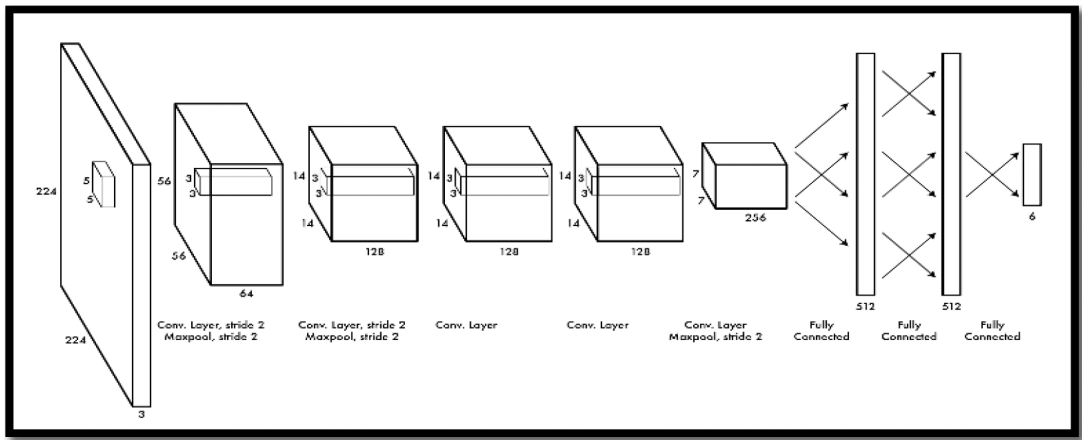

Figure 4. AlexNet Architecture

order to normalize values and increase network speed. In convolution layers drop out of 0.30 is used while in two fully connected layers drop out of 0.5 were used in order to reduce the overfitting problem and the last fully connected layers are used for SoftMax operation.

FC stands for Fully Connected in the given expression which is a layer in the CNN for feature extraction providing an appropriate arrangement for segregation and classification process of images and other associated contents in the videos etc. FC6, FC7, and FC8 are layers that have diverse features in contrast with one another that is achieved by the required task. In short, these are the levels of fully connected layers as mentioned 6, 7, and 8 which denote that each level has an improvement as compared to the second last one. FC7 is an improved and extension of FC6 and FC8 is an improved and extension of FC7. In the FC6 layer, there exist units which connect it with additional features to be carried out. For the fully connected layers, let N_i, P_i, and U_i be the number of output units, parameters (weights), and connections of layer L_i.

- ReLU Nonlinearity

Instead of the tanh function, AlexNet uses Rectified Linear Units. This was very famous and standard at the time. The advantage of ReLU's is its time of training; a 25% error was CNN using ReLU on the CIFAR-10 dataset six times faster than a CNN using tanh.

- Multiple GPUs

GPUs are working back in the day around three GBs of space; the training had 1.2 million pictures they were especially bad. AlexNet can operate on multiple GPUs they put half neuron of the model on one system and a half on another system. This is not enough to train the bigger model but it also Not only does mean that a bigger model can be trained, but it also counts down the training time.

- Overlapping Pooling

At the point when the creators create the cover, they saw a decrease in mistake by about 0.5% and found that models with covering pooling by and large think that it's harder to over-fit.

- The Overfitting Problem

The AlexNet has sixty million parameters included in the model, a general problem in overfitting. Two techniques were applied to decrease the overfitting problem:

- Data Augmentation

The authors were utilized name safeguarding change to spare their information increasingly solid. In particular, they made picture transformations and even reflections, which amplified the preparation, set by a factor of 2048. Performed Principle Component Analysis (PCA) was additionally applied to the RGB pixel esteems to change the forces of RGB channels, which diminished the main 1 blunder rate by over 1%.

- Dropout

This methodology contained «turning off» neurons with a prearranged probability (e.g. 50%). Its mean parameter of the model is used for different iteration, another random neuron from which they get robust features they forced these neurons. They also increase the training time needed for model modification.

Table 1. Detail Summary of Proposed Alexnet Model

	Layer	Kernel Size	Padding	Stride	Activation	Input	Output
1	Convolution	3*3	P=Same	S=1	Relu	(50,50,3)	(48,48,128)
2	B-Normalization	-----	--------	--------	--------	(48,48,128)	(48,48,128)
3	Max Pooling	2*2	P=Same	S=1	--------	(48,48,128)	(47,47,128)
4	Dropout = 0.10	-----	--------	--------	--------	(47,47,128)	(47,47,128)
5	Convolution	3*3	P=Same	S=1	Relu	(47,47,128)	(45,45,256)
6	B-Normalization	-----	--------	--------	--------	(47,47,128)	(45,45,256)
7	Max Pooling	2*2	P=Same	S=1	--------	(45,45,256)	(44,44,256)
8	Dropout = 0.10	-----	--------	--------	--------	(44,44,256)	(44,44,256)
9	Convolution	3*3	P=Same	S=1	Relu	(44,44,256)	(42,42,256)
10	B-Normalization	-----	--------	--------	--------	(42,42,256)	(42,42,256)
11	Max Pooling	2*2	P=Same	S=1	--------	(42,42,256)	(41,41,256)
12	Dropout = 0.10	-----	--------	--------	--------	(41,41,256)	(41,41,256)
13	Convolution	3*3	P=Same	S=1	Relu	(41,41,256)	(39,39,256)
14	B-Normalization	-----	--------	--------	--------	(39,39,256)	(39,39,256)
15	Dropout = 0.10	-----	--------	--------	--------	(39,39,256)	(39,39,256)
16	Convolution	3*3	P=Same	S=1	Relu	(39,39,256)	(37,37,256)
17	B-Normalization	-----	--------	--------	--------	(37,37,256)	(37,37,256)
18	Dropout = 0.10	-----	--------	--------	--------	(37,37,256)	(37,37,256)
19	Convolution	3*3	P=Same	S=1	Relu	(37,37,256)	(35,35,256)
20	B-Normalization	-----	--------	--------	--------	(35,35,256)	(35,35,256)
21	Dropout = 0.10	-----	--------	--------	--------	(35,35,256)	(35,35,256)
22	Convolution	3*3	P=Same	S=1	Relu	(35,35,256)	(33,33,512)
23	B-Normalization	-----	--------	--------	--------	(33,33,512)	(33,33,512)
24	Max Pooling	2*2	P=Same	S=1	--------	(33,33,512)	(32,32,512)
25	Dropout = 0.30	-----	--------	--------	--------	(32,32,512)	(32,32,512)
26	Flatten	-----	--------	--------	--------	(32,32,512)	(524288)
27	Dense1	-----	--------	--------	--------	(524288)	(1024)
28	B-Normalization	-----	--------	--------	--------	(1024)	(1024)
29	Dropout=0.50	-----	--------	--------	--------	(1024)	(1024)
30	Dense2	-----	--------	--------	--------	(1024)	(2000)
31	B-Normalization	-----	--------	--------	--------	2000	2000
32	Dropout=0.50	-----	--------	--------	--------	2000	2000
33	Dense3	--------	--------	--------	--------	2000	2

3.1.8 Euclidian Distance-Based Similarity

Euclidean distance method is used for the difference between the input query and database query. The Euclidian equation subtracts one query from another through feature vector and measures the difference, which is mention below.

Where the features are extracted from the input query and database query, in which the first parameter v is the input query vector and the second one database query vector. Distance 'd' is calculated for query videos with database query videos. The greater will be occurring when the d is smaller. When the d values become smaller, the result will be good between two (query video and database video) and hence the results will be greater when claimed. Where features are extracted from query video as well as dataset video and find the most similar video are retrieved from the database, which is near to query video contents and follow the most well-known equation (Euclidean distance measurement) that is most efficient distance measurement equation, which is proved through experimental results (Iqbal et al., 2018).

3.2 Testing and Performance Evaluation

After the successful design of the proposed research work, the performance has been evaluated and tested via the evaluation parameters. These parameters have shown different accuracy with respect to the desired seniors and expected outcomes. It is obligatory to test the proposed work. What if there still exist problems which need to be refined. The following are the parameters for the proposed work that has been used for analysis and evaluation perspectives.

3.2.1 Simulation Tool

The tool used for simulation in the proposed research is MATLAB which stands for MATrix LABoratory. MATLAB is created by math works for simulation and graphical user interface and using for animated types scenarios of the real work objects or scenarios. One of the best features of MATLAB is that it supports a high level of mathematical equations and some numerical statements which involve high math. For image processing, it has admirable features.

3.2.2 Performance Evaluation

Based on the proposed performance evaluation parameters the proposed work has been checked. For evaluation two parameters have been used as a core contribution. The performance evaluations have been thoroughly discussed in Section IV in the form of tabular and graphical representation.

3.2.3 Performance Evaluation Parameters

The given are the performance evaluation parameters that are used for the analysis and accuracy of the proposed research work.

3.2.3.1 Accuracy

Accuracy is a technique to characterize the classifier assessment and figure the presentation of the framework. Precision computes all right expectation perception isolated by the all-out perception number or genuine number, if characterization exactness is higher than the exhibition of the framework is better. Precision is determined by utilizing Equation 1.

$$Accuracy = \frac{TotalCorrectVideos}{TotalRetrievalVideos} \tag{1}$$

3.2.3.2 Loss

The loss method is used to check the learning and testing procedures of profound neural systems. Misfortune is generally utilized in profound learning. Misfortune can be characterized as the normal «blunder» is the distinction between genuine and anticipated qualities.

$$Loss = \frac{TotalFalseVideos}{TotalRetrievalVideos} \tag{2}$$

3.3 Precision and Recall

Our system is evaluated on two metrics and recall. The most popular evaluated methods, these two techniques are used for the content-based video retrieval system. Precision P is calculated for the number of same elements retrieved and the number of elements retrieved from the database; it calculates the accuracy of the retrieval system. Recall R is calculated between the two values. The ratio between the number of correct elements retrieved and the total number of related elements that are presented in the database. Precision P and recall R are calculated by the following equations.

3.4 F-Measure

It calculates precision, mean, and recall calculating score. It supposes both the recall and precision of the system to test the f score. F score is very accurate and balanced recall and precision. It is implicit as a weighted normal of recall and precision. It is defined in Equation 3 as:

$$F_{Score} = \frac{2 * Precision * Recall}{Precision + Recall} \tag{3}$$

3.5 Mean square error

Mean square error (MSE) is a metric that fined the difference between the predicted values from the observed values between 0 and 1 in the analysis. The symbol of sigma in mathematics, the character that looks like E is called summation. That is the summation of all values, from start i=1 till n, through this we add all points from start to an end. For each point, we take the y is the predicted points for the ith observations, and the 'y' is the correct observed values for the ith observation. We subtract the correct observed values from the 'y' predicted values and calculate the square of the result. In the last to put the summation of all the $(y - y')^2$ values, and divide it by n, n is the total number of videos in the database which will give the mean square error (Mathieu et al., 2015).

3.6 Dataset

We are using UCF101; this data set in the current time is a huge data set of human actions. They contain 101 classes of actions and 13k short clips and 27 hours of video data. This data set contains user-uploaded video and cluttered backgrounds. Moreover, this data set provides standard activity acknowledgment results on this new dataset utilizing the standard sack of words approach with the general execution of 44.5%. As far as we could possibly know, UCF101 is at present the most testing

dataset of activities because of its enormous number of classes, countless clasps, and furthermore unconstrained nature of such clasps.

In this section, a thorough overview of the proposed model has been given along with the simulation tool and simulation performance parameters for evaluation and analysis. This section has given a complete idea that how the proposed work has been carried out by using the proposed simulation environment and parameters. In short, this section gave a thorough idea about the research methodology step by step. The research methodology is the core contribution of the proposed research work which is a simulation-based framework for achieving the results from the simulation.

4. Results and Discussion

In this section, a detailed discussion of the simulation results has been discussed. This section is consisting of simulation-based results in which each scenario has been explained with proper justification. These have been portrayed in the form of graphs and tables for better and clear understanding. Also, the evaluations have been debated with the other feature extraction models of the video retrieval system.

4.1 Experimental Setup

The proposed system is evaluated and tested on GeForce NVidia 8 GB dedicated GPU with window 10 operating system is installed. MATLAB is used as a simulation and programming tool which is best suitable for rapid prototyping.

4.2 Dataset

We are using UCF101; in which data is organized in 101 groups, And 13320 videos from 101 categories, which is as of now the biggest dataset of human activities. It comprises 101 activity classes, this informational index comprises over 13k clasps and 27 hours of video information. The database contains sensible client transferred recordings containing camera movement and a jumbled foundation. Moreover, this informational index gives gauge activity acknowledgment results on this new dataset utilizing the standard sack of words approach with the general execution of 44.5%. Supposedly, UCF101 is right now the most testing dataset of activities because of its enormous number of classes, countless clasps, and furthermore unconstrained nature of such clasps.

Table 2. Experimental Setup

Name	**Configuration**
Operating System	Microsoft Windows 10
Simulation Tool	MATLAB
Libraries	Numpy, Time, SciPy, PyLab, Matplotlib, OpenCV
Implementation Environment	TensorFlow Keras
Dataset	UCF101

Figure 5 shows the 101 actions of the UCF101 data set in a single frame. The border color of the frame specifies which action belongs to which category. Like human action interaction etc. and the label on the frame specifies which class the video belongs to or from whom that video belongs.

The average length of the clips for each action is depicted in green.

Figure 6 shows that the total time of videos for each class is using blue bars and the average length for each class in green color. All clips have the same frame rate and resolution 320*240. And all files have the avi formats.

The distribution of clip durations is illustrated by the colors.

The graph in Figure 7 shows the total number of clips in each class. The clips are distributed in colors. The colors in each bar show the duration of different clips in each class.

Figure 5. Visual Results of UCF101 dataset

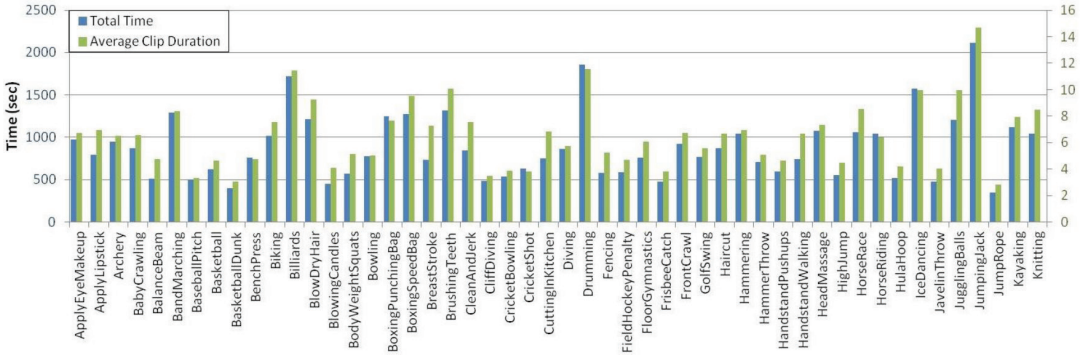

Figure 6. Total time of videos for each class is illustrated using the blue bars

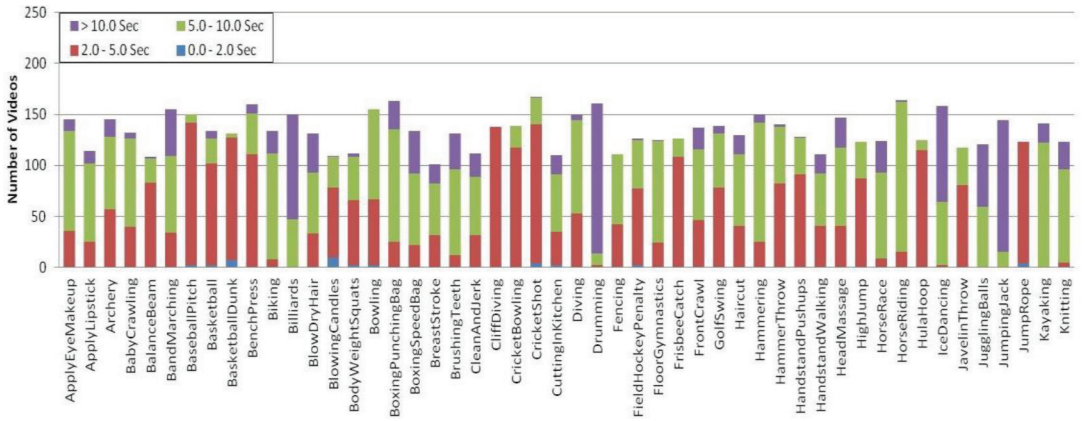

Figure 7. Number of clips per action class

4.3 Evaluation for the Proposed System

In the case of classification problems having only one classification accuracy might not give you the whole picture. So, Accuracy or loss is used to summarize the algorithm performance. Through the calculation of accuracy and loss, we find out where the system is wrong and where the system is right and which type of error coming into the system. An accuracy and loss are used to check the performance of a classification model on a set of test data for which the true values are known.

4.4 Results on UCF-101 Dataset

The UCF-101 dataset is the most testing one, incompletely because of the enormous volume of video and somewhat because of the high level of perspective varieties with which the video is gathered. This isn't basically an observation dataset rather the recordings were gathered to assess normal movement acknowledgment errands. Given an inquiry video, the goal is to recover a video of a similar

movement independent of the perspective varieties, and changes in hues or surfaces, and so on. The recordings in this dataset were taken with shifting foundations, which make it a fine possibility for assessing the reasonableness of our technique. We extricated CNN highlights utilizing AlexNet from these pictures of the recordings and utilized them to recover the top-positioned video. The goal was to recover however many pertinent recordings as could be allowed. Figure 8 contains aftereffects of picked question recordings, where the furthest left video is the inquiry and the remaining are top-positioned video dependent on the separation between the question and the dataset video. Essentially, in inquiry 2, the best 6 pictures have been recovered accurately. In the third inquiry, applicable pictures were recovered at positions 2, 4, 5, 6, and 8, regardless of the way that there exists a tremendous divergence in their experiences. In the remainder of the inquiries, pertinent pictures were recovered at top positions which show the abilities of proposed highlights. Despite the fact that the outcomes on this dataset are not solid, we accept that if an all the more remarkable CNN model is utilized, these outcomes can be significantly improved.

In figure 8 the experimental result on dataset UCF101 shows on one video 1, 2, 3, 4, and 5. It made 6 numbers of clusters for all videos. From the FC6 simulation result, we obtained 44% accuracy and 56% loss. This layer of the AlexNet model gives very poor performance.

Figure 8. Retrieve result of FC6 on UFC101 Dataset

Figure 9. Retrieve result of FC7 on UFC101 Dataset

In Figure 9 the experimental result on videos 1, 2, 3, 4, and 5. It made 6 numbers of clusters for all videos. From the FC7 simulation result, we obtained 60% accuracy and 40% loss. This layer of the AlexNet model gives poor performance. This 20% good than FC6.

In Figure 10 the experimental result on videos 1, 2, 3, 4, and 5. It made 6 numbers of clusters for all videos. From the FC8 simulation result, we obtained 90% accuracy and 10% loss. This layer of the AlexNet model gives poor performance. The proposed method test on the UCF101 data set and give very good results on layer FC8.

4.5 Retrieval Performance of Videos

The system performance is evaluated on the matching of input query and database query by subtracting the input query from the database query. The difference between the two videos found through the Euclidean distance formula. The keyframes are the head of the clusters. These keyframes applying the AlexNet model to extract features. For the performance evaluation, different types of videos are tested. For presentation, a different type of videos is acquired from UCF101 including shorts movies, military documentaries, music videos, and cartoon videos. The video length is 1-2 minutes. Different keyframes are selected from videos and save features in the feature vector. The redundant frames are skipped during the reading of videos. The same videos are searched on the selected keyframes. The experimental result on videos 1, 2, 3, 4, and 5 is divided by the total number of frames 164, 151, 300, 188, and 159 respectively. It made 6 numbers of clusters for all videos. From the FC6 simulation result

Figure 10. Retrieve result of FC8 on UFC101 Dataset

we obtained 60% accuracy and 20% loss from video 1, 60% accuracy and 40% loss form video 2, 60 % accuracy and 40% loss form video 3, 40% accuracy and 10% loss form video 4, 40% accuracy and 60% loss form video 5. The experimental result on videos 1, 2, 3, 4, and 5 is divided by the total number of frames 164, 151, 300, 188, and 159 respectively. It made 6 numbers of clusters for all videos. From the FC7 simulation result we obtained 60% accuracy and 40% loss from video 1, 80% accuracy and 20% loss form video 2, 80% accuracy and 20% loss form video 3, 60% accuracy and 40% loss form video 4, 80% accuracy and 20% loss form video 5. The experimental results on videos 1, 2, 3, 4, and 5 are divided by the total number of frames 164, 151, 300, 188, and 159 respectively. It made 6 numbers of clusters for each video. From the FC8 simulation results, we have obtained 80% accuracy and 20% loss from video 1. Similarly, 90% accuracy and 10% loss from video 2. 90% accuracy, and 10% loss from video 3. 90% accuracy, and 10% loss from video 4. And 90% accuracy and 10% loss from video 5. Table 3 is regarding the accuracy comparison of FC8 with FC7 and FC.

The results of the proposed method on dataset UCF101 are shown in Figure 11. The retrieving of the same vides from different colors and having the same structure. Some of the different videos are also collected from the database. This is due to the same nature of th4e videos that are similar to the input query.

In Figure 11 shows the results of all three layers in the term of accuracy. We have evaluated the precession and recall. We have got 40% accuracy on fc6, and 60% accuracy on fc7, and 90% accuracy

Table 3. Accuracy Comparison of FC8 with FC7 and FC6

Model Name	Video_1 Accuracy in %age	Video_2 Accuracy in %age	Video_3 Accuracy in %age	Video_4 Accuracy in %age	Video_5 Accuracy in %age	Average Accuracy
FC6	40	40	60	40	40	44%
FC7	60	80	80	60	80	72%
FC8	80	90	90	90	90	90%

Note: The value of 44% in Table 3 is the model accuracy which is FC6 and 72% is the accuracy of FC7 whereas, 90% is the accuracy in average accuracy of the proposed model named FC8. In this sense, the FC8 is the proposed model which has given the highest accuracy as compared to FC6 and FC7.

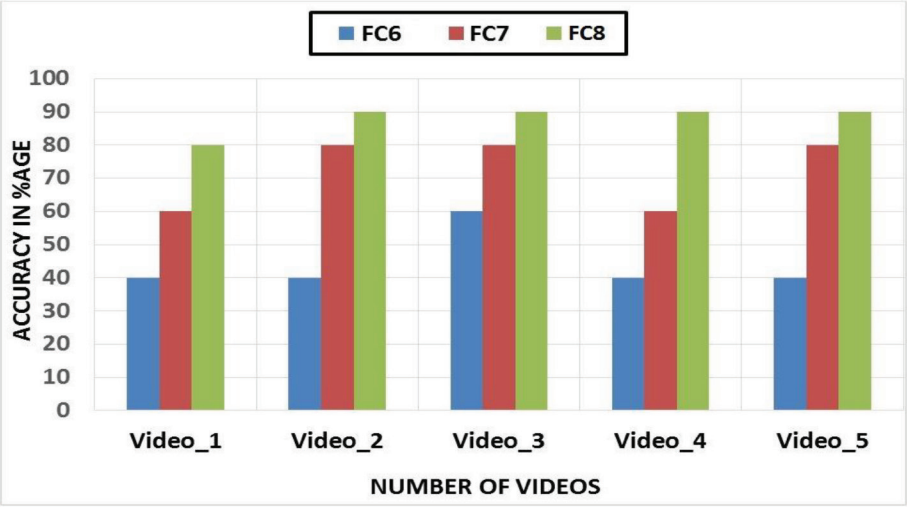

Figure 11. Comparison of FC8 (the proposed model) with FC7 and FC6

Note: The Figure 11 actually illustrates the overall accuracy in percentage of the proposed model FC8 in comparison with the FC6 and FC7. 5 videos have been executed in which each accuracy has been generated from which the proposed model FC8 has given highest accuracy in all scenarios.

on Fc8. The accuracy graph Figure 4.8, shows that the retrieval performance of our proposed method is better than the other state-of-the-art methods. Hence, it is clear from these results that the proposed algorithm outclasses existing techniques by a significant margin.

4.6 Loss of Model layers in terms of MSE

Mean Square Error (MSE) is a metric that fined the difference between the predicted values from the observed values between 0 and 1 in the analysis. The symbol of sigma in mathematics, the character that looks like E is called summation. That is the summation of all values, from start i=1 till n, through this we add all points from start to an end. For each point, we take the y is the predicted points for the ith observations, and the y' is the correct observed values for the i^{th} observation. We subtract the correct observed values from the y predicted values, and calculate the square of the result. In the last

to put the summation of all the $(y-y')^2$ values, and divide it by n, n is the total number of videos in the database which will give the mean square error. In table 4 shows the results of the mean square error of layers fc6, fc7, and fc8 with corresponding videos.

4.7 Precision and Recall

Our system is evaluated on two metrics and recall. The most popular evaluated methods, these two techniques are used for the content-based video retrieval system. Precision P is calculated for the number of same elements retrieved and the number of elements retrieved from the database; it calculates the accuracy of the retrieval system. Recall R is calculated between the two values. The ratio between the number of correct elements retrieved and the total number of related elements that are presented in the database.

4.8 F-Measure

Through the f score, we calculate precision and recall measuring scores. It calculates more accurately the results of precision and recall. It defines the ratio between precision into recall and precision plus recall into two. F-score is a more accurate and balanced value of recall and precision. After the calculation of the f-score, we get the values which are given in Table 5.

Table 5 shows the simulation results of three layers in terms of precision, recall, and F-score. In the above table, every layer is five videos such as 1, 2, 3, 4, and 5 each video are precision, Recall, and F-Score values of every layer. On five videos fc6, fc7, and fc8 give the above mention values in the

Table 4. MSE of Layers FC6, FC7, and FC8

Model Layer	**MSE**
FC6	0.7
FC7	0.6
FC8	0.2

Note: The value of MSE is actually the error in which the lowest error value denotes that the model has performed well. In Table 4, the values of MSE are illustrated in which the FC6 has 0.7, FC7 has 0.6 and FC8 has the MSE of 0.2. This means that the lowest error ratio is shown by FC8 which is better as compared to other FC6 and FC7.

Mean square error towards 1 is means a high error in the model while 0 gives a low error. In the given table fc6 gives 0.7 error its mean high error. Fc7 and give 0.6 errors which are one point low than fc6. And fc8 gives 0.2 errors its mean give a very low error. With the state of the art techniques, the fc8 results are very good in performance.

Table 5. Precision, Recall, and F score evaluation comparison of three layers

	FC6			**FC7**			**FC8**		
Video	**Precision**	**Recall**	**F score**	**precision**	**Recall**	**F score**	**precision**	**Recall**	**F score**
1	0.40	0.20	0.16	0.60	0.30	0.40	0.80	0.40	0.53
2	0.40	0.20	0.16	0.60	0.30	0.40	1.00	0.50	0.76
3	0.60	0.30	0.40	0.60	0.30	0.40	1.00	0.50	0.76
4	0.60	0.30	0.40	0.80	0.40	0.53	1.00	0.50	0.76
5	0.60	0.30	0.40	0.80	0.40	0.53	1.00	0.50	0.76

table; the Fc6 layer obtained precision value 0.60, Recall 0.30, and F-Score 0.40. Similarly, the fc7 layer obtained precision value 0.80; Recall value 0.40 and F-score values are 0.53 respectively, while Fc8 obtained precision values 1.00, Recall values 0.50, F-score values are 0.76. Furthermore, the proposed Alex Net Fc8 layer achieved the highest values of precision, Recall, and F-Score.it is observed that the precision value of the proposed AlexNet Fc8 layer is better than other Fc6 and fc7 layers.

Furthermore, in figure 4.10 the proposed Alex Net Fc8 layer achieved the highest values of precision, Recall, and F-Score.it is observed that the precision value of the proposed AlexNet Fc8 layer is better than other Fc6 and fc7 layers. Because fc6 and fc7 give low precision and recall values and fc8 get a high value towards 1. When precision and recall value near one it shows that the system performance is better. Figure 12 is regarding the performance of proposed AlexNet layer-based Confusion matrix.

4.9 Time Complexity Performance

The proposed system is evaluated and tested on GeForce NVidia 8 GB dedicated GPU with Windows 10 operating system is installed. MATLAB is used as a simulation and programming tool which is best suitable for rapid prototyping.

Figure 13 shows the time of videos taken on GPU testing. From the first video checked total numbers of frames are 164 and take six frames as keyframes and takes a total time of 1186 seconds. From the second video checked total numbers of frames are 151 and take six frames as keyframes and takes a total time of 1120 seconds. From the third video checked total numbers of frames are 188 and take

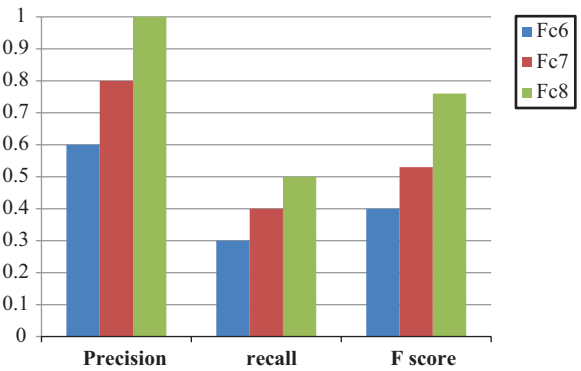

Figure 12. performance of proposed Alex net layer-based Confusion matrix

Table 6. Time Complexity.

Video No	Total Number of Frames	Number of cluster Frames	Time on GPU
1	164	6	1186 sec
2	151	6	1120 sec
3	300	6	1471 sec
4	188	6	1232 sec
5	159	6	1594 sec

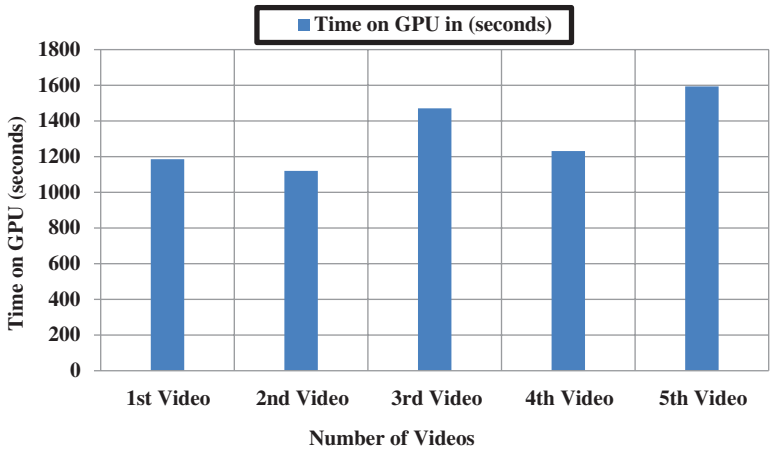

Figure 13. Performance of Proposed Method on GPU

six frames as keyframes and takes a total time of 1471 seconds. From the fourth video checked total numbers of frames are 188 and take six frames as keyframes and takes a total time of 1232 seconds. From the fifth video checked total numbers of frames are 159 and take six frames as keyframes and takes a total time of 1594 seconds.

4.10 Training and testing of the proposed system

Training and testing is the major and key aspect in the CBVR system in which the real-time concept is used for the requirements of the recently taken datasets to be generated the needy output. This system is especially and particularly utilized in YouTube/Dailymotion etc. for the last two decades in which many related videos are displayed because of the contents in the queried videos. This is actually the activity of the searched video which results to display several related videos in which the same not exact copy but somehow similar activities are taking place. First of all the data are taken and then trained for testing but it is a broad term. In some systems, the data are stored in their databases i.e., pre-stored data in which the feature models do exist. In the new FV (Feature Vector) system the features are saved of the dataset and then this is tested against the mentioned dataset. Now, if we take a dataset that will need training and then that new dataset will be taken accordingly. These two datasets will be tested against each other in the current scenario for further processing. In other words, the new data will be taken and then tested in an efficient and effective manner with respect to the desired task. The proposed CBVR is a real-time system which needs no pre-data storage or taking data from the database. This system is actually taken the dataset as input which is trained and then it is tested. It can take new queries for that data to be processed. The data is tested against the recent taken data due to the real-time nature of the proposed system. The new queries are used for searching for the data that is taken as input. The queries are actually used for searching in the existing database to find train/test the required data. Due to this nature of real-time, there we've not taken training & testing. The most and key factor by not taking the data for training and testing especially for storing is that this is a REALTIME system in which only the recent data are processed and it generates the required output/result from that data. Table 7 is regarding the Illustration of FC6, FC7 & FC8 with Units, Weights, and Connections.

Table 7. Illustration of FC6, FC7 & FC8 with Units, Weights, and Connections

Layer	Units	Weights	Connections
FC6	1000	4,096,000	4,096,000
FC7	4096	16,777,216	16,777,216
FC8	4096	67,108,864	67,108,864
		Total	87,982,080

4.11 Evaluation & Justification of FC6 and FC7 with FC8 Layer

FC stands for Fully Connected in the given expression which is a layer in the CNN for feature extraction providing an appropriate arrangement for segregation and classification process of images and other associated contents in the videos etc. FC6, FC7, and FC8 are layers that have diverse features in contrast with one another that is achieved by the required task. In short, these are the levels of fully connected layers as mentioned 6, 7, and 8 which denote that each level has an improvement as compared to the second last one. FC7 is an improved and extension of FC6 and FC8 is an improved and extension of FC7. In the FC6 layer, there exist units which connect it with additional features to be carried out. For the fully connected layers, let N_i, P_i, and U_i be the number of output units, parameters (weights), and connections of layer L_i.

FC6 Layer:
N_6=1000 units. $P_6 = U_6 = 4096*1000 = 4{,}096{,}000$.
FC7 Layer:
$N_7 = 4096$ units. $P_7 = U_7 = 4096*4096 = 16{,}777{,}216$.
FC8 Layer:
N8 = 4096 units. (Max pooling: input 13/2 = 6) $N_8 = 4096$; $P_8 = U_8 = 8*8*256*4096 = 67{,}108{,}864$.

Notice that the number of parameters is much larger for these layers than the convolutional ones.

Overall, AlexNet has about 660K units, 61M parameters, and over 600M connections.

Notice: the convolutional layers comprise most of the units and connections, but the fully connected layers are responsible for most of the weights. More modern networks can do better with fewer parameters (e.g., GoogLeNet).

From the above explanation and justification, it has been revealed that FC8 Layer is much more effective than the other FC6 and FC7 Layers. Due to a high number of units, weights and connections FC8 layer has high accuracy and low loss and therefore this layer has given the best results in the entire scenario. After a successful simulation from MATLAB, the proposed model AlexNet FC8 has shown outstanding performance in contrast with FC6 and FC7. The accuracy of FC8 is greater in percentage as compared to FC6 and FC7. Furthermore, the loss of video retrieval is also much less as compared to the existing models. The proposed model FC8 has shown the best results which are considered the excessively best among other models as mentioned.

5. Conclusion

In this work, we have proposed a new technique through which we extract the feature of contents from videos. For video retrieval, the color histogram of the video is stored in the feature vector and

calculating the color histogram of selected frames. Keyframes are selected through the k-mean algorithm. From the keyframes, we are extracting color histogram features through the AlexNet model of CNN. The proposed approach is fundamentally very powerful because of the less consumption of time. The results of the system calculated through the Euclidean distance equation. The value of input query subtracting from the database query. The lowest values become the topmost result. The performance was evaluated on the equation of accuracy and loss. While the recent methods have the ability to process millions of videos in a very short time duration. The contribution is the core achievement in research which denotes the role and character in the desired field of study. In this research, the main contribution is the implementation of the color histogram and AlexNet that were not implemented with the existing work. To the best of our knowledge, this is the first attempt that has been conducted on video retrieval systems for the extraction of features from video files rather than text files. By applying color histogram and AlexNet the proposed work have given above 90% accuracy which is considered outstanding in contrast with the existing state of the art solutions. Another main and key contribution is the evaluation based on the performance parameters for improvement in the existing work by modifying and generating the alternative results and in the end, these were then compared with proper justification.

Color-Based approaches have been suggested in this research but the enhancement of content-based video retrieval and video organization processes still has a very long research area. A content-based video retrieval system has more gaps for research. Some suggestions for a future research study are being provided in the field of CBVR.

1. The framework we used in the selected technique for the analysis of the color histogram may be increased in different directions or on a different scale to get more accuracy in the CBVR.
2. The color histogram features are very light and sample, in the future, better feature analysis techniques can be used to get better results.
3. A good method of similarity measure can be used to get more accuracy because the Euclidean distance is a very easy method for similarity measures.

Conflicts of Interest

Authors declare no conflict of interest.

References

Asha, D., Y. M. Latha, and V. S. K. Reddy. 2018. Content based video retrieval system using multiple features. Int. J. Pure Appl. Math.(IJPAM), 118(14), 287-294.

Bolettieri, P., F.Carrara, F. Debole, F. Falchi, C. Gennaro, L. Vadicamo, and C. Vairo, 2019. An image retrieval system for video. In International Conference on Similarity Search and Applications (pp. 332-339). Springer, Cham.

Chen, X., Y. Zhang, Q. Ai, H. Xu, J. Yan, and Z. Qin. 2017. Personalized key frame recommendation. In Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 315-324). ACM.

El Ouadrhiri, A. A., E. M. Saoudi, S. J. Andaloussi, O. Ouchetto, and A. Sekkaki. 2017. Content based video retrieval based on bounded coordinate of motion histogram. In 2017 4th International Conference on Control, Decision and Information Technologies (CoDIT) (pp. 0573-0578). IEEE.

Han, S, J, Pool, J. Tran and W.Dally, 2015. Learning both weights and connections for efficient neural network. In Advances in neural information processing systems (pp. 1135-1143).

Iqbal, S., A. N. Qureshi, and A. M. Lodhi. 2018. Content Based Video Retrieval Using Convolutional Neural Network. In Proceedings of SAI Intelligent Systems Conference (pp. 170-186). Springer, Cham.

Jones, S., and L. Shao. 2013. Content-based retrieval of human actions from realistic video databases. Information Sciences, 236, 56-65.

Lingam, K. M., and V. S. K. Reddy. 2019. Key Frame Extraction Using Content Relative Thresholding Technique for Video Retrieval. In Soft Computing and Signal Processing (pp. 811-820). Springer, Singapore.

Mathieu, M. C. Couprie, and Y. LeCun. 2015. Deep multi-scale video prediction beyond mean square error. arXiv preprint arXiv: 1511.05440.

Rossetto, L., I. Giangreco, H. Schuldt, S. Dupont, O. Seddati, M. Sezgin, and Y. Sahillioğlu. 2015. IMOTION—a content-based video retrieval engine. In International Conference on Multimedia Modeling (pp. 255-260). Springer, Cham.

Sedighi, V., and J. Fridrich. 2017. Histogram layer, moving convolutional neural networks towards feature-based steganalysis. Electronic Imaging, 2017(7), 50-55.

Sikos, L. F. 2018. Ontology-based structured video annotation for content-based video retrieval via spatiotemporal reasoning. In Bridging the Semantic Gap in Image and Video Analysis (pp. 97-122). Springer, Cham.

Song, J., H. Zhang, X. Li, L. Gao, M. Wang, and R. Hong. 2018. Self-supervised video hashing with hierarchical binary auto-encoder. IEEE Transactions on Image Processing, 27(7), 3210-3221.

Thanh, T. M.,P.T Hiep, T.M Tam and K.Tanaka, 2014. Robust semi-blind video watermarking based on frame-patch matching. AEU-International Journal of Electronics and Communications, 68(10), 1007-1015.

Tarigan, J. T., and E. P. Marpaung. 2018. Implementing Content Based Video Retrieval Using Speeded-Up Robust Features. International Journal of Simulation–Systems, Science & Technology, 19(3).

Zhang, C., Y. Lin, L. Zhu, A. Liu, Z. Zhang, and F. Huang, 2019. CNN-VWII: An efficient approach for large-scale video retrieval by image queries. Pattern Recognition Letters, 123, 82-88.

Author's Biography

Altaf Hussain received his MS and BS Degrees in Computer Science from The University of Agriculture Peshawar, Pakistan (2017) and University of Peshawar, Pakistan (2013), respectively. He worked at The University of Agriculture as a Student Research Scholar from 2017-2019. During his MS Degree he has completed his research in Computer Networks especially in Routing Protocols in Drone Networks. His recent approach is PhD in Computer Science & Technology. He has served as a Lecturer in Computer Science Department in Govt Degree College Lal Qilla Dir L, KPK Pakistan from 2020-2021. He has published many research papers including survey/review and conference papers. He was Research Scholar in (Career Dynamics Research Academy) Peshawar, Pakistan for one and a half year. Currently he is working as a Research Assistant in the Department of Accounting & Information Systems, College of Business and Economics, Qatar University. His research specialties/interests include Wireless Networks, Sensor Networks, and Unmanned Aerial Vehicular Networks.

Mehtab Ahmad received master's degree in Computer Science in 2020 from Institute of Computer Science & IT (ICS/IT), The University of Agriculture Peshawar, Pakistan. He worked at The University of Agriculture as a Student Research Scholar from 2017-2019. During his MS Degree he has completed his research in Deep Learning with AlexNet especially in Content based video retrieval. His recent approach is PhD. His research interest includes Wireless Computing, Deep Learning, Image Processing, and Big Data.

Tariq Hussain received his MS and BS Degrees in Information Technology from the Institute of Computer Sciences and Information Technology The University of Agriculture Peshawar, Pakistan (2019), and University of Malakand, Pakistan (2015) respectively. He has published many research papers in the area of Computer Networks. Currently he is a Doctoral Student at School of Computer Science and Information Engineering, Zhejiang Gongshang University China. His research interests are the Internet of Things, 3D Point Cloud, Big Data, Data analytics and AI.

Ijaz Ullah is working as a visiting Lecturer with the Department of Computer Science, University of Swabi, Swabi Pakistan. He has completed his Bachelor degree in Computer Science from University of Peshawar, Pakistan and Master Degree from EIT Digital Master School. EIT Digital Master School provides dual degrees. So, he got Master in Cloud Computing from University of Rennese 1 France and ICT Innovation degree from Technical University of Berlin Germany. His research interest includes Cloud Computing and Networks.

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal
Regular Issue, Vol. 11 N. 2 (2022), 237-248
eISSN: 2255-2863
DOI: https://doi.org/10.14201/adcaij.27777

Comparative Evaluation of Techniques for n-way Stream Joins in Wireless Sensor Networks

Djail Boubekeur

LCSI Department, High School of Computer Science (ESI), Algiers 16000
b_djail@esi.dz

KEYWORDS

wireless sensor networks; communication cost; in-network join; n-way join

ABSTRACT

In wireless sensor networks, sensor data are accessed using relational queries. Join queries are commonly used to retrieve the data from multiple tables stored in different parts of a wireless sensor network. However, such queries require large amounts of energy. Many studies have intended to reduce query energy consumption. However, most of the proposed techniques addressed binary joins which are performed between static tables. N-way joins between data streams were rarely considered. Join queries using data streams work continuously and require increasing energy, which is why n-way joins involving several tables consume so much energy. Thus, the challenge lies in reducing energy dissipation. Additionally, it is necessary to determine the appropriate execution order for an n-way join. The number of possible implementations of an n-way join grows exponentially with the tables' number. In this paper, interesting approaches for n-way joins between streams of data are evaluated. The methods that have been compared are extern-join, Sens-join of Stern et al, and the two techniques NSLJ (N-way Stream Local Join) and NSLSJ (N-way Stream Local Semi-Join). Comparisons are conducted according to several parameters to determine which use case is appropriate for each technique. NSLSJ works best for join queries with low join selectivity factors, while extern-join is more suitable for queries with very high selectivity factors.

1. Introduction

A wireless sensor network regroups a set of nodes, where each node corresponds to a sensor. Sensors have limited memory and computing capabilities.

Sensors detect events and save corresponding data. Data records at each node form a dataset. The data of all nodes make up a distributed database table. Access to data is performed by relational queries. Joins are relational queries widely used in wireless sensor networks, such as in the following applications: vehicle surveillance and tracking, animal habitat monitoring, environment monitoring, home and commercial building automation, precision agriculture, and water resource management (Kang, 2013).

A join query consists of assembling data from many nodes of the same network. This query type requires higher energy consumption. Since the nodes have limited energy, join execution might cause the failure of each node and the entire network. With n-way join queries in wireless sensor networks, the challenge lies in significantly reducing energy consumption. Additionally, it is necessary to determine the most appropriate order for the execution of the query. Note that the number of order possibilities grows exponentially with the number of tables being considered.

Several techniques have been proposed to treat binary joins and joins between static tables. Established solutions consist in reducing the number of messages that are sent. Thus, it was confirmed that energy consumption is higher during the transmission of messages than it is during data processing at the nodes. For continuous n-way joins, several studies have been carried out, such as SENS-join (Stern, Buchmann, and Böhm, 2009) of Stern et al., NSLJ (N-way Stream Local Join) (Djail, Hidouci, and Loudini, 2016) and NSLSJ (N-way Stream Local Semi-Join)(Djail, Hidouci, and Loudini, 2019). SENS-join determines a filter at the sink, then communicates it to interval nodes to recuperate joinable tuples. The final result is calculated at the base station. NSLJ performs the join query at internal nodes, without using filters. NSLSJ improves NSLJ by adopting semi-join to filter the non-joinable tuples.

The remainder of the paper is organized as follows: Section 2 summarizes the main features of join queries in wireless sensor networks. Section 3 describes related work. Section 4 presents the principles of the techniques selected for comparison. Section 5 discusses the conducted tests and the obtained results. Finally, the last section concludes the paper.

2. Join Features in Wireless Sensor Networks

2.1. Definitions

A join between two tables R and S is the table that contains tuples of R matched tuples of S according to a condition called: join predicate. When the join is performed between two tables, it is referred to as a binary join. If more than two tables are considered, the join is an n-way join.

A join with join predicates that use arbitrary comparison operators defines a theta-join. An equi-join uses only the equality operator in its join predicate.

2.2. Implementation of Join Queries in Wireless Sensor Networks

There are mainly two implementations of join queries in wireless sensor networks: extern and in-network join executions. (Kang, 2013)

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal
Regular Issue, Vol. 11 N. 2 (2022), 237-248
eISSN: 2255-2863 - https://adcaij.usal.es
Ediciones Universidad de Salamanca - CC BY-NC-ND

Extern-join executes the join query entirely at the sink. The tuples of designed tables must be sent to the station base. It is the easiest to implement, but also the most energy-intensive.

In-network implementation decreases consumed energy considerably by reducing the number of transmitted messages by performing the query at the internal nodes (Kang, 2013).

2.3. Join Types in Wireless Sensors Network

Considering the spatial aspect, the joins in the wireless sensor networks can be divided into two large classes: unique-region joins and inter-region joins. (Kang, 2013)

A unique-region join is performed between nodes tuples of the same region, while, an inter-region join (Figure 1) is executed between nodes tuples of two distinct regions.

Considering the temporal aspect, join queries in wireless sensor networks can be split into two categories: One-shot joins and continuous joins.

With one-shot joins, a fixed window is defined for each table. It corresponds to a tuples number or to a time period. Query joins are performed on windows as a single execution using static tables.

With continuous joins, sliding windows are used to permit an ongoing execution of the query. A continuous join that is performed repeatedly at periodic intervals is called periodic join.

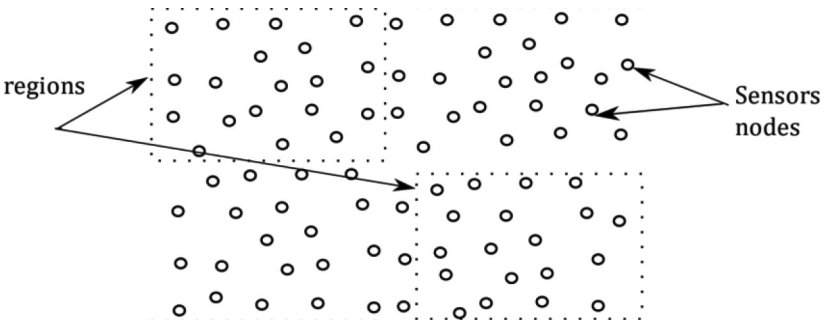

Figure 1. Inter-region join principle

3. Related Works

Join queries have been addressed by many studies. Thus, the proposed techniques can be divided into two large categories: techniques with filtering of non-joinable tuples and techniques without filtering.

Joins without filtering are performed by considering all the tuples of the tuples. Thus, no filter was generated or applied. Yao and Gehrke (Yao and Gehrke, 2003) compared an extern join to an in-network join in terms of communication costs. They concluded that the in-network technique permits less energy consumption for a low join selectivity. Bonfils and Bonnet (Bonfils and Bonnet, 2004) assessed the optimal node for an in-network join. The site situated on the shortest path between the two nodes participating in the query is the winner. Coman et al. (Coman, Nascimento, and Sander, 2007) presented local join and mediated join techniques for treating an inter-region join. Local join performs the query locally at nodes of one of the two regions. However, mediated join executes the query at an intermediate region. It has resulted in that no specific join strategy has the best performances for all queries.

The techniques with filtering adopt various principles, such as semi-join, to filter non-joinable tuples. These techniques permit a considerable gain of energy and are most recently used. Yu et al. (Yu, Lim, and Zhang, 2006) presented Synopsis Join to deal with a one-shot inter-region join. They adopt a distributed alternative of the semi-join approach to decrease tables' sizes. Coman et al. (Coman et al., 2007) proposed Local Semi-Join technique based on the semi-join principle. The join operation is executed in one of the two regions. Min et al (Min, Yang, and Chung, 2011) suggested various plans to perform a join query and they developed a cost model to select the optimal plan under various conditions.

Other authors addressed specific join queries. Mo et al. (Mo, Fan, Li, and Wang, 2014) treated spatial queries in a wireless sensor network. Kang et al (Kang, 2015) addressed the iceberg join query, a special type of join where only tuples whose cardinality exceeds a certain threshold are accepted to the join operation. Min et al. (Min, Kim, and Shim, 2014) presented a solution-based time-windowed principle to treat continuous joins.

These techniques were mostly proposed for binary joins. Few studies addressed n-way join and joins between data streams. Stern et al. in (Stern et al., 2009) suggested a strategy to treat all join types, including n-way join queries. The strategy consists in performing the join at the sink, by using filters that are determined at internal nodes based on the relevant records. Djail et al. proposed NLJ (N-way Local Join) (Djail, Hidouci, and Loudini, 2015), NLSJ (N-way Local Semi Join) (Djail et al., 2016) and NMSJ (N-way Mediated Semi Join) (Djail, Hidouci, and Loudini, 2018) (Djail, Hidouci, and Loudini, 2020) techniques to address n-way join between static tables and NSLJ (Boubekeur, Khaled, and Malik, 2018) and NSLSJ (Djail et al., 2019) to treat n-way join between data streams. NLSJ and NSLSJ apply semi-join to filter non-joinable tuples.

4. The Evaluated Techniques

This study evaluates four approaches to treating n-way joins between streams in wireless sensor networks. Among them are the extern join for data streams, the Sens-join developed by Stern et al., and the NSLJ and NSLSJ proposed by Djail et al. The evaluation is based on estimating the communication cost of each research technique. The communication cost corresponds to the number of transmitted messages. These four techniques are described here:

4.1. N-way Stream Extern Join Technique

Extern joins are used to join data streams at the base station. Periodically, different sites transmit sets of tuples to the sink. For each set reception, a join query is executed (Kang, 2013) (Figure 2).

The extern join technique is simple to use, but its transmission cost is extremely high.

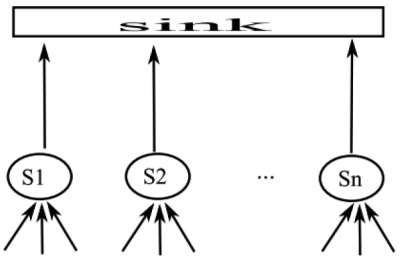

Figure 2. Extern join execution

4.2. N-way Stream Local Join Technique

N-way Stream Local Join (NSLJ) (Djail et al., 2019) treats n-way join queries between data streams in wireless sensor networks. NSLJ does not adopt tuples filtering. It executes intermediate joins for an n-way stream join at in-network by selecting the destination node to do this operation.

NSLJ uses the left linear trees technique to choose the best execution order of the join operations (Steinbrunn, Moerkotte, and Kemper, 1993). This choice is guided by knowing geographical zone positions to select the nearest region as the next destination node.

Additionally, NSLJ adopts the principle of the technique 'distributed join processed at destination node' proposed in (Tran and Lee, 2010) for classical distributed systems.

NSLJ technique runs in three phases:

Phase 1. Query dissemination

Initially, the query is first generated at the base station. Then, it is transmitted to the specified regions. A location routing protocol such GPSR (Karp and Kung, 2000) is used to ensure that the query is correctly received at root nodes of regions. A root node is a principal node at a region organized in tree, where it is assumed that each node knows its location and the locations of its neighbors, via GPS or via localization algorithms (Savvides, Srivastava, Girod, and Estrin, 2004).

Phase 2. Query execution

An n-way join executes several operations. At each operation, an intermediate join is performed. An operation is realized between a pair of nodes. Node pairs are determined on the basis of a technique of left linear trees which, at each step, select the next node to participate in the following operation.

Assuming a node pair (S_i, S_{i+1}), NSLJ performs the following (Figure 3):

- A set of tuples (B_{11}) is transmitted from a site Si to a site S_{i+1}.
- An intermediate join is performed between B_{11} and the window W_{22} which is maintained at the site S_{i+1}.

The result of an intermediate join is communicated to the next node, which repeats the same steps until the final result is determined at the site

Phase 3. Final result transmission

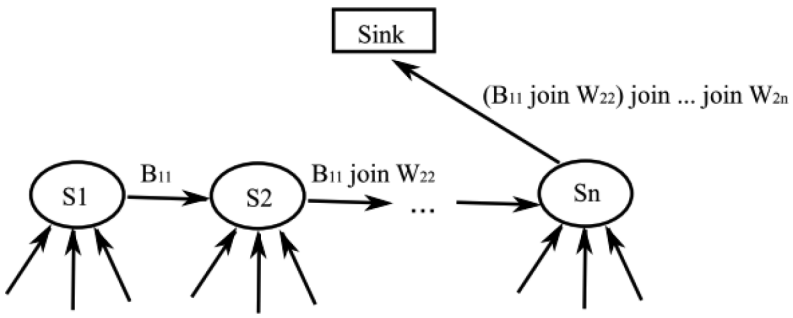

Figure 3. N-way Stream Local Join (SNLJ) execution

Once the intermediate joins have been completed, the final result is communicated to the base station.

4.3. N-way Stream Local Semi-join Technique

N-way Stream Local Semi-Join (NSLSJ) (Djail et al., 2019) is a filtering technique that uses the semi-join principle to filter tuples and improve the performances of the latest proposed technique NSLJ (Boubekeur et al., 2018) and NMSJ (Djail et al., 2018).

NSLSJ performs a join query in three phases:

Phase 1. Query dissemination

In the same way as the NSLJ technique described before, this phase is performed. The query is communicated to root nodes using a location protocol such as GPSR.

Phase 2. Query execution

Each root node Si maintains a window, designated W2i, that contains the tuples received from its region. A set of tuples, noted B1i, represents the tuples transmitted periodically by the site Si. K1i characterizes the projection of B1i on the join attributes.

For a pair of nodes (S_i , S_{i+1}), an intermediate join is executed as follows (Figure 4):

i. A projection K_{1i+1} is transmitted from S_{i+1} to Si.
ii. A semi-join is performed, at S_i, between w_{2i} and K_{1i+1}.
iii. The result $W_{2i'}$ is transmitted to S_{i+1}.
iv. The final result of the intermediate join is determinate at S_{i+1}.

Phase 3. Final result transmission

With the last intermediate join executed, the determined result is communicated to the sink.

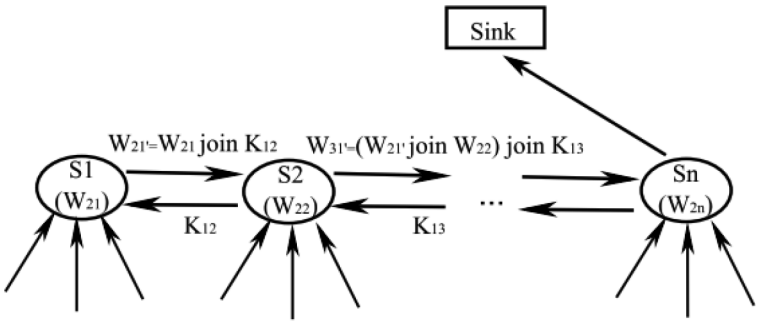

Figure 4. N-way Stream local Semi-Join (NSLSJ) execution

4.4. Sens-join for Data Streams

Sens-join (Stern et al., 2009) is the technique proposed by Stern et al. to treat all types of join queries in wireless sensor networks. Sens-join has five phases:

Phase 1. Query dissemination

A query is initially diffused from the sink to all specified root nodes.

Phase 2: Join attributes transmission

The join attributes are communicated by root nodes to the sink (Figure 5). The aim is to fix the filter of the join query.

Phase 3: Filter determination

The sink produces a filter that is then directed to root nodes in order to perform the semi-join operation.

Phase 4: Semi-join accomplishment

The root nodes perform the semi-join operation. The result is then transmitted to the sink.

Phase 5: Final execution

At the base station, the final result of a join is determined according to the results that have been received from root nodes.

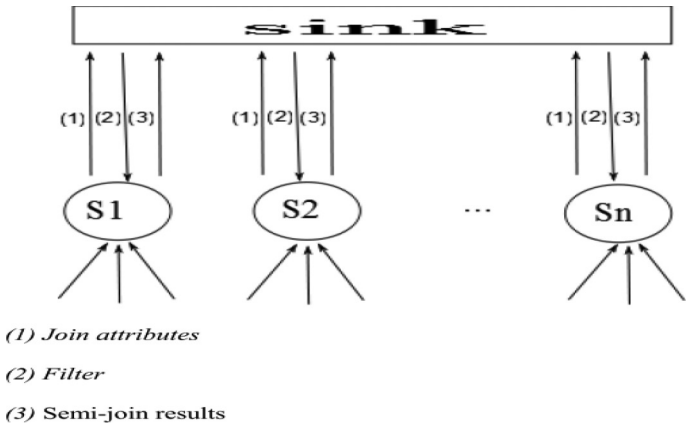

Figure 5. SENS-join execution for n-way stream

5. Experimentation and Performance Analysis

5.1. General Description

A comparative study of the four techniques has been conducted with the aim of determining the technique that performs best and under what conditions of experimentation. This experiment, we consider continuous inter-region joins with the syntax as follows:

SELECT S1.attributs, S2.attributs,…,Sn.attributs

FROM S1, S2, …, Sn

WHERE predicat(S1) AND predicat(S2) … AND predicat(Sn)

AND join-exp (S1.join-attributs , S2.join-attributs ,…, Sn.join-attributs)

where:

Si is the stream of the i^{th} region.

predicat (Si) is a selection predicate of the stream Si.

join-exp is the join condition.

In this experiment, the example of vehicle traffic control across a variety of geographic areas has been used. For three regions, we write:

SELECT Veh1.VId, Veh1.time, Veh2.time, Veh3.time

FROM Veh1, Veh2, Veh3

WHERE (Veh1.time IN r1) and (Veh2.time in r2) and (Veh3.time in r3) and (Veh1.VId = Veh2. VId) and (Veh2. VId= Veh3. VId)

where:
r1, r2, and r3 indicate time ranges during which the Vehicles passed respectively through zones 1,2 and 3.

5.2. Experimentation Environment

In this evaluation, the NS3 simulator is used to simulate an n-way join query. The techniques described above were evaluated by assuming this follows:

- Tuple size is 40 bytes.
- Message size is 40 bytes.
- A Column is 10 bytes.
- The result tuple size is 30 bytes.

As a principal measure, the communication cost was evaluated according to multiple parameters during the actual tests:

- Selectivity factor
- Number of streams
- Size of a stream window.

Selectivity factor values are considered in two intervals: one for low values [10^{-5} , 10^{-4}] and another for high values [10^{-4} , 10^{-3}]. All values are generated randomly. An average value is determined between those engendered for all intermediate joins. A simulation is realized for three streams and then for five streams. The size of stream windows is assumed to equal to 900 tuples.

The number of streams considered in the second evaluation is selected between two and seven. Two evaluations were performed: one with a selectivity factor value in the lowest interval and equal to 0.000025, other with a selectivity factor value in the high interval and equal to 0.00025.

For the third evaluation, stream window sizes are selected between 700 and 1100 tuples. Two simulations were performed with two distinct values of selectivity factor: 0.000025 and 0.00025.

6. Impact of Join Selectivity Factors

In all the performed tests, NSLSJ has performed better than all the other techniques (Figures 6-9). The success of NSLSJ lies in its use of the semi-join principle to realize tuples filtering.

Given specific conditions, the NSLJ technique executes better than SENS-join.

Five streams with low values of selectivity factor up to the value 0.00025, NSLSJ returns low-cost communication.

The performance of Sens-join technique is close to that of NSLSJ. Sens-join is a filtering technique, which is based on the use of filters to remove non-joinable tuples before query execution. The disadvantage of Sens-join is that the query is performed at the sink, but not at the internal nodes of the network.

Extern-join introduces poor results in the tested intervals, but the cost communication remains constant for all the values of all intervals. It has been confirmed that extern-join is attracted in use with very high selectivity factors.

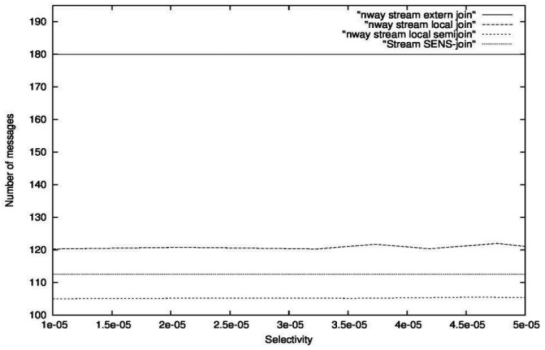

Figure 6. Communication cost for 3 streams in the interval [10^{-5}, 10^{-4}]

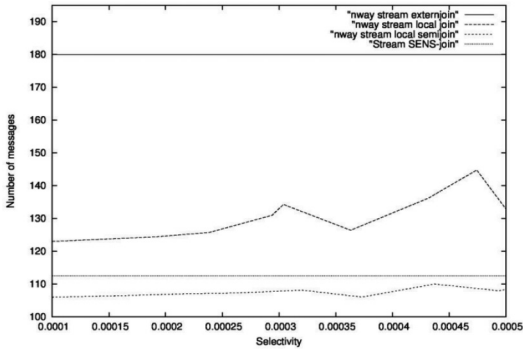

Figure 7. Communication cost for 3 streams in the interval [10^{-4}, 10^{-3}]

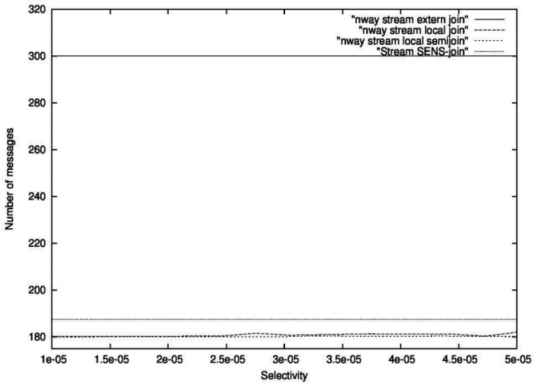

Figure 8. Communication cost for 5 streams in the interval [10^{-5}, 10^{-4}]

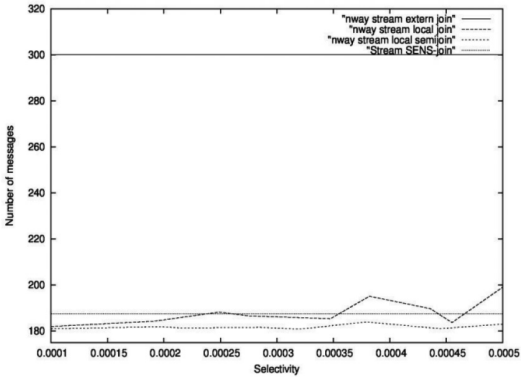

Figure 9. Communication cost for 5 streams in the interval [10^{-4}, 10^{-3}]

7. Impact of Number of Streams

NSLSJ is the technique that permits high performances but up to five streams. Beyond this limit, NSLJ is the best (Figures 10-11). This shows that NSLJ, sends fewer messages for a high number of streams.

However, NSLJ provides worse results than NSLSJ and Sens-join for a lower number of streams, specifically, for less than five streams. Sens-join always offers performances close to those of NSLSJ, but it is less efficient. With Extern-join, the number of transmitted messages is very high and grows considerably.

8. Impact of Sizes of Tables

NSLSJ continuously presents the best results, with minor values for the number of the transmitted messages.

SENS-join is less efficient than NSLSJ, but more efficient than NSLJ for all values of the tested intervals (Figures 12-13). Extern-join has poor results; its values are very high and very far from the results of the other tested techniques.

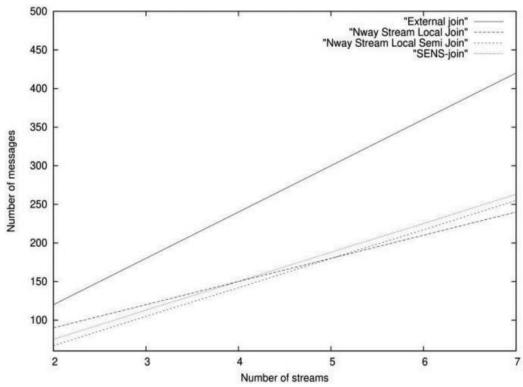

Figure 10. Communication cost depending on streams numbers with selectivity factor equal to 0.000025

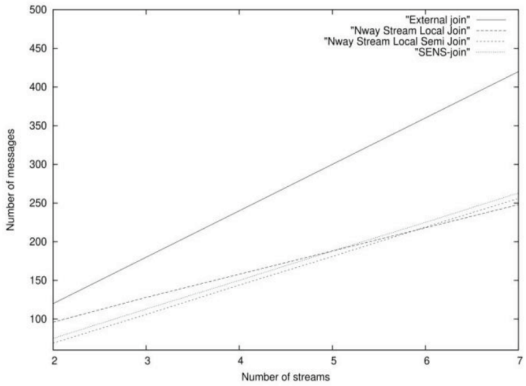

Figure 11. Communication cost depending on streams numbers with selectivity factor equal to 0.00025

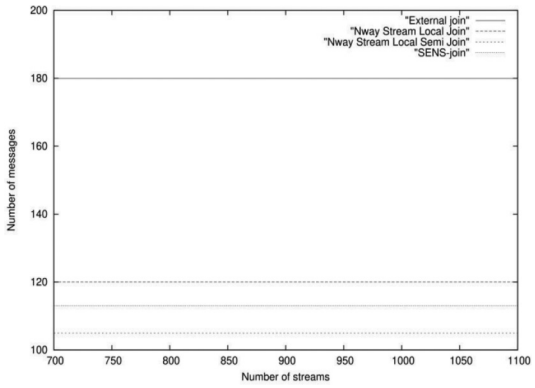

Figure 12. Communication cost depending on windows sizes with selectivity factor equal to 0.000025

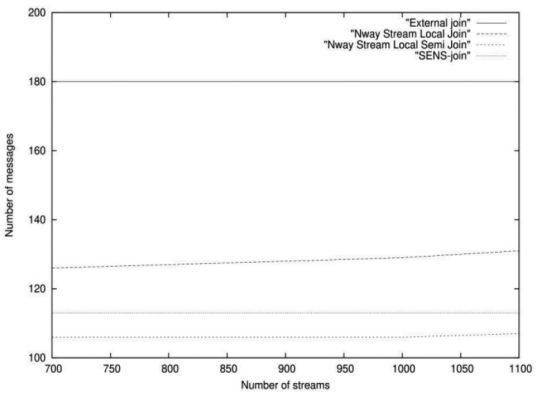

Figure 13. Communication cost depending on windows sizes with selectivity factor equal to 0.00025

8.1. Discussion

All the conducted tests confirm that NSLSJ is the technique that offers the best performance, whether with selectivity factor, number of streams or windows sizes. The efficiency of the NSLSJ lies in its use of the semi-join principle in addition to in-network execution.

Sens-join performances are closer than those of NSLSJ. Sens-join is not as efficient because it uses insufficient filtering to eliminate non-joinable tuples and executes the query totally at the sink.

NSLJ is less than NSLSJ and Sens-join, but for a high number of streams, it offers interesting results. This can be explained by the fact of the reduction of the number of transmitted messages after each execution of an intermediate join.

Extern-join continuously indicates high values of transmitted messages for the selected values of selectivity factor because all concerned tuples must be communicated to the sink where the join query must be performed, which leads to bad performance. Note that Extern-join offers the best execution choice for high value of selectivity factors.

9. Conclusion

Four techniques of n-way stream join queries in the wireless sensor networks have been compared. The first technique is Extern-join, a reference technique in the join execution in wireless sensor networks. The second is Sens-join, an interesting technique with high performance, proposed by Stern et al. The two other techniques have been proposed by Djail et al.

After performing evaluations, NSLSJ showed the best performance due essentially to its adoption of semi-join and the in-networks execution principles. Sens-join has close performances than NSLSJ. However, NSLJ and Extern-join present weak results. A conclusion has been reached that NSLSJ is the best choice for join queries with low join selectivity factors, and extern-join is more accommodating for very high values of selectivity factor.

Future research will focus on improving filtering in NSLSJ, to have more performances than those achieved. Join queries for specific join queries in wireless sensor networks will also be studied. Specific join queries in wireless sensor networks will also be addressed. Recent studies in this field are those of Mo et al. (Mo et al., 2014) for spatial queries and Kang (Kang, 2015) for iceberg joins.

References

Bonfils, B. J., and Bonnet, P. (2004). Adaptive and decentralized operator placement for in-network query processing. Telecommunication Systems, 26(2-4), 389-409.

Boubekeur, D., Khaled, H. W., and Malik, L. (2018). An energy-efficiency technique for n-way stream joins in wireless sensor networks. Nature & Technology Journal, http://www.univ-chlef.dz/revuenatec Vol. A: Fundamental et Engineering Sciences, 18 (2018), 09-15.

Coman, A., Nascimento, M. A., and Sander, J. (2007). On join location in sensor networks. Paper presented at the 2007 International Conference on Mobile Data Management.

Djail, B., Hidouci, K. W., and Loudini, M. (2015). N-way Local Join: a technique for n-way join in wireless sensors networks Paper presented at the International conference on advanced communication systems and signal processing 8-9 November 2015 Tlemcen Algeria.

Djail, B., Hidouci, K. W., and Loudini, M. (2016). N-way Local SemiJoin : A Filtering Technique for N-Way Joins in Wireless Sensors Networks. Journal of Electronic Systems, 6(1), 7-16.

Djail, B., Hidouci, W.-K., and Loudini, M. (2018). NMSJ: A Filtering Technique for N-Way Joins in Wireless Sensors Networks. University Politehnica of Bucharest Scientific Bulletin Series c-Electrical Engineering and Computer Science, 80(4), 23-34.

Djail, B., Hidouci, W. K., and Loudini, M. (2019). A filtering technique for n-way stream joins in wireless sensors networks. Revista de Direito, Estado e Telecomunicações, 11(1).

Djail, B., Hidouci, W. K., and Loudini, M. (2020). A comparative evaluation of techniques for N-way joins in wireless sensors networks. Pollack Periodica, 15(2), 13-24.

Kang, H. (2013). In-network processing of joins in wireless sensor networks. Sensors, 13(3), 3358-3393.

Kang, H. (2015). In-Network Processing of an Iceberg Join Query in Wireless Sensor Networks Based on 2-Way Fragment Semijoins. Sensors, 15(3), 6105-6132.

Karp, B., and Kung, H.-T. (2000). GPSR: Greedy perimeter stateless routing for wireless networks. Paper presented at the Proceedings of the 6th annual international conference on Mobile computing and networking.

Min, J.-K., Kim, J., and Shim, K. (2014). TWINS: Efficient time-windowed in-network joins for sensor networks. Information Sciences, 263, 87-109.

Min, J.-K., Yang, H., and Chung, C.-W. (2011). Cost based in-network join strategy in tree routing sensor networks. Information Sciences, 181(16), 3443-3458.

Mo, S., Fan, Y., Li, Y., and Wang, X. (2014). Multi-attribute join query processing in sensor networks. Journal of Networks, 9(10), 2702-2712.

Savvides, A., Srivastava, M., Girod, L., and Estrin, D. (2004). Localization in sensor networks Wireless sensor networks (pp. 327-349): Springer.

Steinbrunn, M., Moerkotte, G., and Kemper, A. (1993). Optimizing join orders: Citeseer.

Stern, M., Buchmann, E., and Böhm, K. (2009). Towards efficient processing of general-purpose joins in sensor networks. Paper presented at the 2009 IEEE 25th International Conference on Data Engineering.

Tran, T. M., and Lee, B. S. (2010). Distributed stream join query processing with semijoins. Distributed and Parallel Databases, 27(3), 211-254.

Yao, Y., and Gehrke, J. (2003). Query Processing in Sensor Networks. Paper presented at the CIDR.

Yu, H., Lim, E.-P., and Zhang, J. (2006). On in-network synopsis join processing for sensor networks. Paper presented at the 7th International Conference on Mobile Data Management (MDM'06).

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal
eISSN: 2255-2863 - DOI: https://doi.org/10.14201/ADCAIJ2022112 - CDU: 004 -
IBIC: Computación e informática (U) - BIC: Computing & Information Technology (U) -
BISAC: Computers / General (COM000000)

Regular Issue, Vol. 11, N. 2 (2022)

INDEX

Ediciones Universidad Salamanca