A Proposed Technique for Cheating Detection in MCQ Test based on the K-means Method in an Adaptive E-learning System

Noor Al-Deen Alaa Mohammed Tahaa[1], Assist. Prof. Dr. Israa Tahseen Ali[1]

[1]Department of Computer Science, University of Technology, Baghdad, Iraq.

***Abstract*— The most important problem facing adaptive e-learning platforms is cheating in exams and the difficulty of detecting cheating cases and making recommendations for these cheating cases. In this research paper, a technique to detect cheating in MCQ tests is proposed in a proposed adaptive e-learning system, where web usage mining techniques and the k-means algorithm with Levenshtein distance are used to detect cheating by dividing the learners into clusters according to the similarity in the numbers of their choices in the MCQ test. Also using the Levenshtein distance to make a comparison within each cluster between the number series chosen by the learners to show the corresponding learners. The IQ ratio among the apparent learners from the matching process is used to make recommendations for cheating cases. The data set used to test the proposed technique is two data sets. The first data set is a proposed data set and the second data set is a real data set for learners taking MCQ exams in the Department of Computer Science at the University of Technology. When testing the efficiency of this proposed technique, the measures of performance as it comes: Accuracy is 98.182 %, Precision is 100 %, Recall is 98.182 % and F1-measure is 99.1 %.**

***Keywords* — Cheating detection, Web mining, Web usage mining, E-learning, Adaptive E-Learning, K-Means, Levenshtein distance.**

I. INTRODUCTION

Web usage mining is an approach to extract knowledge from the analysis of network usage data about a specific website [1]. This usage data is obtained from server logs and can analyze the behavioral patterns and profiles that interact with the websites [2][3]. This analyzed data is useful and might be used for various needs like web personalization, recommender systems, presentation of promotional contents [4].

An adaptive environment as a platform that includes soft and hard technologies with an aim of raising the learning experience of users through adaptation [5]. Such environments include adaptive hypermedia environments, collaborative learning environments, and simulative/ immersive environments [6]. Adaptive hypermedia environments look to present learning content in such a way that suits a user's academic background, interests, and proclivities [7]. Collaborative learning environments give a method of group learning where learners improve on knowledge by sharing complementary ideas [8], while simulative/immersive environments (through a collection of predefined rules) change according to user actions [9].

K-means clustering algorithm is an unsupervised algorithm that is used to cluster a group of data objects into a number of clusters [10], each cluster includes a set of objects with similar properties among themselves, and these objects are different from the objects in other clusters [11]. The Levenshtein distance is a measure of the difference between two sequences [12][13].

This research paper includes a proposed technique to detect cheating in MCQ tests in a proposed adaptive e-learning system, where usage-based web mining techniques are used with its steps data collections, data preprocessing, pattern discovery and pattern analyzes. And the K-means algorithm with the Levenshtein distance criterion is used in pattern discovery step. also created the IQ feature for each learner to help confirm the case of cheating.

DOI:http://doi.org/10.24086/cocos2024/paper.1472

II. Related works

1) In 2016, Harish kumar B T et al. used hierarchical clustering technique with modified Levenshtein distance, and page rank using access time length, frequency, and higher order Markov model for web page access prediction. They have used NASA web log files as an input to their proposed work. The advantage of the proposed work is improving web prediction accuracy. It can be used to prefetch the web pages before they are being requested by the user, this reduces the access latency. Their results showed that the prediction accuracy for the similarity level of 65% to 75% gives better accuracy [14].

2) In 2017, Mason Chen et al. used Data Mining Algorithms such as Multivariate Correlation, Hierarchical Dendrogram Clustering, Heat Map, and Principal Component Analysis to detect patterns in responses to multiple choice exams that indicate cheating took place among students. The advantage of this work is that the prediction accuracy is very reliable since the answer choice correspondence patterns were identified using various data mining tools and achieved statistical significance. Their results showed that the predictive model approach using Data Mining tools was very powerful for the analysis of complex exam cheating patterns [15].

3) In 2019, Zhizhuang Li et al. proposed a method for detecting cheating in multi-index examinations based on a feed-forward neural network. This method makes comprehensive use of information such as students' cognitive level, seat distribution in the examination room, students' habit of guessing answers at ordinary times, and similarity of examination paper to detect cheating. The advantage of this work is solving disadvantages of existing cheating detection methods such as insufficient modeling accuracy for students, lag in cognitive diagnosis, difficulty in detecting multi-source plagiarism, and low accuracy. Their results showed that the average accuracy rate of this method is 79.4% and the average recall rate of this method is 81.0% [16].

4) In 2020, Vicenzo Abichequer Sangalli et al. proposed several metrics to identify two types of cheating based on co-occurring events and measures of interaction with the course. Depending on the number of accounts in the course, the pairs that solve exercises very near to each other are considered to be collaborating accounts. The proposed metrics are computed for these pairs of accounts and They employ k-means clustering to differentiate between pairs of real students working together in relation to students who are using fake accounts to get the right answers to exercises. The advantage of the proposed work is that the user receives instant feedback when finishing a task. The disadvantage of the proposed work is that this strategy would be hindered due to the time windows used to identify cheating behaviors. Their results show that the accuracy is over 95% in classifying these types of cheating [17].

5) In 2021, Haiyang Hu et al. proposed a cheating detection method based on random forest. For a specific exam question, they find out which exercises the student has done in the usual practice with the same knowledge point as the

DOI:http://doi.org/10.24086/cocos2024/paper.1472

exam question. they use the student's right or wrong of these exercises as the eigenvalues, establish a random forest model, and predict whether the students can correctly answer the questions in the exam. After the establishment of the random forest model, they used the random forest model to predict the student's scores for each question and compare them to the student's actual scores on the exam. Finally, they judge whether the students cheat or not according to the gap between the predicted score and the real score and the similarity between the students and the test papers of surrounding students. The advantage of this work is solving disadvantages of existing cheating detection methods such as insufficient modeling accuracy for students, lag in cognitive diagnosis, difficulty in detecting multi-source plagiarism, and low accuracy. Their results show that the accuracy rate and recall rate of this method are significantly higher than the commonly used cheating detection methods based on test paper similarity and personal fitting index [18].

6) In 2022, Ali M. Duhaim et al. proposed a recommendation system to detect cheating during the online exam using statistical methods, similarity measures, and clustering algorithms (k-means method) by presenting a set of features extracted from the online exam based on the Moodle platform. The results demonstrate that the proposed online exam system is effective in reducing cases of cheating and providing a dependable and reliable online test and the students are cheating at all stages of education. The fourth stage of cheating is estimated at 32%, the 3rd stage at 30%, the 2nd stage at 24%, and the 1st stage at 14% [19].

7) In 2022, Tathagata Sadhukhan et al. proposed a method that provides an advancement to the already existing means to identify the groups of potential cheaters at any fixed level of confidence without the necessity of seating arrangement-based information. They also give a novel and robust quantitative measure to determine the extent of interaction within the suspected groups within which cheating has occurred. The procedure relies on a newly defined Gamma Index which provides a new framework for quantification of the similarity between answers of a pair of candidates. The proposed methodology has three components, A Similarity score (Gamma Index), A clustering technique (Agglomerative hierarchical clustering), and Group Similarity scores. The advantages of this work is that the methodology performed well even in the presence of contaminated responses in the data set and This methodology does not require any prior information corresponding to seating arrangements, time to answer, or wrong-to-right erasures [20].

8) In 2023, Manika Garg et al. used an unsupervised machine learning method to detect students involved in Internet cheating. They used the K-means clustering algorithm to identify two distinct student clusters based on three kinds of attributes, namely assessment data (Score and Time), process data (Revisions, Visits, and Tab), and personality data (Agreeableness and Conscientiousness).

DOI:http://doi.org/10.24086/cocos2024/paper.1472

Analyzing the characteristics of the clusters shows that there are cheaters and honest students in each cluster. The advantages of this work is helping teachers to automatically detect potential cheaters and increase their supervision. This result suggests the importance of analyzing tab-switching behavior along with students' scores, exam times, and personalities to detect cheating [21].

9) In 2024, Mariana Carrasco et al. proposed a novel and robust method to detect potential cheating in online multiple-choice question (MCQ) exams. For the first time, the probability of communication between learners is statistically evaluated based on the answer time and concordance of responses with zero expectations and is used to identify potential cheating. This model is sensitive to the direction of communication actions and distinguishes between content consumption and production and multiwire communication channels. Online remote testing for engineering courses at Técnico Lisboa is used as a case study. They show that repeated occurrences of consistent responses among students contribute cumulatively and provide opportunities to signal misbehavior. Separating content production and consumption highlights the fundamental role of students in potential cheating. Because collusive behavior is assessed using a null model of fraud and compliance, it is statistically assembled and provides a strong baseline to help tutors detect fraud and prevent communication [22].

III. The Proposed Adaptive E-Learning Platform

A learning approach known as "adaptive e-learning" involves teaching or changing content according to the preferences or learning styles of the Learners [23][24].

The proposed system is an adaptive e-learning system contains a proposed technique to detect cheating in MCQ test.

One of the most important tasks carried out by the system, which will be used later in the proposed technique for detecting cheating in the MCQ test, is to calculate the intelligence quotient (IQ) for each Learner based on all the MCQ tests he took in all the subjects belonging to his class and academic branch. This process is done as follows:

Step (1): If the Learner takes only one test, the IQ score will take this test score only as shown in equation (1):

$$\text{IQ} = \text{Mark} \ldots\ldots. (1)$$

Step (2): When a Learner takes a new MCQ test, the process of calculating the intelligence quotient (IQ) is done by dividing the sum of the old IQ multiplied by the number of old MCQ tests completed by the Learner with the score of the new MCQ test by the new number of MCQ tests completed by the Learner as shown in equation (2):

$$(\text{IQ})\ \text{New} = \frac{((\text{IQ})\ \text{Old})\ ((\text{SN})\ \text{Old})) + (\text{Mark})\ \text{New}}{(\text{SN})\ \text{New}} \ldots\ldots. (\mathbf{2})$$

*(SN) Old= The number of MCQ tests completed by the Learner.

*(SN) New= The number of MCQ tests completed by the Learner after doing a new MCQ test.

*(Mark) New= The Learner's score for the new completed MCQ test.

*(IQ) Old= The value of (IQ) for the learner.

DOI:http://doi.org/10.24086/cocos2024/paper.1472

*(IQ) New= The value of (IQ) for the learner after completing a new MCQ test.

IV. The Proposed Technique for the Report Cheating Cases in the MCQ Test

This proposed technique is to make a recommendation report on cheating in the MCQ test by recommending Learners among whom there is a possibility of cheating. In this task, a technique is proposed using web usage mining techniques, where the K-means clustering algorithm is used to cluster the data, represented by the numbers of Learners' choices in the MCQ test, to cluster these data into clusters according to the amount of similarity between the numbers of Learners' choices on the questions. The output of this clustering process is clusters, each cluster contains a set of choice number series for the tested questions, and each series belongs to each Learner. As each set of chains in a particular cluster is close to each other, which differs from other clusters. Since the entered data into the algorithm is the Learner's choice chains in the test, the Levenshtein distance is used to measure the distance between the center and the Learners' choice chains in the test based on the minimum distance. The next step in the proposed technique is to compare the strings of Learners' choices in each product cluster where the Levenshtein distance is used to compare each of the two series and measure their similarity. The similarity ratio to be reached is also determined in order to compare it with the similarity ratio between two series resulting from the comparison process resulting from the Levenshtein distance, where if the ratio is greater and equal to the specified similarity ratio, the two series are entered to the next step. The last step of the proposed technique is to take the absolute difference between the IQ ratio of the two Learners to whom the two resulting series belong and compare it with the IQ difference percentage determined in advance. Where if the difference in IQ resulting from the last step of the proposed technique is greater and equal to the IQ difference percentage determined in advance, the Learners and their proposal to the lecturer are shown by the system as a case of cheating in the test with the percentage of cheating shown. Figure (1) shows the block diagram of report cheating cases in the MCQ test technique task. This proposed technique contains the following steps, as follows:

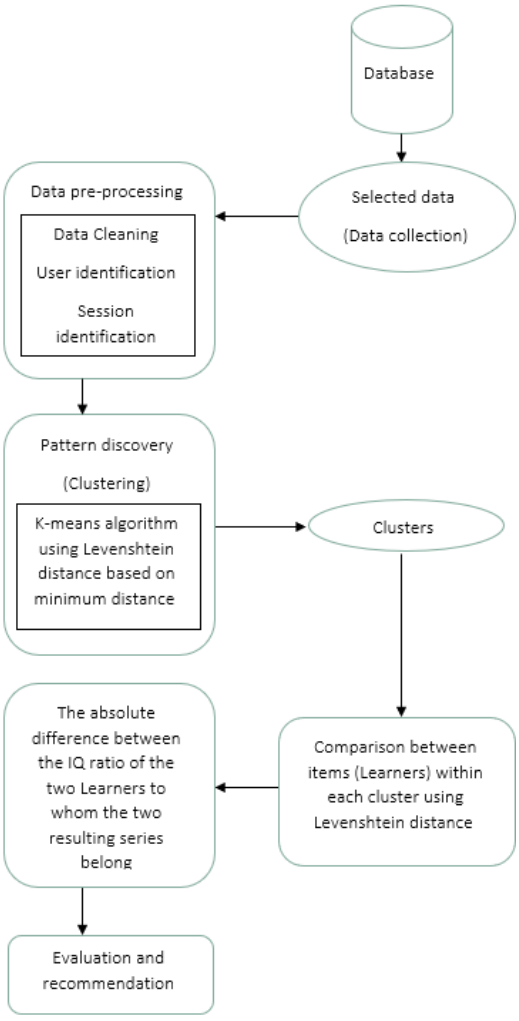

Figure (1): - Block diagram of report cheating cases in the MCQ test technique task.

DOI:http://doi.org/10.24086/cocos2024/paper.1472

Step (1): The collecting of data

The first step is to collect the relevant data from the data log file. This data is represented in the proposed system by the Learner's choices numbers on the questions for each MCQ test, the IQ of each Learner and the number of questions for each test.

Step (2): Data preprocessing

In the data preparation step, the data is first cleaned of noise and the empty data records are deleted. Secondly, in the process of preparing the data, users or Learners are identified through the ID of each user or Learner, the IQ for each Learner, the Learner's class, the scientific subject, and the scientific topic tested in which the data are related to this scientific topic.

The process of data preparing is the definition of the user session. It is represented in the proposed system of testing the Learner for each scientific topic. Where this process is done by first defining the Learner's choice number for each question. The next step of the session identification process is to arrange the choices numbers for the completed test questions into a series of choices numbers.

Finally, the resulting data is a set of series of numbers that the Learner has chosen in the test questions, where the number of choice numbers for each series depends of course on the number of test questions. Where each Learner has a series of numbers of his own choices.

Step (3): Pattern discovery

In the process of pattern detection, the k-means clustering algorithm is used, and the Levenshtein distance (LD) is used to measure the distance between the center and the elements. The reason for using the Levenshtein Distance method is that the function of the (LD) method is to calculate the distance and the closeness of two strings [25]. In this proposed system technique, the minimum distance is adopted to generate groups or clusters.

This clustering process is done by entering the prepared data represented by the set of choices numbers strings (E), choosing the number of clusters (K), and determining the maximum number of cycles (MaxIter). Then the number of centers (C) is chosen randomly and the distance between the centers and elements is measured to complete the clustering process. Since the input data is a set of choices numbers strings (E), the Levenshtein distance is used to measure the minimum distance (LDM) between the center's choice number series and the group's choice number series.

Upon completion of the first cycle of the clustering process, the centers are renewed by comparing each choice number with the corresponding choice number from the other series from all the choice number series in each resulting cluster, where if the appearance of the number one of the first choice is more than the number two, three and four, the number one is taken to form the new center, and if the appearance of the number two occurs, the number two is taken to form the new center, and the same is true for the number three and four, and this comparison process continues for all the choices numbers of the choice number series for a particular cluster until the new center is created. This clustering process continues until the specified number of clustering cycles is reached.

The output of this clustering process are clusters of the number series of Learners' choices (L) on the test questions that are close to each other according to the amount of similarity.

DOI:http://doi.org/10.24086/cocos2024/paper.1472

Step (4): Comparison between items (Learners) within each cluster using Levenshtein distance

After the clustering process is completed, the output is a set of clusters. Each cluster contains a group of convergent Learner choices number series, with each Learner having a series of choices number on the completed test questions, which depend on the number of questions.

In this step of the proposed technique, the comparison between the number series of choices within each cluster is done. Where the comparison is between every two strings of choice numbers for each string of numbers present in the same cluster. This comparison process is done using the Levenshtein distance.

Then the percentage of congruence between each two series is measured by dividing the product of subtracting the total number of questions for the test (N) from the distance resulting from the comparison process by the total number of questions (N) for the test multiplied by one hundred as in the equation (3):

$$\text{Percentage of congruence} = \frac{\text{N} - \text{distance}}{\text{N}} * 100 \dots\dots (\mathbf{3})$$

And by specifying a percentage of congruence (Percentage goal) by the lecturer, which is intended to be reached as a case of cheating, if the resulting match percentage is greater and equal to the specified match percentage, then these two series will be included in the last step of the proposed technique.

Step (5): The absolute difference between the IQ ratio of the two Learners to whom the two resulting series belong

After the comparison step is completed and the two sequences belonging to two Learners appear, it is necessary to strengthen the recommendation of this technique to these two Learners as a case of cheating. In this step, the IQ of the two Learners is taken as two factors to measure the extent of the difference between their IQ. If the difference is large, these two Learners are recommended by the system as a case of cheating, and the system also recommends who the cheating Learner is, based on the Learner with the lowest IQ.

This process is done by first determining the percentage of the difference between the IQ to be reached (IQ goal) and secondly by taking an absolute subtraction of the IQ of the two Learners to whom the two series of choices numbers resulting from the previous step belong. Where if the difference resulting from the subtraction process is equal to or greater than the specified difference, the system recommends that there is a case of cheating between these two Learners, with the Learner recommending cheating based on the lower IQ (R[i,j]). Algorithm (1) show the steps of this technique.

Algorithm (1): Cheating detection.

Input: E= {e1, e2, …, e(n)}
K
MaxIter
Percentage goal
IQ goal

Output: R[i,j]

Step (1): For each c(i) in C do (Choosing random centroids from (E))
C ← centroid(E)
End For

DOI:http://doi.org/10.24086/cocos2024/paper.1472

Step (2): For each e(i) in E do (Distance between the centroids and elements)
L(e(i)) ← LDM (e(i), c(j))
End For

Step (3): Changed ← false (No change)

Step (4): Iter ← 0 (Zero iterations)

Step (5): For each c(i) in C do (Renewal of centroids)
c(i) ← centroid (L(i))
End For

Step (6): For each e(i) in E do (Distance between the new centroids and elements)
Mindist ← LDM (e(i), c(j))
If Mindist ≠ L(e(i)) then
L(e(i)) ← Mindist
Changed ← true
End If
End For

Step (7): Iter+1 (Increase the number of iterations by 1)

Step (8): Repeat Step (5), Step (6) and Step (7)
Until (Changed = true) and (Iter ≤ MaxIter)

Step (9): For each L(e(i)) do (Comparison between items within each cluster)
For each e(i) Є E do
Distance = LD (e(i), e(j)) in L(e(i));
Percentage of congruence = (N – Distance / N) * 100;
If Percentage of congruence >= Percentage goal then
G [i, j] ← e(i), e(j)
End If
End For
End For

Step (10): For each G [i, j] do (The difference between the IQ of the two Learners)
IQ = | IQ(e(i)) – IQ(e(j)) |
If IQ >= IQ goal then
R [i, j] ← e(i), e(j)
End If
End For

V. EVALUATION OF PERFORMANCE

In order to measure the quality of the used method in the proposed technique, various measures are used, namely: Accuracy (A), Recall (R), Precision (P), and F-Measure (F1) [26].

Accuracy is that the parameter used to assess the effectiveness of the projected system with relevance to all three techniques. A matrix is built to measure accuracy as shown below in table (1) [27].

Table (1): Recommendation matrix [27].

	Recommended items by the system	Items not recommended by the system
Expected item	True ─ Positive (TP)	False ─ Negative (FN)
Not an expected item	False ─ Positive (FP)	True ─ Negative (TN)

Based on the recommendation matrix we calculate accuracy (A), recall (R) and precision (P) as shown in the equations below [27].

1) Accuracy: The accuracy is the ratio of true positive to the sum of all the items recommended [27].

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \ldots\ldots (\mathbf{4}).$$

2) Recall: The recall is outlined as a fraction of all relevant items that are recommended by the system [27].

$$\text{Recall} = \frac{\text{True\_Positive (TP)}}{\text{True\_Positive (TP)} + \text{False\_Negative (FN)}} \ldots\ldots (\mathbf{5}).$$

3) Precision: Precision can be a fraction of all the recommended products that are relevant [27].

DOI:http://doi.org/10.24086/cocos2024/paper.1472

$$\text{Precision} = \frac{\text{True\_Positive (TP)}}{\text{True\_Positive (TP)} + \text{False\_Positive (FP)}} \ldots\ldots (\mathbf{6}).$$

4) F – Measure or F1 score: F-Measure conjointly referred to as the F1 Score provides the weighted average of precision and recall. F1 is calculated as in equation [26].

$$F1 = \frac{2\,(R * P)}{(R + P)} \ldots\ldots (\mathbf{7}).$$

TP → True – Positive → Sequences correctly identified [26].

TN → True – Negative → Sequences correctly rejected [26].

FP → False – Positive → Sequences incorrectly identified [26].

FN → False – Negative → Sequences incorrectly rejected [26].

VI. EXPERIMENTAL RESULTS

In the proposed system, two types of data have been used or entered, the first being suggested or assumed data, and the second type being real data.

The proposed data is the initial proposed Learner data to test the proposed technique, which has been proposed in a variety of ways to test the validity of the technique's work. Fifty-four Learner accounts are created and these accounts are tested in more than one suggested MCQ test for more than one proposed topic and related to a proposed scientific subject.

The real data are MCQ tests for Learners of the University of Technology - Department of Computer Science, which belong to real topics, and these topics belong to a real scientific subject. This data consists of MCQ questions, the choices of these questions, the correct answer to these questions, the Learners' choices or their answers to each question in each test, and the Learners' scores in each test. It is worth noting that the number of Learners is seventy-four.

The results extracted from the proposed cheating detection technique are explained, as follows:

Phase (1): Data preprocessing results

After collecting the represented data by the Learners' choices in the exam and the IQ of each Learner, this data enters the stage of preparing the data, which is the identification of each Learner and a series of numbers chosen by each Learner in the test. Step (2) is related to data processing as in paper.

Phase (2): The discovery of pattern results

After the data preparation process, the data is entered into the cheating detection technique. The output of this technique for the proposed data is shown in figure (2) and figure (3).

DOI:http://doi.org/10.24086/cocos2024/paper.1472

Figure (2): Report matching the choices of Learners obtained from the cheat detection technique for the first test of the proposed data.

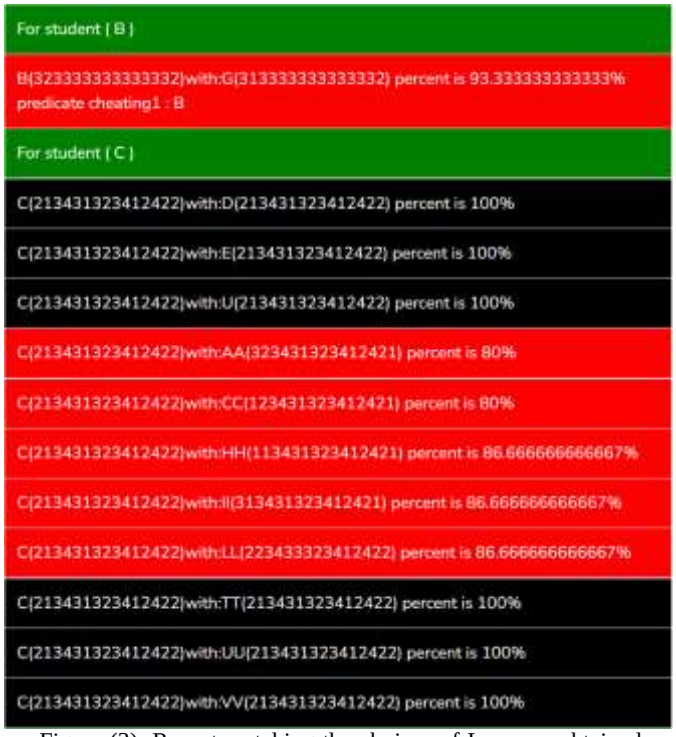

Figure (3): Report matching the choices of Learners obtained from the cheat detection technique for the second test of the proposed data.

As for the real data the output of this technique is shown in figure (4):

Figure (4): Report matching the choices of Learners obtained from the cheat detection technique for the real data.

The aforementioned reports set the percentage of congruence at 80 percent. To calculate the accuracy measures of the proposed cheat detection technique, a normal comparison is made between the number series of students' choices, and the percentage of congruence is determined 80 percent, as the outputs of this comparison certainly do not accept error. Then, the comparison with the results of the congruence report from the proposed technique, in which the K-means clustering algorithm is used, where satisfactory results are obtained, and these results are illustrated in the following table (2):

The proposed technique	Table (2): Performance measures for the proposed cheating detection technique.			
	Accuracy	**Precision**	**Recall**	**F1-Measure**
	98.182 %	100 %	98.182 %	99.1 %

After the matching process, the system recommend which Learners may be cheating by calculating the

DOI:http://doi.org/10.24086/cocos2024/paper.1472

difference between the IQs of the two Learners. Where the difference between the IQ score of five and above has been determined, where if this condition is fulfilled among the identical Learners, the system will recommend that they are cheating, as shown in the following figures:

Figure (5): Recommendation of cheating by the system based on the difference between the IQs of the two Learners for the proposed data.

Figure (6): Recommendation of cheating by the system based on the difference between the IQs of the two Learners for the real data.

VII. CONCLUSION

Using web usage mining techniques and the k-means algorithm with Levenshtein distance to divide learners. Then using the Levenshtein distance within each resulting cluster to make a comparison and show the corresponding learners. The IQ of the matching Learners is taken in the number sequences of the Learners' choices and the outcomes of the algorithm, then the difference between the IQ ratios is calculated to make the cheating recommendation. It is worth noting that in the clustering process using the k-means algorithm for the cheating detection technique, the problem of the algorithm not stopping appeared, as this problem is solved by determining the number of iterations in the algorithm. As a conclusion, the best results for the cheating detection technique are obtained when making the number of iterations of the k-means algorithm 50 iterations and the results are: Accuracy is 98.182 %, Precision is 100 %, Recall is 98.182 % and F1-measure is 99.1 %. And it is worth noting that the number of iterations specified more than once. For example, when setting the number of iterations to 31 iterations, the results are: Accuracy is 97.436 %, Precision is 100 %, Recall is 97.436 % and F1-measure is 98.7 %, so the best results are obtained when the iterations set to 50 iterations.

REFERENCES

[1] M. Dhandi and R. K. Chakrawarti, "A comprehensive study of web usage mining," 2016 Symp. Colossal Data Anal. Networking, CDAN 2016, 2016, doi: 10.1109/CDAN.2016.7570889.

[2] R. K. Shukla, P. Sharma, N. Samaiya, and M. Kherajani, "WEB USAGE MINING-A study of web data pattern detecting methodologies and its applications in data mining," 2nd International Conference on Data, Engineering and Applications, IDEA 2020. 2020. doi: 10.1109/IDEA49133.2020.9170690.

[3] S. Yadao, A. V. Babu, M. Janarthanan, and A. Bhaumik, "Web usage mining: A comparison of WUM category web mining algorithms," Proceedings of the 3rd International Conference on Intelligent Communication Technologies and Virtual Mobile Networks, ICICV 2021. pp. 1020–1024, 2021. doi: 10.1109/ICICV50876.2021.9388539.

[4] N. P. Jilhedar and S. K. Shirgave, "User Web Usage Mining for navigation improvisation using semantic related frequent patterns," International Conference on Computing and Communication Technologies, ICCCT 2014. 2014. doi: 10.1109/ICCCT2.2014.7066697.

[5] P. C. Hu and P. C. Kuo, "Adaptive learning system for E-learning based on EEG brain signals," 2017 IEEE

DOI:http://doi.org/10.24086/cocos2024/paper.1472

6th Global Conference on Consumer Electronics, GCCE 2017, vol. 2017-Janua. pp. 1–2, 2017. doi: 10.1109/GCCE.2017.8229382.

[6] U. C. Apoki, H. K. M. Al-Chalabi, and G. C. Crisan, "From Digital Learning Resources to Adaptive Learning Objects: An Overview," Communications in Computer and Information Science, vol. 1126 CCIS. pp. 18–32, 2020. doi: 10.1007/978-3-030-39237-6_2.

[7] R. D. Ara´ujo, R. G. Cattelan, and F. A. Dorc¸a, "Towards an Adaptive and Ubiquitous Learning Architecture," 2017 IEEE 17th International Conference on Advanced Learning Technologies Towards. pp. 539–541, 2017. doi: 10.1109/ICALT.2017.63.

[8] P. Chopade, S. M. Khan, D. Edwards, and A. Von Davier, "Machine Learning for Efficient Assessment and Prediction of Human Performance in Collaborative Learning Environments," 2018 IEEE International Symposium on Technologies for Homeland Security, HST 2018. 2018. doi: 10.1109/THS.2018.8574203.

[9] J. Elen and M. J. Bishop, "Situated learning in virtual worlds and immersive simulations," Handbook of Research on Educational Communications and Technology: Fourth Edition. pp. 347–248, 2014.

[10] M. Mardi and M. R. Keyvanpour, "GBKM: A New Genetic Based K-Means Clustering Algorithm," 2021 7th International Conference on Web Research, ICWR 2021. pp. 222–226, 2021. doi: 10.1109/ICWR51868.2021.9443113.

[11] Y. S.Thakare and S. B. Bagal, "Performance Evaluation of K-means Clustering Algorithm with Various Distance Metrics," International Journal of Computer Applications, vol. 110, no. 11. pp. 12–16, 2015. doi: 10.5120/19360-0929.

[12] D. Soyusiawaty and F. Rahmawanto, "Similarity detector on the student assignment document using levenshtein distance method," in 2018 International Seminar on Research of Information Technology and Intelligent Systems, ISRITI 2018, 2018, pp. 656–661. doi: 10.1109/ISRITI.2018.8864339.

[13] Sugiarto, I. G. S. M. Diyasa, and I. N. Diana, "Levenshtein distance algorithm analysis on enrollment and disposition of letters application," Proceeding - 6th Information Technology International Seminar, ITIS 2020. pp. 198–202, 2020. doi: 10.1109/ITIS50118.2020.9321030.

[14] B. T. Harish Kumar, L. Vibha, and K. R. Venugopal, "Web page access prediction using hierarchical clustering based on modified levenshtein distance and higher order Markov model," Proceedings - 2016 IEEE Region 10 Symposium, TENSYMP 2016. pp. 1–6, 2016. doi: 10.1109/TENCONSpring.2016.7519368.

[15] M. Chen, "Detect multiple choice exam cheating pattern by applying multivariate statistics," Proceedings of the International Conference on Industrial Engineering and Operations Management, vol. 2017, no. OCT. pp. 173–181, 2017.

[16] Z. Li, Z. Zhu, and T. Yang, "A Multi-Index Examination Cheating Detection Method Based on Neural Network," 2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI). pp. 575–581, 2019. doi: 10.1109/ICTAI.2019.00086.

[17] V. A. Sangalli, G. Martinez-Munoz, and E. P. Canabate, "Identifying cheating users in online courses," IEEE Global Engineering Education Conference, EDUCON, vol. 2020-April. pp. 1168–1175, 2020. doi: 10.1109/EDUCON45650.2020.9125252.

DOI:http://doi.org/10.24086/cocos2024/paper.1472

[18] Hu, Haiyang, Z. Li, and Z. Wang, "Test cheating detection method based on random forest," 2021 3rd International Conference on Computer Science and Technologies in Education (CSTE). IEEE, 2021. doi: 10.1109/CSTE53634.2021.00017.

[19] A. M. Duhaim, S. O. Al-mamory, and M. S. Mahdi, "Cheating Detection in Online Exams during Covid-19 Pandemic Using Data Mining Techniques," Webology, vol. 19, no. 1. pp. 341–366, 2022. doi: 10.14704/web/v19i1/web19026.

[20] T. Sadhukhan, D. Bag, M. Paul, A. Chattopadhyay, and B. K. Roy, "Detection algorithm of large-scale collusion in M.C.Q. examinations," HAL open science. pp. 1–19, 2022. hal-03768538.

[21] M. Garg and A. Goel, "Detection of Internet Cheating in Online Assessments Using Cluster Analysis," Springer Nature Singapore Pte Ltd. 2023 N. Sharma et al. (eds.), Data Management, Analytics and Innovation, Lecture Notes in Networks and Systems 662. pp. 77–90, 2023. doi: 10.1007/978-981-99-1414-2_7.

[22] M. Carrasco, A. R. Silva, and R. Henriques, "Detecting Fraudulent Student Communication in a Multiple Choice Online Test Environment," IEEE TRANSACTIONS ON COMPUTATIONAL SOCIAL SYSTEMS, VOL. 11, NO. 1, FEBRUARY 2024. pp. 1108–1120.

[23] H. A. El-Sabagh, "Adaptive e-learning environment based on learning styles and its impact on development students' engagement," International Journal of Educational Technology in Higher Education. Springer. pp. 1–24, 2021. doi: 10.1186/s41239-021-00289-4.

[24] H. K. Majeed Al-Chalabi and A. M. A. Hussein, "Pedagogical Approaches in Adaptive E-learning Systems," CALIFORNIA INSTITUTE OF TECHNOLOGY. IEEE. 2020.

[25] N. Anggraini and M. J. Tursina, "Sentiment Analysis of School Zoning System on Youtube Social Media Using the K-Nearest Neighbor with Levenshtein Distance Algorithm," 2019 7th International Conference on Cyber and IT Service Management, CITSM 2019. 2019. doi: 10.1109/CITSM47753.2019.8965407.

[26] O. El Aissaoui, Y. El Madani El Alami, L. Oughdir, and Y. El Allioui, "Integrating web usage mining for an automatic learner profile detection: A learning styles-based approach," 2018 Int. Conf. Intell. Syst. Comput. Vision, ISCV 2018, vol. 2018-May, pp. 1–6, 2018, doi: 10.1109/ISACV.2018.8354021.

[27] P. Lopes and B. Roy, "Dynamic recommendation system Using web usage mining for E-commerce users," Procedia Comput. Sci., vol. 45, no. C, pp. 60–69, 2015, doi: 10.1016/j.procs.2015.03.086.

DOI:http://doi.org/10.24086/cocos2024/paper.1472