# Cheating Detection in Online Exams during Covid-19 Pandemic Using Data Mining Techniques

**Ali M. Duhaim**
Informatics Institute for Postgraduate Studies, Baghdad, Iraq.
E-mail: ms20180516@iips.icci.edu.iq

**Safaa O. Al-mamory**
University of Information Technology and Communications, Baghdad, Iraq.
E-mail: salmamory@uoitc.edu.iq

**Mohammed Salih Mahdi**
University of Information Technology and Communications, Baghdad, Iraq.
E-mail: mohammed.salih@uoitc.edu.iq

*Received August 07, 2021; Accepted November 24, 2021*
*ISSN: 1735-188X*
*DOI: 10.14704/WEB/V19I1/WEB19026*

## Abstract

Face-to-face learning has been replaced by E-learning due to the closing of academic institutions in the world during the covid-19 pandemic. Educational institutions faced many challenges in the online platforms and the most important of which was assessing students' performance, which resulted in the general problem of cheating detection in the online exams. E-learning has grown significantly every day over the last decade with the growth of the internet and technology. Therefore, an online examination can be beneficial for people to take the exam, but cheating in tests is a common phenomenon around the world. As a consequence, the prevention of cheating can no longer be completely effective. Many researchers discussed online examination cheating without addressing an important point, which is analyzing students' answers to find similar responses between them.

This paper proposed a recommendation system for evaluating students' answers and detecting cheating during an online exam utilizing statistical methods, similarity measures, and clustering algorithms by presenting a set of features derived from an online exam based on the Moodle platform. The results showed that the suggested online examination system effectively reduces cheating and provides a reliable online exam. In conclusion, presenting an effective and fair system that maintains academic integrity, which is the most important aspect of education.

## Keywords

E-learning, Moodle, Online Exam, Cheating Detection, Similarity Measures, Clustering Algorithms.

## Introduction

In today's world, E-learning has grown in popularity among academic institutions and organizations. The main benefit of E-learning is that it is accessible to all individuals, regardless of age, place, or time available to learn the contents. The Learning Management System (LMS) is an essential tool in an E-learning system. Many educational institutions use the LMS as a platform to access E-learning materials. In an E-learning environment, students will determine the device for content learning, such as laptop/tablet/mobile. Once the students have learned the materials, they must be evaluated by exams. As a result, in an E-learning environment, exams are essential for assessing the learner's performance (Deborah L et al., 2019).

Today's online exam is an essential part of E-learning solutions for efficient and equal evaluation of students' results. The most challenging aspect of E-learning is evaluating the students' performance during online exams. In particular, online examinations are usually performed on E-learning sites without students and teachers being physically present in the same area. This creates some loopholes in online exams in terms of honesty and fairness. In addition, online examination environments are susceptible to cheating. It is possible to access many data resources online without any checks or balances from students (Muzaffar et al., 2021).

To avoid cheating during an online exam, researchers offered a variety of solutions, such as biometric methods and online proctoring to ensure integrity and fairness depending on artificial intelligence techniques.

This research aims to construct a new model for cheating detection in the online exam based on a reliable dataset and affected features. Also, to create the fairest and effective system for assessing students' performance. In particular, the main contribution of this system is divided into three layers:

1. In the first layer, three online exam features were defined statistically: IP address for each student, the time spent in the exam, and the time late for the exam.
2. In the second layer, the similarity between students' answers was calculated using an overlap similarity algorithm. This layer utilizes the essay question type.
3. The students' answers were divided into similar groups in the third layer using the simple k-means algorithm. The question types used in this layer are (multichoice, true & false, calculated, numerical, multi-answer, and drag & drop).

The rest of the paper is structured as follows. In the second section, we explored the literature review. The third section covers research methodology, including the proposed online exam system and research techniques. In the fourth section, we described the results and evaluation of the system. Finally, the fifth section includes a summary of the system.

## Literature Review

Educational institutions use the online exam system to improve the quality of education by assessing students' performance in self-paced learning environments. However, despite the importance of the online exam, students engaging in cheating is a widespread phenomenon worldwide (Ghizlane et al., 2019). Therefore, in the field of online exams, several academic researchers have been conducted, including continuous authentication, biometrics methods, face-tracking techniques, and other approaches described below:

For instance, (Chuang et al., 2017) introduced a method for determining head position and time delay for detecting cheating in the online exam session. They also discussed that a student's head position variation compared to a computer screen has a strong statistical relationship with cheating behavior. Thereby can automatically identify suspicious student activities in the online course. Similarly, (Hu et al., 2018) proposed a new method for monitoring the student's abnormal behavior during an online exam, which determines the relationship between the head and mouth of the examinee through a webcam. Experiments have shown that the proposed method was effective for identifying abnormal behavior in the online course.

Moreover, students' strategies for detecting cheating in online exams were discussed. (Bawarith et al., 2017) suggested an e-exam monitoring system to detect and avoid cheating during the exam. The system used continuous authentication of the fingerprint reader and the eye tribe tracker. As a result, the system classified the examinee's status as cheating or non-cheating based on two parameters: the examinee's total time on screen and the number of times the examinee is off-screen.

(Mungai & Huang, 2017) reviewed the significance of keystroke dynamics in keeping security in online exams. The proposed system used a three-stage authentication method, using statistical verification, machine learning, and logical comparison. When an applicant first logs into the system, his typing style is automatically registered, and a template is generated for him. These templates are used as a guide to ensure that the user is authenticated at all times when taking an online exam, based on several parameters, which are: dwell time (time difference between pressing and releasing keys) and flight time (time difference between key release and the next keypress) and typing speed of user for better precision and responsiveness.

(Prathish et al., 2017) proposed an inference system that would assist the instructor in monitoring students during the online exam. They identified the examinee's face based on differences in yaw direction, audio appearance, and successful window capture. The system was checked in an E-learning environment and effectively achieved in online exam monitoring. In a similar study, (Ketab et al., 2017) presented the development of a more reliable, flexible, and continuous authentication system for online assessments. The system has a continuous user identification using multimodal biometrics to monitor the examiner to ensure that only a valid student takes the exam; a security layer that uses an eye tracker to watch/record student eye movement; and speech recognition to detect unwanted contact.

(Mahadi et al., 2018) discussed several techniques and suggested combining (facial recognition and keystroke dynamics) could be the best classifiers in the online course for behavioral biometric authentication. Similarly, (Ghizlane et al., 2019) also suggested a combination of smart cards (to check student's identity) and face recognition techniques (for continuous monitoring of a student's webcam) to detect any suspicious behavior during the online exam and avoid any kinds of cheating attempts.

(Shdaifat et al., 2020) proposed a model that uses a biometric iris recognition technique in addition to the traditional method of mobile examination login in mobile learning. The suggested model captures iris images randomly, which helps improve the student's authentication during the exam. The study aimed to avoid student impersonation and cheating in mobile exams.

A study suggested by (Golden & Kohlbeck, 2020) paraphrasing questions was used to minimize the benefits of online cheating. They challenged students with a verbatim test bank question and a paraphrased question for each topic chosen. Students recorded higher performance on verbatim questions comparing to paraphrasing (80.4% vs. 69.1%). The study showed that they could not quickly answer a paraphrased test bank question since it does not appear online in its original and verbatim form. Thereby, cheating is minimized, academic integrity is preserved, and useful for professors who wish to eliminate the risks of using test banks.

(A et al., 2020) implemented an intelligent monitoring system to detect suspicious student activity in the examination hall using a high-density camera to record all of the participants in the session. This study helps identify the students' abnormal behavior, avoiding the presence of a supervisor in the hall and providing evidence of cheating.

## Research Methods

This section describes the proposed system of cheating detection in the online exam, problem assumptions, features extraction, and techniques used in this paper. The following sections discuss the results and implementation.

### 1. Proposed System of Cheating Detection in the Online Exam

Previous researches in the area of online exam integrity have several limitations. Some have regularly taken images of each student, while others have employed video cameras to record the students' behavior during exams. However, these systems violate the privacy of students and require fast internet access and powerful software. The primary goal of this research is to use data mining techniques to assess students' answers after the exam. The suggested online examination system is described in Figure 1.

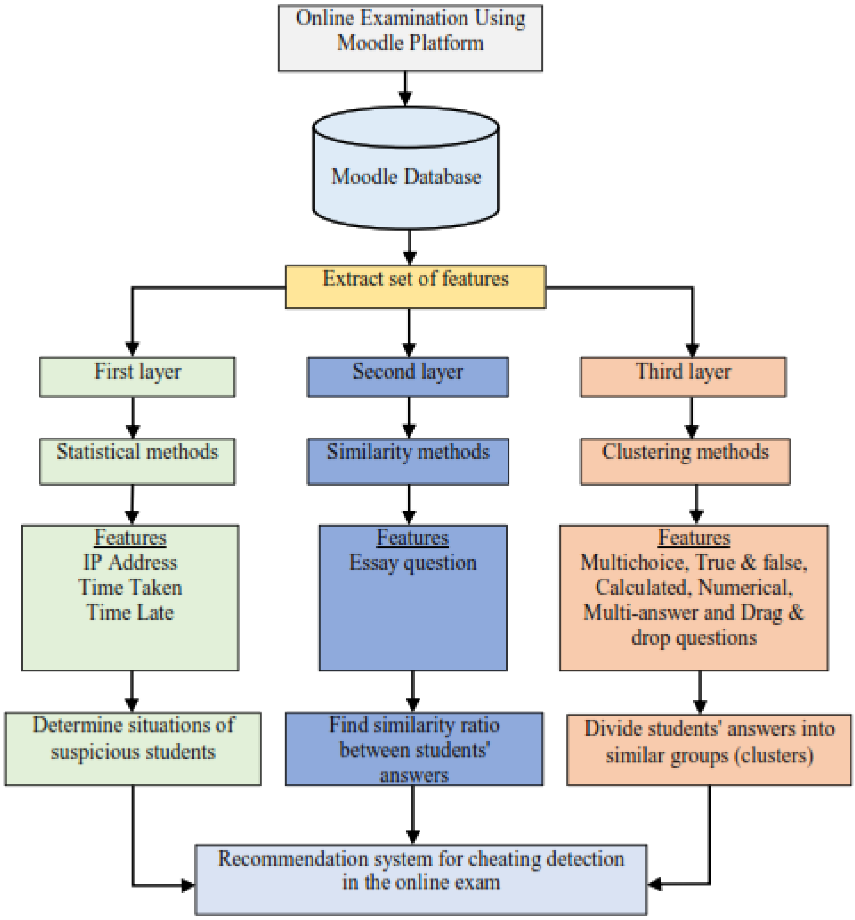

Online Examination Using Moodle Platform
Moodle Database
Extract set of features
First layer
Second layer
Third layer
Statistical methods
Similarity methods
Clustering methods
Features
IP Address
Time Taken
Time Late
Features
Essay question
Features
Multichoice, True & false, Calculated, Numerical, Multi-answer and Drag & drop questions
Determine situations of suspicious students
Find similarity ratio between students' answers
Divide students' answers into similar groups (clusters)
Recommendation system for cheating detection in the online exam

**Figure 1 Reliable Online Examination System**

Initially, the student performs the exam on the Moodle platform, and then exam information is generated and stored in the Moodle database. The entire procedure of our proposed system is organized into three layers. In the first layer, three features derived from the exam (IP address, time taken, and time late) were employed to evaluate the examinee's status using statistical methods.

In the second layer, similarity algorithms were utilized to calculate the similarity between students' answers to the essay questions. Clustering algorithms were used in the third layer to separate students' answers into related groups based on different question types such as multichoice, true & false, calculated, numerical, multi-answer, and drag & drop. Finally, the examiner was provided with a recommendation system for students who cheated in the online exam.

## 2. Problem Assumptions

We considered the following assumptions in this research:

1. The online exam was implemented using the Moodle platform (Https://Moodle.Org/, n.d.).
2. The student must perform his exam alone; otherwise, the proposed system regarded the presence of more than one student in the same location as evidence of cheating.
3. Handwriting questions are not included in our proposed system.
4. When creating an online exam, you can utilize any type of these questions (multichoice, true & false, essay, calculated, numerical, multi-answer, and drag & drop).
5. A recommendation system has been presented to the examiner about students who cheated in the online exam.

## Features Extraction and Techniques

This part contains a detailed description of every technique and feature utilized in this research.

### A) First Layer

To identify cheating situations during an online exam, statistical methods were used based on the following features:

1. **IP Address**: IP address is the student's network address, which must be unique for each student. During the exam, most students congregate in one area to exchange answers and assist each other. Thereby, if the students connect to the same network, the system will detect them by matching IP addresses.

2. **Time Taken**: time taken is the difference between the finish time and the start time for each student. Several students finish the online exam in a quarter-time given by the examiner, which is against the examination rules because the student cannot leave the exam session while taking the face-to-face exam in such a scenario. The students can share solutions with each other using social media platforms, leading to faster answers, and the exam is done in a quarter of the time.
3. **Time Late**: time late is the difference between the start time of the student and the exam's start time. For example, some students are late accessing the online exam at the scheduled time to get the correct answers from others who took the exam.

As a result, our proposed system is considered evidence of cheating when students utilize the same IP address, complete the online exam in a quarter-time, and late for an exam more than ten minutes.

### Pre-processing of the First Layer

1. Convert each time from Unix format to readable date to precisely determine the time and know the hours, minutes, and seconds of each exam, for example (**1595232387** converted to **08:06:27 AM**).
2. Encoding the IP address before searching for similar networks between students; because the network address is divided into four digits, searching for one digit is faster when encoded, for example (**107.10.208.3** encoded to **1).**

### B) Second Layer

To calculate the similarity between the answers, similarity measures were applied to the essay questions (as features) in this layer. An essay question is a test question that requires a written analysis or summary of a specific topic, usually of a defined length. It includes a paragraph, sentence, or short composition.

As a result, if the ratio of matching between responses is greater than 65%, our proposed system considers it evidence of cheating.

### Pre-processing of the Second Layer

Before applying similarity algorithms to essay questions, a set of operations must be processed in order to identify students who have the same answer:

1. All unwanted symbols are converted to space since it's not necessary during the matching process such as "$", "@", "%", etc.

2. Convert all words from upper case to lower case to unify the letters of the word during matching.
3. All punctuation marks and numbers are removed.
4. All white spaces at the beginning, end, and middle of the essay are stripped.
5. The English stop words are removed, which are commonly used terms such as ("the", "an", "a", etc.) since they do not help distinguish between two essays.

## Similarity Measures

The principle of similarity measurement between documents is a fundamental concept in information retrieval and text mining. It is commonly used in Natural Language Processing (NLP) applications like text summarization and machine translation. Data is collected from different sources like online reviews, emails, tweets, spreadsheets, and surveys (Qurashi et al., 2020). The primary goal of similarity measurements is to quantify the similarity of two documents or between a document and a query. In other words, the calculation of similarity is a function that measures the degree of similarity between two documents. All similarity measurements fall into the [-1, 1] or [0, 1] range. The minimal similarity is represented by 0 or -1, while absolute similarity is represented by 1 (Afzali & Kumar, 2017). Three types of similarity algorithms are employed in this layer:

1. **Overlap Similarity** is a measure of how close two sets are. It's determined by dividing the intersection size of two sets by the smaller size of them. If one set is a subset of the other, it is considered a full match (H.Gomaa & A. Fahmy, 2013). The overlap similarity between A and B is defined as,

$$O(A,B) = \frac{|A \cap B|}{min(|A|,|B|)} \qquad (1)$$

The degree of similarity measurement is between 0 and 1. When the two documents are identical, or one of them is a subset of the other, the value is 1; when the two documents are entirely different, the value is 0 (M.K & K, 2016).

2. **Cosine Similarity** is a measure that specifies how related documents are regardless of their size. Mathematically, it computes the cosine of the angle generated by two vectors projected in multidimensional space (Jain et al., 2020). The cosine similarity between A and B is known as,

$$\mathrm{C}(A,B) = \frac{A \cdot B}{\|A\| \times \|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \times \sqrt{\sum_{i=1}^{n} B_i^2}} \qquad (2)$$

The value of cosine differs between [-1, 1]. If two documents are identical, their vectors originate in the same direction, creating a slight angle with a cosine value nearby 1. Conversely, when two vectors point in opposite directions from the origin, they form a large angle, and the cosine value is close to -1; thus, the documents are dissimilar, and no similarity is mapped (Afzali & Kumar, 2018; Reddy et al., 2018).

3. **Jaccard Similarity** compares two sets for similarity. It is defined as the intersection size divided by the union size of two sets (Jain et al., 2020). The Jaccard similarity between A and B is referred to as,

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A|+|B|-|A \cap B|} \qquad (3)$$

A number between 0 and 1 represents the level of similarity. When the value is 1, two documents are identical; when the value is 0, two documents are dissimilar (Afzali & Kumar, 2018; Reddy et al., 2018).

## C) Third Layer

Clustering algorithms were applied to separate students' answers into several groups based on the number of k values. The questions types (features) that used in this layer are:

1. **A multiple-choice question (MCQ)** requests the respondent to select one or more options from a limited list. An MCQ includes the correct answer as well as distractors.
2. **A true & false question** is a statement that required a true or false answer. The true & false format can be used in a variety of forms such as "correct" or "incorrect", "yes" or "no" and "agree" or "disagree, etc.
3. **Calculated questions** are specific numerical questions that are based on a formula and use variables or "wild cards" (i.e. {a}, {b}). When the exam is taken, these wild cards are randomly selected from a collection of values.
4. **The numerical question** type needs a number as a response. The values are fixed in the question text.
5. **Questions with multiple answers** allow students to identify more than one choice. When there are multiple correct answers, this form of the question is used.
6. **A drag & drop question** contains a list of two or more potential responses, which can drag to response targets. The goal may be a table, a block, or any other element on the screen.

## Pre-processing of Third Layer

Before implementing the clustering algorithms, converting students' answers to an encoding format for these questions' types (multichoice, true & false, calculated, numerical, multi-answer, and drag & drop); because clustering algorithms deal with numerical data, not categorical data. For example, questions with two responses are converted to 0 and 1, while questions with three responses are transformed to 0, 1, 2, and so on.

## Details of the Dataset for Clustering Layer

We used 32 examinations from our dataset, specifically final exams from two semesters. The graphic presents the distribution of the different datasets in each exam. The number of attributes and instances are displayed in Figure 2.

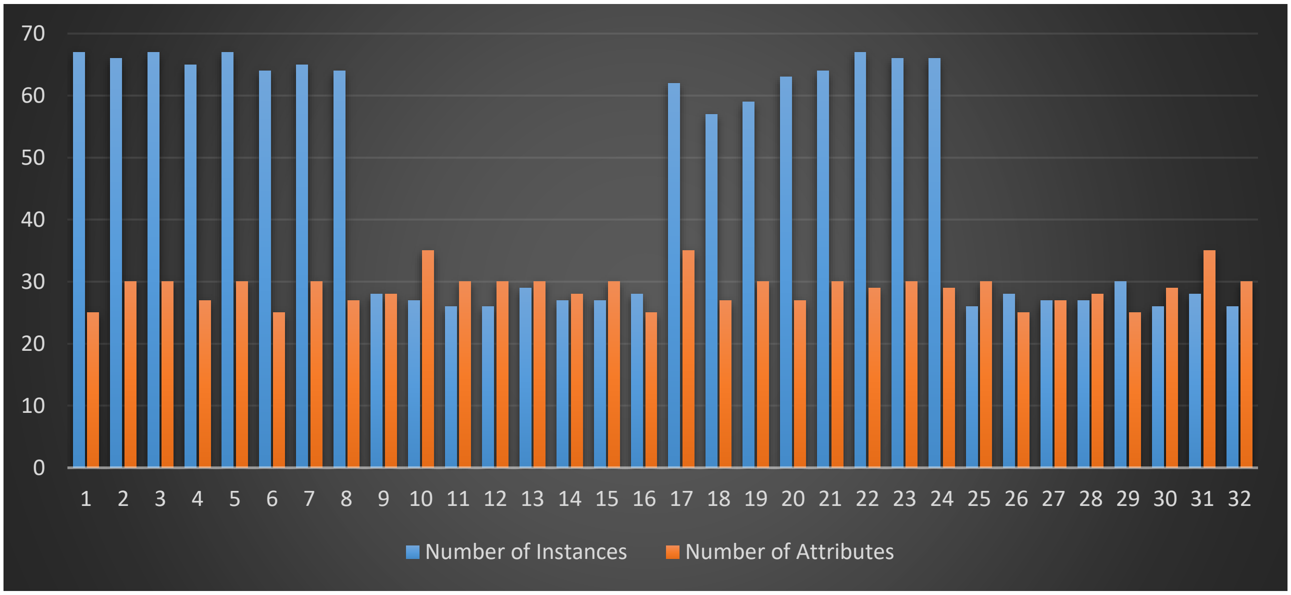


**Figure 2 Graphical Representation for the Number of Attributes and Instances**

The number of instances indicates the total number of students, while the number of attributes represents the total number of questions in each exam.

## Clustering Algorithms

Clustering is the process of grouping together similar data objects into clusters. Cluster analysis is used to summarize data, compact it, and find the nearest neighbors efficiently. Different types of clustering are partitional, hierarchical, overlapping, exclusive, fuzzy, complete, and partial. Clustering algorithms are divided into four types: prototype-based clustering, density-based clustering, scalable clustering algorithms, and graph-based clustering (Alzubaidi et al., 2021). Several important factors must be considered when selecting an effective clustering algorithm, like characteristics of clusters, type of clustering, number of data objects, characteristics of attributes and datasets, cluster description, noise & outliers, and domain-specific issues (Pandey et al., 2020).

Clustering categorizes a set of objects (typically defined as points in multidimensional space) into groups of related objects. Cluster analysis is a valuable component in data analysis. It resembles each other more than patterns from different clusters. The procedure for creating data clusters is shown in Figure 3 (RAMAKRISHNAN, n.d.):

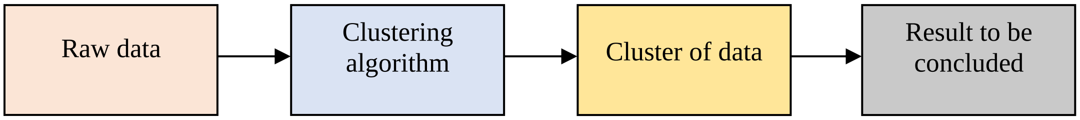


**Figure 3 The Process of Data Clustering**

In the beginning, we obtain raw data and apply a clustering algorithm to get data clusters. This is the process of using the clustering algorithm to create data clusters. Clustering is commonly used for unsupervised datasets, but it can also be used with supervised datasets.

Algorithms play a role in developing a well-designed clustering strategy for a particular problem in clustering. In this layer, three types of clustering algorithms are used:

## 1. K-Means Clustering Algorithm

k-means algorithm is simple unsupervised learning that works on iterations to group data objects into clusters to solve the well-known clustering problem. The process follows a simple and easy method for classifying a given data set using a specific number of clusters (suppose k clusters). The principal concept is to identify k centers, one for each group. These centers should be strategically placed because different locations produce different results. So, the best choice is to position them far from each other as much as possible. The next step is to associate each point in a dataset with the nearest center. When there are no pending points, the first stage is completed, and an early group age is finished. At this stage, we must re-calculate k new centroids as the barycenter of the clusters generated in the previous step. After obtaining these k new centroids, the same dataset points and the closest new data center have to be linked again. There has been created a loop. This loop means that the k centers change their position step by step until no changes have been made or that the centers no longer shift (Singh & Surya, 2015). Finally, the k-means algorithm aims to minimize an objective function known as the squared error function (Gnanapriya, 2017), which is defined as follows:

$$F = \sum_{i=1}^{n} \sum_{j=1}^{\mathrm{m}} \left(\|x_i - y_j\|\right)^2 \qquad (4)$$

Where,

$F$: represents the objective function.
$n$: represents the number of clusters.
m: represents the number of instances.
$\|x_i - y_j\|$: represents the Euclidean distance function.

K-means clustering algorithm steps (Kaur & Verma, 2017)

Let R = ($r_1$, $r_2$, …, $r_n$) be data points set and S = ($s_1$, $s_2$, …, $s_n$) be centers set.

1. The initial cluster centers 'c' is randomly chosen.
2. Compute the distance between all data points and cluster centers.
3. Allocate the information point to the cluster center with the shortest distance between it and all other cluster centers.
4. Use the following formula to re-calculate the new cluster center:

$$\boldsymbol{v}_i = (1/c_i) \sum_{j=1}^{c_i} x_i \qquad (5)$$

where 'ci' is the number of data points in i$^{th}$ cluster.

5. Re-calculate the distance of each data point to the new cluster centers.
6. Stop if no data points were reassigned; otherwise, start over at step 3.

## 2. Hierarchical Clustering Algorithm

A hierarchical clustering algorithm is one of the most common and simple clustering techniques, which forms a hierarchical cluster arrangement called a dendrogram. The dendrogram tree can be divided into several levels to generate different data clusters. This technique is divided into two types (agglomerative clustering and divisive clustering). The bottom-up approach is used in the agglomerative clustering algorithm. This clustering method assumes each document to be a single cluster, allowing all pairs of clusters to be combined into a single group containing all of the documents. On the other hand, the top-down approach is used in the divisive clustering algorithm—this method of clustering recursively separating the clusters from a single cluster to several groups (Kaur & Verma, 2017). Generally, merges and splits are calculated in a greedy manner.

### Hierarchical Clustering Algorithm Steps (Gnanapriya, 2017)

Given a collection of N items for clustering,

1. Begin by assigning each object to its cluster. If you have N items, you will now have N clusters, each including only one item. Let the distances between clusters to match the distances between the objects contained within them.
2. Find the most related (closest) pair of clusters and combine them into a single cluster, resulting in one less cluster.
3. Calculate the distances between each of the old clusters and the new clusters.
4. Steps 2 and 3 can be repeated until all items are grouped into a single N-size cluster.

### 3. Expectation-Maximization Clustering Algorithm

Expectation-Maximization (EM) algorithm is an iterative method for determining the maximum likelihood estimates of parameters in mathematical models that depend on unobserved latent variables (variables inferred from the values of other known variables but are not explicitly observable). The EM iteration alternates between doing an expectation (E) step, which calculates parameters maximizing the expected log-likelihood, and a maximization (M) step, which calculates parameters maximizing the expected log-likelihood found on the E step. In the next E step, these parameter estimates are used to calculate the distribution of the latent variables. EM gives a probability distribution to each case, which indicates the likelihood of it belonging to one of the clusters (Sehgal & Garg, 2014). This algorithm is the basis of many unsupervised clustering algorithms in machine learning, which is an extension of the k-means algorithm.

## Results

The previous section explained the proposed system and every feature & technique that used in this paper. The research results will be discussed in this section.

### 1. Data Collection Method

A private database was used in the proposed system provided by an Iraqi university without specifying the university's name for personal reasons. Table 1 shows the basic Moodle statistics of our dataset for the last two years.

**Table 1 Moodle Statistics for our Dataset**

| Item | Total |
|---|---|
| Number of courses | 180 |
| Number of students | 941 |
| Number of quizzes | 510 |
| Number of questions | 6645 |
| Number of assignments | 388 |
| Number of resources | 3064 |

As illustrated in Table 1, the dataset contains 941 participants, 510 exams, 180 courses, 3064 study resources, and 388 assignments for all stages in the first and second semesters. As a result, 32 final exams were used in our proposed system.

## 2. First Layer Results

A comprehensive description of the results is offered in Figure 4, which includes the total number of students and the number of students who cheated by (IP address, time taken, and time late).

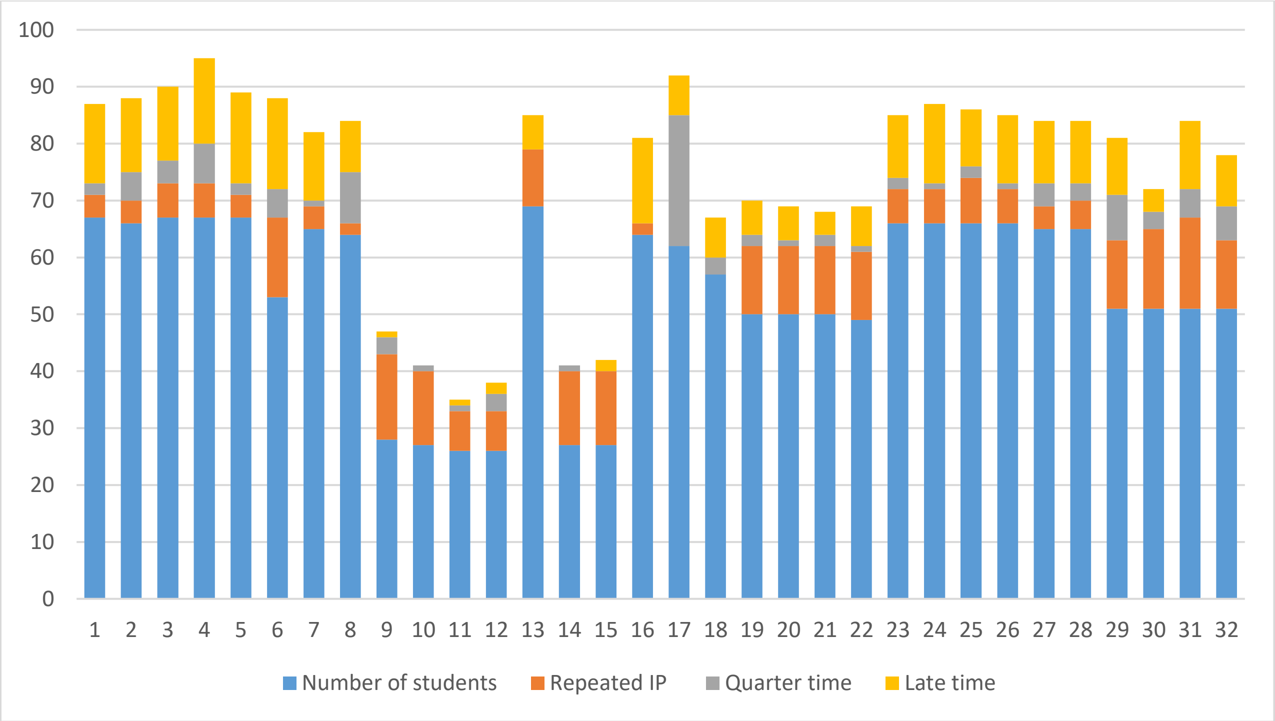


**Figure 4 The number of students identified as potential cheaters in the first layer**

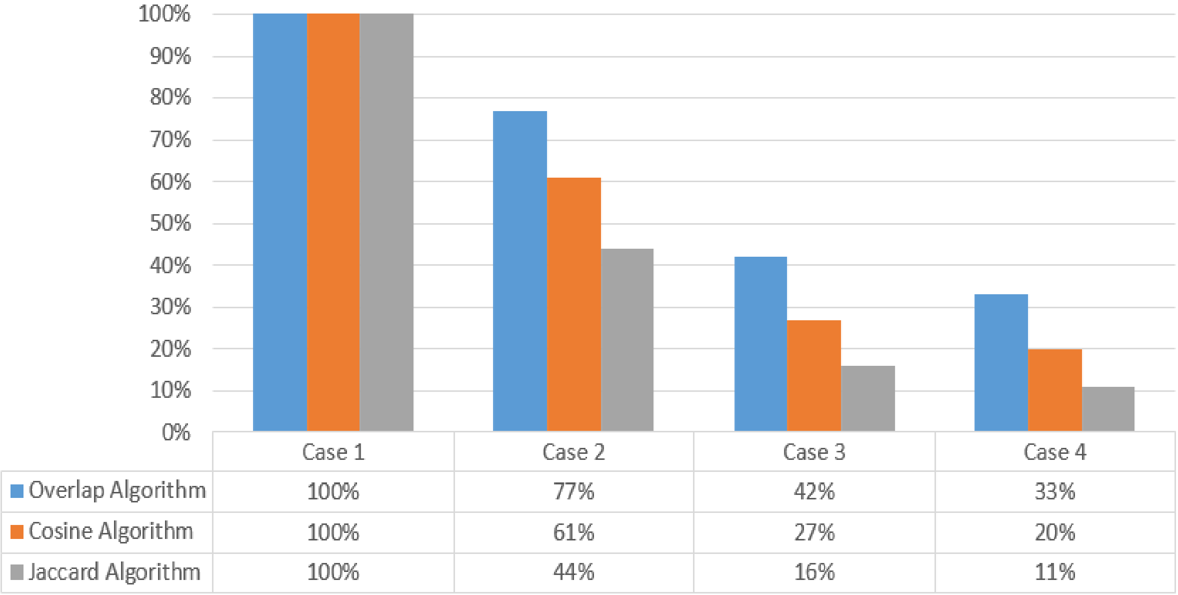
According to the above graph, which indicates the number of students who cheated in the first layer. The IP address had the highest rate of cheating since most of the students sit in the same location, followed by the time late and time taken.

## 3. Second Layer Results

To select the best similarity algorithm, four cases with different characteristics were considered for evaluation in this layer:

1. **First case** includes two similar documents.
2. **Second case** contains two documents. One of the documents involves a paragraph that exists entirely in the other document.
3. **Third case** includes two different documents on the same subject.
4. **Fourth case** contains two different documents.

In each of the four sentences, the preprocessing steps in the last section have been used. The overlap similarity, cosine similarity, and Jaccard similarity were applied. Figure 5 shows the results of all three algorithms.



| | Case 1 | Case 2 | Case 3 | Case 4 |
|---|---|---|---|---|
| 🟦 Overlap Algorithm | 100% | 77% | 42% | 33% |
| 🟧 Cosine Algorithm | 100% | 61% | 27% | 20% |
| ■ Jaccard Algorithm | 100% | 44% | 16% | 11% |

**Figure 5 The results of similarity measures**

All three measurements have a similarity of 100% in case 1, which contains exactly two identical documents. In cases 2, 3, and 4, the best result is provided by overlap similarity followed by cosine similarity and Jaccard similarity. However, the practical method and similarity measure is based on the characteristics of the experimental data and the work that users plan to do.

The results of the second layer are summarized in Figure (6) below, which include the total number of students and students who cheated in the essay questions.

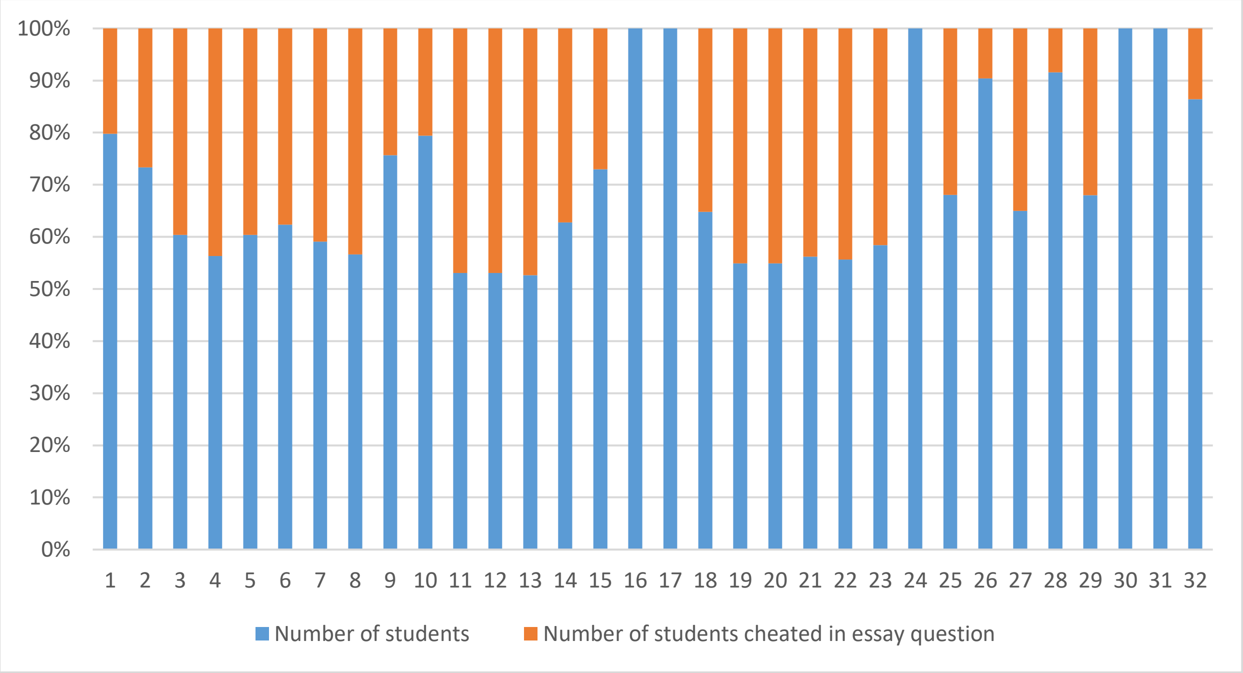


**Figure 6 Total number of Students and Students who Cheated in the Second Layer**

As illustrated in the figure, 27 of 32 exams were cheated by essay questions. As a result, this layer revealed more cheating than the previous one.

## 4. Third Layer Results

We compared three clustering algorithms (k-means, EM, and Hierarchical) based on the number of clusters, the sum of squared error (SSE), cluster instances, log-likelihood, and time is taken to build the model using the Weka (3.8.5) tool. Table 2 displays the results of our experiments while comparing clustering algorithms. The k value (the number of clusters) must be defined for each algorithm.

**Table 2 Results Comparison of Clustering Algorithms using the Weka Tool**

| Name | Exam-1 (number of instances is 66) | | | | | Exam-2 (number of instances is 68) | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Number of Clusters | Cluster Instances | Sum of Squared Error (SSE) | Log-likelihood | Time is taken to build the model (sec) | Number of Clusters | Cluster Instances | Sum of Squared Error (SSE) | Log-likelihood | Time is taken to build the model (sec) |
| K-Means | 2 | 28 (42%)<br>38 (58%) | 44.750 | | 0 | 3 | 30 (44%)<br>15 (22%)<br>23 (34%) | 32.481 | | 0 |
| EM | 2 | 15 (23%)<br>51 (77%) | | -11.50149 | 0.01 | 3 | 14 (21%)<br>53 (78%)<br>1 (1%) | | -0.5893 | 0.11 |
| Hierarchical | 2 | 65 (98%)<br>1 (2%) | | | 0.01 | 3 | 1 (1%)<br>66 (97%)<br>1 (1%) | | | 0.02 |
| K-Means | 4 | 18 (27%)<br>26 (39%)<br>12 (18%)<br>10 (15%) | 31.072 | | 0 | 5 | 21 (31%)<br>10 (15%)<br>14 (21%)<br>8 (12%)<br>15 (22%) | 28.163 | | 0 |
| EM | 4 | 17 (26%)<br>19 (29%)<br>20 (30%)<br>10 (15%) | | -4.36442 | 0.03 | 5 | 8 (12%)<br>48 (71%)<br>7 (10%)<br>1 (1%)<br>4 (6%) | | 2.2433 | 0.07 |
| Hierarchical | 4 | 63 (95%)<br>1 (2%)<br>1 (2%)<br>1 (2%) | | | 0.01 | 5 | 1 (1%)<br>64 (94%)<br>1 (1%)<br>1 (1%)<br>1 (1%) | | | 0.01 |
| K-Means | 6 | 14 (21%)<br>14 (21%)<br>8 (12%)<br>10 (15%)<br>10 (15%)<br>10 (15%) | 26.976 | | 0 | 7 | 16 (24%)<br>11 (16%)<br>12 (18%)<br>6 (9%)<br>11 (16%)<br>2 (3%)<br>10 (15%) | 24.588 | | 0 |
| EM | 6 | 15 (23%)<br>8 (12%)<br>15 (23%)<br>9 (14%)<br>9 (14%)<br>10 (15%) | | -4.4077 | 0.01 | 7 | 9 (13%)<br>27 (40%)<br>7 (10%)<br>1 (1%)<br>11 (16%)<br>3 (4%)<br>10 (15%) | | 5.43905 | 0.04 |
| Hierarchical | | 60 (91%)<br>1 (2%)<br>1 (2%)<br>1 (2%)<br>1 (2%)<br>2 (3%) | | | 0.01 | 7 | 1 (1%)<br>62 (91%)<br>1 (1%)<br>1 (1%)<br>1 (1%)<br>1 (1%)<br>1 (1%) | | | 0.01 |

The best result was obtained from selected clustering algorithms to evaluate our dataset (k-means followed by EM and hierarchical algorithms). The k-means algorithm performed best with execution time and clustered instances compared to EM and hierarchical algorithms. Figure 7 shows the results of cluster algorithms compared in terms of time complexity.

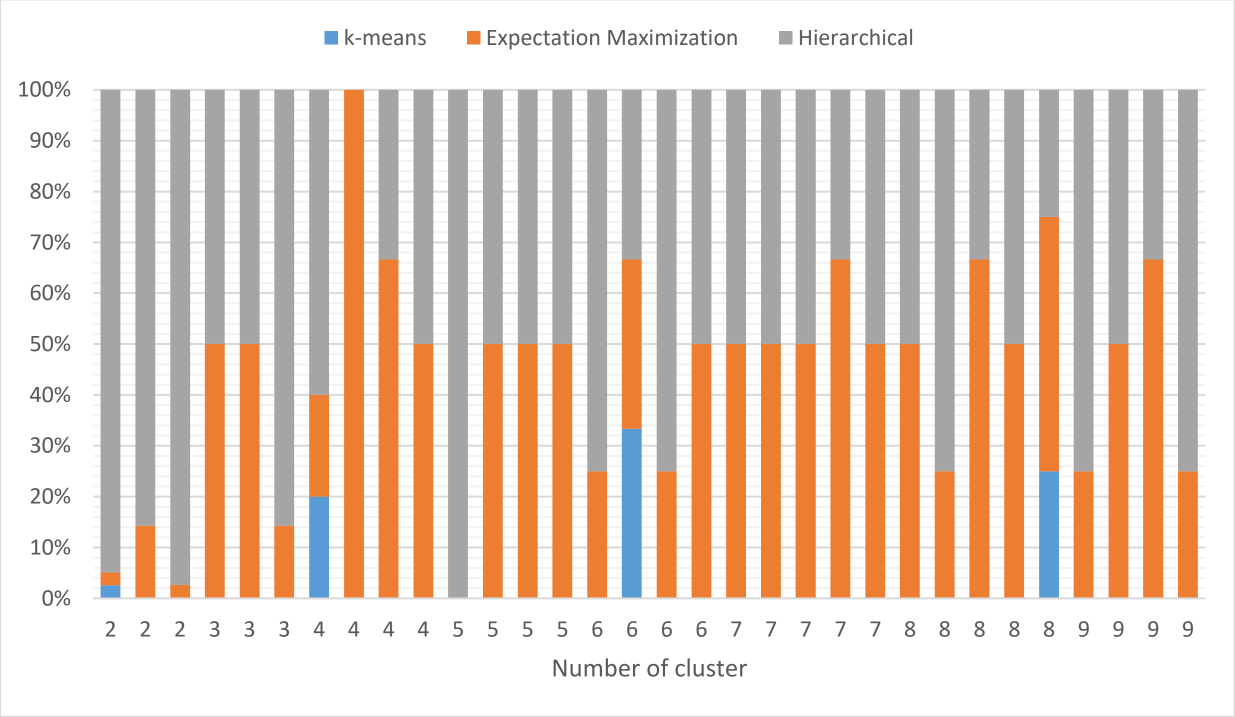


**Figure 7 Time Taken for the Clustering Algorithms**

The results of the simple K-mean, hierarchical, and EM were compared in terms of time complexity on the 32 exams of our datasets. The k-means algorithm has the minimum execution time compared to other algorithms.

## 5. Result Implementation

As one of the experiment results of the proposed system, the examination status was evaluated after the exam was finished. A sample from the existing exams were selected for the fourth stage of the first semester. Table 3 displays the proposed system's results.

**Table 3 Result Implementation of the Proposed System**

| ***First layer*** ***(Statistical layer)*** | | | | ***Second layer*** ***(Similarity layer)*** | ***Third layer (Clustering layer)*** |
|---|---|---|---|---|---|
| **Student ID** | **IP Address** | **Time Taken** | **Time Late** | **Ratio of similarity** | **Student' groups** |
| 241 | Repeated IP (176.10.99.200) | used quarter time of exam: (00:37:20) | on time | Low matching | **Group – 1**<br>374, 241, 673, 240, 612, 236, 242, 216 |
| 612 | unique IP | used full time of exam | on time | High matching (94%) between 612 and 245 | |
| 635 | unique IP | used full time of exam | Time late is: (00:36:38) | High matching (100%) between 635 and 670 | |
| 245 | unique IP | used full time of exam | on time | High matching (94%) between 245 and 612 | |
| 216 | unique IP | used full time of exam | on time | High matching (89%) between 216 and 673 | |
| 929 | unique IP | used full time of exam | on time | High matching (96%) between 929 and 612 | |
| 242 | Repeated IP (176.10.99.200) | used full time of exam | on time | High matching (96%) between 242 and 671 | |
| 670 | Repeated IP (185.121.69.40) | used full time of exam | on time | High matching (100%) between 670 and 635 | |
| 673 | unique IP | used quarter time of exam: (00:25:51) | on time | High matching (91%) between 673 and 245 | **Group – 2**<br>635, 672, 222, 670, 227, 671 |
| 672 | Repeated IP (185.220.103.5) | used full time of exam | Time late is: (00:21:50) | Low matching | |
| 223 | Repeated IP (51.15.82.176) | used full time of exam | on time | High matching (94%) between 223 and 612 | |
| 240 | Repeated IP (176.10.99.200) | used full time of exam | on time | Low matching | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 215 | Repeated IP (51.15.82.176) | used quarter time of exam: (00:36:51) | on time | | Low matching | | |
| 643 | unique IP | used full time of exam | Time late is: (00:25:23) | | Low matching | | |
| 226 | unique IP | used full time of exam | on time | | Low matching | | **Group – 3**<br>215, 217, 223, 219 |
| 243 | unique IP | used full time of exam | on time | | High matching (92%) between 243 and 612 | | |
| 220 | unique IP | used full time of exam | on time | | High matching (94%) between 220 and 242 | | |
| 221 | unique IP | used full time of exam | Time late is: (00:33:07) | | Low matching | | |
| 374 | Repeated IP (176.10.99.200) | used full time of exam | on time | | High matching (89%) between 374 and 673 | | |
| 361 | unique IP | used full time of exam | on time | | Low matching | | |
| 236 | unique IP | used full time of exam | on time | | High matching (94%) between 236 and 612 | | **Group – 4**<br>245, 929, 243, 220 |
| 217 | Repeated IP (51.15.82.176) | used full time of exam | Time late is: (00:17:33) | | Low matching | | |
| 671 | Repeated IP (185.220.103.5) | used full time of exam | on time | | High matching (96%) between 671 and 242 | | |
| 286 | unique IP | used full time of exam | Time late is: (00:27:10) | | High matching (90%) between 286 and 220 | | |
| 222 | Repeated IP (185.121.69.40) | used quarter time of exam: (00:31:51) | Time late is: (00:20:38) | | Low matching | | **Group – 5**<br>643, 226, 221, 361, 286 |
| 219 | Repeated IP (51.15.82.176) | used full time of exam | on time | | High matching (89%) between 219 and 223 | | |
| 227 | Repeated IP (185.121.69.40) | used full time of exam | on time | | Low matching | | |

This exam was taken by 27 students, and the following results were obtained after implementing our proposed system:

1. In the first layer, 13 duplicate IP addresses were discovered and divided into four groups, meaning that each group of students (241, 374, 240, 242), (670, 222, 227), (672, 671), and (219, 223, 215, 217) sat in the same place to take the exam. In addition, four students (241, 673, 215, 222) completed the exam within the quarter-time limit, whereas seven students (635, 672, 643, 221, 217, 286, 222) were late for taking the exam on time.
2. In the second layer, 16 students had a high matching of answers with other students, while 11 students had a low matching.
3. In the third layer, we divided students' answers into five groups (the number of clusters is five), each group containing several students who had similar responses.
4. Cheating was detected for some students in all three layers, like (242,374), who cheated by using a shared IP address and obtained a high match rate in the second layer, as well as being isolated in the same group in the third layer.
5. Some students did not cheat in the first layer, but cheated in the second layer like (236, 612) and then were isolated in the same group in the third layer.
6. Most of the cheating cases were detected in the second layer; 16 cheats were identified out of 27 students.
7. Some students did not cheat in the first and second layers such as (226,361), but they were grouped in the third layer. This indicates that the students did not cheat or their responses were similar, as showed by the third layer's result. In this instance, the examiner determines whether the students' status is cheating or not.

## Conclusion

Students and educational institutions have paid a lot of attention to E-learning and distance education in the Covid-19 pandemic. E-learning has grown in popularity around the world due to its flexibility, accessibility, and user-friendliness. However, the primary challenge in online education is assessing students in the online exam because cheating in the examination is simple and a significant issue in education and undermining efforts to evaluate a student's performance.

In this paper, a solution was proposed to reduce cheating during online exams by extracting a set of reliable features from the Moodle platform using data mining techniques. These features are divided into three layers. In the first layer, statistical methods were used to these features (IP address, time taken, and time late) to detect cheating in the exam. In the second layer, similarity measurements were applied to essay questions to calculate the ratio of similarity between students' answers. In the third layer, clustering algorithms were employed to these questions (multichoice, true & false, calculated, numerical, multi-answer, and drag & drop) to divide students' answers into several related groups. Finally, a recommendation system is presented to assist the examiner in deciding suspicious students' responses. As a consequence of the proposed system, the following points have been identified:

1. Some students cheated at the first layer by sitting in the same location, which was identified by their identical IP address. They also cheated in the second layer, in which they had a high similarity ratio. However, the clustering algorithm grouped them in the third layer.
2. Some students cheated by time (time taken or time late), and they didn't sit in the same location. They also detected in the similarity and cluster algorithms.
3. Some students cheated in the first layer, either by IP address or time, and they were also detected in the second and third layers with other students who did not exist in the first layer.
4. The overlap similarity was the best method in the second layer since it has the highest accuracy compared to other algorithms according to sets of different characteristics.
5. The best algorithm in the third layer was the k-means algorithm, which required less time to execute and achieved the best clustering instances. As a result, when the number of clusters is large, the SSE is lower, and the clustering instances are better.
6. The second layer had the most significant rate of cheating, followed by the first and third layers.
7. The proposed system reported that 60% of students cheated in the second semester, while 40% cheated in the first semester. Students cheat at all educational stages, with the fourth stage cheating is 32%, the third stage is 30%, the second stage is 24%, and the first stage is 14%. Based on 68 final exams, Figure 8 shows the number of students who cheated at all academic levels.

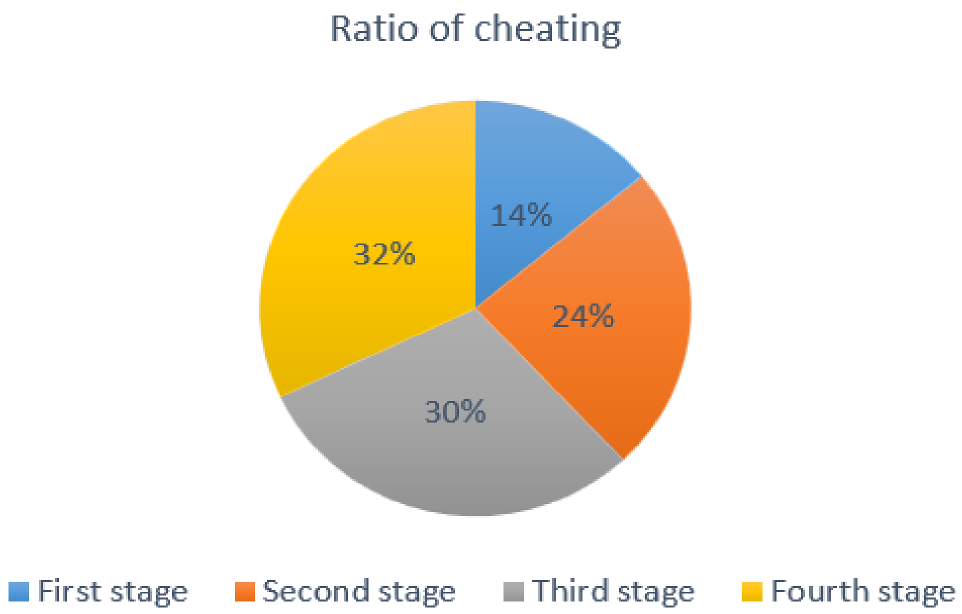


**Figure 8 Ratio of Cheating for all Stages**

## References

Charan, A., Darshan, D., Madhu, N., & Manjunatha, B.S.A. (2020). a Survey on Detection of Anomalous Behaviour in Examination Hall. *International Journal of Engineering Applied Sciences and Technology*, *5*(2), 583–588. https://doi.org/10.33564/ijeast.2020.v05i02.098

Afzali, M., & Kumar, S. (2017). Comparative Analysis of Various Similarity Measures for Finding Similarity of Two Documents. *International Journal of Database Theory and Application*, *10*(2), 23–30. https://doi.org/10.14257/ijdta.2017.10.2.02

Afzali, M., & Kumar, S. (2018). An Extensive Study of Similarity and Dissimilarity Measures Used for Text Document Clustering using K-means Algorithm. *IJ Inf. Technol. Comput. Sci*, *9*, 64–73.

Alzubaidi, L., Zhang, J., Humaidi, A.J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M.A., Al-Amidie, M., & Farhan, L. (2021). Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *Journal of Big Data*, *8*(1), 1–74.

Bawarith, R., Basuhail, A., Fattouh, A., & Gamalel-Din, S. (2017). E-exam cheating detection system. *International Journal of Advanced Computer Science and Applications*, *8*(4), 176-181. https://doi.org/10.14569/ijacsa.2017.080425

Chuang, C.Y., Craig, S.D., & Femiani, J. (2017). Detecting probable cheating during online assessments based on time delay and head pose. *Higher Education Research and Development*, *36*(6), 1123–1137.

https://doi.org/10.1080/07294360.2017.1303456

Karthika, R., Vijayakumar, P., Rawal, B.S., & Wang, Y. (2019). Secure Online Examination System for e-learning. *In IEEE Canadian Conference of Electrical and Computer Engineering (CCECE)*, 1-4.

https://doi.org/10.1109/CCECE43985.2019.9052408

Ghizlane, M., Hicham, B., & Reda, F.H. (2019). A New Model of Automatic and Continuous Online Exam Monitoring. *Proceedings 4th International Conference on Systems of Collaboration, Big Data, Internet of Things and Security, SysCoBIoTS*, 1–5. https://doi.org/10.1109/SysCoBIoTS48768.2019.9028027

Gnanapriya, S. (2017). Evaluation of Clustering Capability Using Weka Tool. *International Journal of Innovations in Engineering and Technology*, *8*(1), 181–187.

https://doi.org/10.21172/ijiet.81.025

Golden, J., & Kohlbeck, M. (2020). Addressing cheating when using test bank questions in online Classes. *Journal of Accounting Education*, *52*.

https://doi.org/10.1016/j.jaccedu.2020.100671

H.Gomaa, W., & A. Fahmy, A. (2013). A Survey of Text Similarity Approaches. *International Journal of Computer Applications*, *68*(13), 13–18.

https://doi.org/10.5120/11638-7118

https://moodle.org/. (n.d.).

Hu, S., Jia, X., & Fu, Y. (2018). Research on Abnormal Behavior Detection of Online Examination Based on Image Information. *Proceedings 10th International Conference on Intelligent Human-Machine Systems and Cybernetics, IHMSC, 2*, 88–91.

https://doi.org/10.1109/IHMSC.2018.10127

Jain, G., Mahara, T., & Tripathi, K.N. (2020). A Survey of Similarity Measures for Collaborative Filtering-Based Recommender System. *In Advances in Intelligent Systems and Computing*, *1053*, 343–352.

https://doi.org/10.1007/978-981-15-0751-9_32

Kaur, H., & Verma, P. (2017). Comparative Weka Analysis of Clustering Algorithm‘s. *International Journal of Information Technology and Computer Science*, *9*(8), 56–67. https://doi.org/10.5815/ijitcs.2017.08.07

Ketab, S.S., Clarke, N.L., & Dowland, P.S. (2017). A Robust e-Invigilation System Employing Multimodal Biometric Authentication. *International Journal of Information and Education Technology*, *7*(11), 796–802.

https://doi.org/10.18178/ijiet.2017.7.11.975

M.K.Vijaymeena, M.K., & Kavitha, K. (2016). A Survey on Similarity Measures in Text Mining. *Machine Learning and Applications: An International Journal*, *3*(1), 19-28. https://doi.org/10.5121/mlaij.2016.3103

Mahadi, N.A., Mohamed, M.A., Mohamad, A.I., Makhtar, M., Kadir, M.F.A., & Mamat, M. (2018). A Survey of Machine Learning Techniques for Behavioral-Based Biometric User Authentication. *Recent Advances in Cryptography and Network Security*, 43–54. https://doi.org/10.5772/intechopen.76685

Mungai, P.K., & Huang, R. (2017). Using keystroke dynamics in a multi-level architecture to protect online examinations from impersonation. *IEEE 2nd International Conference on Big Data Analysis, ICBDA*, 622–627. https://doi.org/10.1109/ICBDA.2017.8078710

Muzaffar, A.W., Tahir, M., Anwar, M.W., Chaudry, Q., Mir, S.R., & Rasheed, Y. (2021). A systematic review of online exams solutions in e-learning: Techniques, tools, and global adoption. *IEEE Access*, *9*, 32689–32712.

https://doi.org/10.1109/ACCESS.2021.3060192

Pandey, S.K., Mishra, B., & Gautam, S.S. (2020). Cluster Based Mining for Prediction of Heart Disease. *International Journal of Computer Science and Mobile Computing*, *9*(7), 136-143.

Prathish, S., Athi Narayanan, S., & Bijlani, K. (2017). An intelligent system for online exam monitoring. *Proceedings International Conference on Information Science, ICIS*, 138–143. https://doi.org/10.1109/INFOSCI.2016.7845315

Qurashi, A. W., Holmes, V., & Johnson, A. P. (2020). Document Processing: Methods for Semantic Text Similarity Analysis. *INISTA 2020 - 2020 International Conference on Innovations in Intelligent Systems and Applications, Proceedings*, 1–6.

https://doi.org/10.1109/INISTA49547.2020.9194665

Ramakrishnan, R. (n.d.). *A Survey on Students Placement Performance Analysis Using Weka Tool*.

Reddy, K.P., Reddy, T.R., Naidu, G.A., & Vishnu, B. (2018). *Impact of Similarity Measures in Information Retrieval*, 54–59.

Sehgal, G., & Garg, D. (2014). Comparison of Various Clustering Algorithms. *International Journal of Computer Science and Information Technologies*, *5*(3), 3074–3307.

Shdaifat, A., Obeidallah, R., Ghazal, G., Srhan, A.A., & Abu Spetan, N.R. (2020). A proposed iris recognition model for authentication in mobile exams. *International Journal of Emerging Technologies in Learning*, *15*(12), 205–216. https://doi.org/10.3991/ijet.v15i12.13741

Singh, P., & Surya, A. (2015). Performance analysis of clustering algorithms in data mining in weka. *International Journal of Advances in Engineering & Technology*, *7*(6).