**Objective evaluation of performances in case of Students**

**based on similarity analyses and Moodle-logs**

**Shagai Turtogtokh**

Kodolányi János University and MY-X research team Hungary

**László Pitlik**

Kodolányi János University and MY-X research team Hungary

**László Pitlik (Jr)**

Kodolányi János University and MY-X research team Hungary

**ABSTRACT**

The increasing adoption of e-learning platforms has revolutionized educational practices and generated rich log data that can be harnessed to evaluate student performance objectively. This study proposes a comprehensive model that leverages 29 distinct attributes extracted from Moodle (e-learning platform) log data to provide a multifaceted/objective evaluation of student performance.

These attributes, which capture aspects such as diligence (e.g., posting frequency, active days, etc.), understanding (e.g., topic relevance, citation usage), and interaction dynamics (e.g., reply time, response length), are organized into an Object Attribute Matrix (OAM), where each attribute's type and direction (e.g. the less is the reply-time, the better is the performance, etc.) are defined.

To quantify topic alignment, the all-MiniLM-L6-v2 model is used to generate sentence embeddings, measuring semantic coherence between student responses and instructor posts via cosine similarity. Suspected AI-generated content is identified using the roberta-base-openai-detector, which assigns a 1–10 score (10 indicating high probability). The framework utilizes the COCO Y0 engine (https://miau.my-x.hu/myx-free/index\_en.php3) — a computational analysis tool — to assess attribute impacts and rank students based on composite performance metrics.

This data-driven approach offers educators an objective framework for assessing student performance while delivering personalized feedback to enhance learning outcomes. Future research will validate the model's objectivity and effectiveness across diverse educational contexts.

**Keywords:** AI, anti-discrimination, optimization, automation

**INTRODUCTION**

The rapid growth of e-learning platforms has transformed how individuals acquire knowledge and develop skills, generating vast amounts of data that offer valuable insights into learning behaviors. According to Statista[[1]](#endnote-1), the global e-learning market is projected to reach nearly $400 billion by 2026, doubling from $200 billion in 2019. This expansion reflects the increasing reliance on digital learning solutions and highlights the potential for data-driven insights to enhance educational outcomes. By examining behaviors such as engagement, consistency, collaboration, and comprehension, educators can personalize learning experiences, identify gaps in understanding, and assess the effectiveness of course design.

However, analyzing such data presents challenges. The large size and complexity of learning data makes it difficult to extract meaningful insights without advanced analytical methods. Additionally, concerns around privacy, data integrity, and algorithmic fairness require careful consideration to ensure evaluations are both accurate and ethical.

This research seeks to address these challenges by utilizing Moodle log data, advanced natural language processing (NLP) models, and an anti-discriminative engine to analyze student performance with 29 different attributes that reveal multiple dimensions of student activity. This approach aims to uncover meaningful insights into engagement patterns and learning behaviors, offering a data-driven framework for improving educational strategies and outcomes. As educational technology continues to evolve, such insights will play a vital role in shaping the future of learning, empowering educators to better support students in achieving their full potential.

**MATERIALS AND METHOD**

Moodle eLearning platform provided us valuable log data of the students' discussions and activities in the platform (c.f. Figure-1):

Figure 1. The structure of the raw log data from the Moodle platform - (source: own presentation)



Used raw log data <https://miau.my-x.hu/miau/315/moodle/discussion.xlsx>

With such complex raw data being provided, relevant attributes for each student can be extracted using queries and other technologies. We extracted 29 chosen attributes that are divided into two categories (24 for diligence and 5 for understanding):

Diligence/ quantitative attributes:

1. total\_posts - Total number of posts
2. active\_days - Number of days that student actively participated in discussions
3. total\_replies\_to\_professor - Number of replies to professor's post
4. total\_characters - Total chars of all posts of the student
5. total\_words - Total words of all posts of the student
6. avg\_words - Average words per post
7. unique\_interactions - The number of unique interactions with other students for each user
8. unique\_discussions - the number of unique discussions for each user
9. engagement\_rate - The attribute measures student engagement by dividing their replies by the professor's total posts, showing how actively they participate in the professor's discussions.
10. normanlized\_score - The used query normalizes total replies and average reply time for all students, assigning a higher weight (70%) to replies and a lower weight (30%) to reply time. This weighting emphasizes the importance of engagement (quantity of replies) over responsiveness (speed of replies) when calculating an overall performance score. The normalized scores ensure fair comparisons regardless of differing scales for the two metrics.
11. deadline\_exceeded(Quasi exam I) - Posts created after given deadline in quasi exam I
12. deadline\_exceeded(Quasi exam II) - Posts created after given deadline in quasi exam II
13. deadline\_exceeded(Quasi exam III) - Posts created after given deadline in quasi exam III
14. consistency\_score - How consistently a student participates in discussions over time. standard deviation based on number of replies per week.
15. max\_streak - Maximum number of activity streak
16. avg\_reply\_time – Average reply time to the professor
17. avg\_charcount - Avarage char count of all post
18. max\_charcount – The longest post char count
19. min\_charcount -
20. modification\_count - How many times student modified his post
21. avg\_modified\_time\_minutes - How many minutes did it take to modify their post after posting
22. average\_posts\_per\_day - Average number of posts per day
23. response\_time\_in\_hours(Task-I) - response\_time\_in\_hours(Task-I)
24. response\_time\_in\_hours(Task-II) - Amount of time to answer Task II

Understanding/ qualitative attributes:

1. Pattern\_followed(quasi exam i) - The number of posts where a student followed a specific pattern provided by the professor (with a maximum of 2 and a minimum of 0)
2. avg\_AI\_involvedMsg\_score - The average AI involvement score is calculated using the "roberta-base-openai-detector" model, which detects AI-generated text. It assigns a score for each response (scaled from 1-10 the more the probability of AI involvement), and averaging these scores indicates the overall prevalence of AI-generated content.
3. topic\_relevance\_score - The Topic Relevance Score for every student's reply to the professor's post using a pre-trained NLP model (all-MiniLM-L6-v2 ) and cosine similarity.
4. citation\_count - How often students use external references (e.g., links, citations) in their posts to measure research-oriented behavior.
5. valid\_response - Quasi exam task I has the correct answer 37 and Task II has the correct answer 5. If the student answers the question correctly for both of the questions then 1 else 0

All these actionable insights are gathered through various queries, computations, and model interpretations. For instance, the query for attribute “deadline\_exceeded(Quasi exam II)” is shown in Figure 2:

Figure 2. Query for extracting the attribute “deadline\_exceeded(Quasi exam II)”

SELECT

 userid, userfullname, COUNT(\*) AS posts\_after\_deadline

FROM

 discussions\_m

WHERE

 subject LIKE '%Quasi Exam II%' -- Filter for posts related to "Quasi Exam III"

 AND created > '2024-11-08 24:00:00' -- Filter posts made after the deadline

GROUP BY

 userid, userfullname

ORDER BY

 userfullname;

All other used queries: <https://miau.my-x.hu/miau/315/moodle/QUERIES.docx>

Most of the attributes that are about diligence can be extracted using queries and codes but we need to analyze text inputs to get the more detailed insights of the student. To get such insights following NLP models are utilized.

Topic relevance score: all-MiniLM-L6-v2

all-MiniLM-L6-v2 is a pre-trained language model specifically designed for generating high-quality sentence embeddings. This means it can convert sentences and paragraphs into numerical representations (vectors) that capture their semantic meaning. Additionally, its fast inference speed allows for scalable analysis of large volumes of student submissions.

Steps:

Importing the libraries:

import sqlite3

import pandas as pd

from sentence\_transformers import SentenceTransformer

from sklearn.metrics.pairwise import cosine\_similarity

Prepare the Data:

* Extract the professor's posts (parent IDs and messages).
* Extract student replies corresponding to each parent post (message where parent = prof\_id).

prof\_query = "SELECT id AS prof\_id, message AS prof\_message FROM discussions\_m WHERE userid = 34004"  *# Professor's posts*

reply\_query = "SELECT id, parent, userid, userfullname, message AS student\_message FROM discussions\_m WHERE parent IN (SELECT id FROM discussions\_m WHERE userid = 34004)"  *# Student replies*

prof\_posts = pd.read\_sql\_query(prof\_query, conn)

student\_replies = pd.read\_sql\_query(reply\_query, conn)

Embed the Messages:

* Use a pre-trained sentence embedding model (e.g., SentenceTransformer).
* Generate embeddings for both professor's posts and student replies.

prof\_posts['embedding'] = prof\_posts['prof\_message'].apply(**lambda** x: model.encode(x) if isinstance(x, str) and x.strip() else None)

*# Filter out replies with invalid parent IDs*

student\_replies = student\_replies[student\_replies['parent'].isin(prof\_posts['prof\_id'])]

Compute Similarity:

* For each student reply, calculate the cosine similarity between the reply's embedding and the professor's post's embedding.

student\_replies['topic\_relevance\_score'] = student\_replies.apply(

    **lambda** x: calculate\_similarity(x, prof\_posts) if x['student\_message'] else None, axis=1

)

Store and Output Scores:

* Save the relevance score for each student reply in the database or output the results as a report.

merged\_df[['id', 'topic\_relevance\_score']].to\_sql('relevance\_scores', conn, if\_exists='replace', index=False)

conn.close()

Full source code:[https://miau.my-x.hu/miau/315/moodle/Topic%20relevance%20score(code).docx](https://miau.my-x.hu/miau/315/moodle/Topic%20relevance%20score%28code%29.docx)

Predicted AI-generated text: roberta-base-openai-detector

Nowadays, lots and lots of educational institutions use AI detector tools for their system so including this metric in this learning assessment is a must. Therefore, the roberta-base-openai-detector model is used for student AI text prediction due to its strong contextual understanding, fine-tuning for AI-generated content detection, and reduced false positive rates.

Steps:

1. Prepare the Environment – getting all important Python libraries. Load and Preprocess Data.
2. Function to clean text (Remove punctuation, Normalize whitespace)
3. Analyze Text Complexity (Calculate readability metrics)
4. Predict AI probability
5. Combine Attributes into a Unified Score
6. Scale to a 1-10 rating
7. Pull the result into the database

Full source code: [https://miau.my-x.hu/miau/315/moodle/(Code)Detecting%20AI%20generated%20text%20using%20pretrained%20model.docx](https://miau.my-x.hu/miau/315/moodle/%28Code%29Detecting%20AI%20generated%20text%20using%20pretrained%20model.docx)

With all 29 attributes extracted using queries and model interpretations, we can create an Object attribute matrix (OAM, c.f. Figure-3):

Figure 3. OAM (24 objects, 29 attributes) - (source: own presentation)



Reference: <https://miau.my-x.hu/miau/315/moodle/OAM_student_analysis_Moodle.xlsx>

OAM rules:

Each of the 29 attributes in our model is assigned a directional value to indicate its impact on performance. A direction value of 0 signifies that more is better, while a value of 1 indicates that less is better. For example, attributes such as the number of posts or replies to the professor are positively correlated with better performance, so they are assigned a direction of 0. Conversely, the AI involvement score, which reflects the likelihood of machine-generated responses, negatively impacts performance and is thus assigned a direction of 1.

To standardize performance measurement, each student begins with a baseline performance index of 1000. This starting point ensures that performance scores can increase or decrease without introducing negative values, which the linear programming tool we employed is less suited to handle. While higher baseline values, such as 10,000 or 1,000,000, could offer even finer granularity, the choice of 1000 provided sufficient precision for our analysis while maintaining clear, interpretable results.

This structured scoring system enables the creation of a ranked table using Excel Rank() function derived from the Object Attribute Matrix (OAM), which is then processed using the COCO Y0 anti-discriminative model to assess students individually and compare student performances (c.f. Figure-4):

Figure 4. Ranked values compared to a performance index: 1000 (norm-value) - (source: own presentation)

 

Reference: <https://miau.my-x.hu/miau/315/moodle/OAM_student_analysis_Moodle.xlsx>

COCO Y0 Engine (https://miau.my-x.hu/myx-free/index\_en.php3):

The COCO Y0 engine is a computational tool designed to evaluate and compare objects based on multiple attributes while ensuring objective and anti-discriminative analysis. By leveraging data organized in an Object Attribute Matrix (OAM), the engine calculates performance indices through a set of predefined conditions to maintain fairness and consistency.

In addition to the direction vector (0,1) and integer-based Y-values, which have already been discussed, the COCO Y0 engine operates under the following key principles[[2]](#endnote-2):

* No Preset Weight Relations: The model does not require predefined relationships between attribute weights or fixed intervals between performance levels (referred to as "stairs"), ensuring unbiased calculations.
* Additive Logic: If an attribute has an extreme value (e.g., zero), the corresponding Y-value will not be zero. Instead, the system maintains additive logic, where each attribute contributes proportionally to the overall score.
* Column Filtering: Identical attribute columns are excluded to improve computational efficiency without compromising the overall evaluation.
* Genetic Potential Interpretation: The sum of the initial values in each attribute column can be interpreted as a form of 'potential,' ensuring all evaluated entities start on an equal footing.

Adhering to these principles makes the COCO Y0 engine suitable for analyzing complex data structures like student performance metrics (c.f. Figure-5):

Figure 5. Error-free estimations (source: own presentation)



Reference: <https://miau.my-x.hu/miau/315/moodle/OAM_student_analysis_Moodle.xlsx>

In Figure 5, the COCO Y0 engine produces results where all evaluated objects receive the same score, often matching the pre-defined baseline value (e.g., 1000 points). This outcome suggests that the model's anti-discriminative logic has functioned effectively to ensure fairness, but it also indicates that some critical differences between objects remain obscured. In such cases, attribute exclusion becomes necessary to refine the evaluation.

The model’s anti-discriminative logic may hide certain attributes that had minimal or no influence in the initial run. These are often attributes that contribute evenly across all objects. The next step is to intentionally remove the attributes that were previously dominant in the initial evaluation. By excluding those attributes, the focus shifts to the previously obscured characteristics, which often carry the subtle yet crucial differences between objects.

To do that, In Figure 6 we can see the stairs2 table and the s1 row values starting from 23 in certain attributes are the obscured attributes (c.f. Figure-6):

Figure 6. COCO result (stairs2 table hidden attributes) (source: own presentation)

 

Reference: <https://miau.my-x.hu/miau/315/moodle/OAM_student_analysis_Moodle.xlsx>

Once we exclude the attributes, we run the COCO engine for the second time and get the refined results as shown in Figure 7:

Figure 7-8. Result and rankings after attribute exclusion (source: own presentation)

 

Reference (Figure-7): <https://miau.my-x.hu/miau/315/moodle/OAM_student_analysis_Moodle.xlsx>

Reference (Figure-8): <https://miau.my-x.hu/miau/315/moodle/OAM_student_analysis_Moodle.xlsx>

Validation of Results

To ensure the reliability and accuracy of our evaluation results, we applied a validation process based on symmetry effects. This method verifies whether the differences in attribute values between students align consistently with their performance rankings, reinforcing the model's predictive reliability.

The validation process:

Reverse Ranking: We first reverse the original ranking order using the formula:

$$Reversed rank=Number of Objects-Original Rank+1$$

Re-Evaluation with COCO Engine: After creating this flipped ranking table, we ran the COCO Y0 engine on the adjusted data.

Delta Calculation: We compute a key metric known as the product of the original delta values and the inverted delta values. This metric acts as a critical indicator of the model's consistency.

$$1st\frac{delta}{fact} \* Inverted\frac{delta}{fact}$$

The result is interpreted according to the following rule:

If the product of the two delta values is zero or less, the model’s results are confirmed to be valid and reliable. If the product is greater than zero, it indicates potential inconsistencies. This could signal errors in certain students' data or weaknesses in the model itself.

With such a validation process we improve our confidence in the model’s accuracy and ensure that the ranking outcomes reflect meaningful distinctions in student performance therefore in Figure 9 we can see all objects are valid.

Expected Outcome for Valid Objects: Direct-ranking vs. Inverse-ranking should produce inverted results with differences centered around the norm value 1000 (c.f. Figure-9):

Figure 9. Validation results (source: own presentation)

Important error: standard row-headers must always be given in case of all OAMs!

Example: (userid = O(1)-O(n) = student1-studentn, etc.)!!!

****

Reference: <https://miau.my-x.hu/miau/315/moodle/OAM_student_analysis_Moodle.xlsx>

After symmetry effect analysis we verified the reliability of the COCO system by giving wrong inconsistent too high or low inputs since it is the key tool for the evaluation (c.f. Figure-10-11-12-13):

Figure 10. Incorrect inputs: too high values(source: own presentation)



Between the figures, it is necessary to have explanations!

Important errors:

column-headers must always be visible/readable AND

standardized in case of ALL attributes in all figures! (Units for all attributes must always be given!

Row-headers must always be given!

Figure 11. outputs of Figure 10. (source: own presentations)

 We do never need names!

In Figure 11, the result is not consistent and does not deliver a clean output due to too high incorrect values being given as inputs.

Figure 12. Incorrect inputs: one person gets too high constant value (source: own presentation)

Standardized row-headers!!! (e.g. Student(1)-Student(n))



Figure 13. Outputs of Figure 12. (source: own presentation)



In Figure 13, all the results are identical due to incorrect inputs that one person gets too high a constant value.

In Figures 10, 11, 12, and 13, we can see that inconsistent low-quality data inputs lead to poor results from the COCO module. However, the potential of real-world/correct data for insightful output is evident in Figures 8 and 9.

**RESULTS (=summary) DISCUSSION (=weakest elements in the entire own logic?)**

In this study, we created OAM consisting of 24 objects and 29 attributes that are categorized into two categories Diligence, and Understanding. In the first run on the COCO Y0 engine, all the students got identical performance scores indicating the anti-discriminative logic of the engine, which ensures fairness by avoiding preset weight relations in Figure 5. However, this uniformity hid some attributes and differences in how well things performed. To get the hidden results, attribute exclusion was performed based on the stairs2 table (Figure 6), which identified attributes with minimal differentiation impact (e.g., those with s1 row values starting at 23) and removed the attributes that were previously dominant in the initial evaluation. After attribute exclusion, we ran the COCO Y0 engine a second time. This gave us a clearer and more detailed ranking of the students, which is demonstrated in (Figure 7). Figure 8 visualizes the performance distribution of students, highlighting the variability in student performance across the group. A symmetry effect analysis was conducted to validate these results. The original ranking was reversed (Reversed Rank = 24 - Original Rank + 1), and the COCO Y0 engine re-evaluated the inverted data. The product of the original and inverted delta values was calculated for each student, with all values satisfying the validity condition (product ≤ 0), as shown in Figure 9. This confirmed the consistency and reliability of the rankings. To further assess the COCO Y0 engine’s reliability sensitivity tests were tested. Incorrect inputs, such as excessively high values (Figures 10 and 11) or constant high values for one student (Figures 12 and 13), were given. In Figure 11, the output was erratic and inconsistent due to the exaggerated inputs, while in Figure 13, all scores converged to identical values, reflecting the model’s response to uniform distortion. These tests demonstrated that the engine appropriately handles poor-quality data, supporting confidence in its performance with accurate inputs. Once all validation processes are complete, we want to see how objective and subjective evaluations differ. To get the subjective evaluations we asked students to rate their peers’ performance and engagement on a scale of 0 to 9.  In Figure 14, the results of the pure subjective evaluation can be seen, and the correlation between Figure 14 and 7 is -0.25—a negative value that tells us these two approaches don’t align. This lack of correlation underscores how subjective perceptions can differ sharply from objective data-driven results (c.f. Figure-14):

Figure 14. Subjective evaluation (Result of the vote within the classroom) (source: own presentation)



Even though the model tries to be objective, it has some limitations we need to consider. The researchers had to choose which features to focus on and decide whether higher or lower values were better (represented as 0 or 1). This introduces some subjectivity. For instance, if the model values how many replies a student makes more than how quickly they reply, it might not give a fair picture of their performance. Also, the accuracy of the text analysis (NLP) depends on how clear and consistent the writing is in the Moodle discussions. If students use informal language or give incomplete answers, the results could be inaccurate. The sensitivity tests (Figures 10–13) show that if the data isn't good, the model's results will be unreliable. There are also ethical concerns about using AI to detect AI use. We need to be careful not to unfairly punish students who are using AI tools in acceptable ways. Clear rules are needed. Finally, the model was only tested on 24 students, which is a small group. This means we can't be sure if it will work well for all students in different educational settings. We need to test it on a larger and more diverse group to see if it's truly reliable. Despite these challenges, the model advances educational assessment by offering a data-driven, anti-discriminative tool that balances quantitative and qualitative dimensions.

**CONCLUSION (reactions concerning the discussed layers)**

This model offers a more equitable and precise approach to evaluating online learning, though further research across diverse settings and ethical considerations are essential for its widespread and responsible application.

Summary!!!

**REFERENCES**

Each URL should be listed here: e.g.

Author1, year1, title1, url1, apropo(journal,conference,etc.)1, page1, ...

Author2, year2, title2, url2, apropo(journal,conference,etc.)2, page2, ...

ABC-ordered

1. <https://www.statista.com/statistics/1130331/e-learning-market-size-segment-worldwide/> [↑](#endnote-ref-1)
2. <https://miau.my-x.hu/myx-free/index_e.php3?x=e01> [↑](#endnote-ref-2)