Mooc Course Recommendation System Model With Explainable AI (XAI) Using Content Based Filtering Method

1st Mugi Praseptiawan
Informatics Engineering
Institut Teknologi Sumatera (ITERA)
Bandar Lampung, Indonesia
mugi.praseptiawan@if.itera.ac.id

2nd M. Fikri Damar Muchtarom
Informatics Engineering
Institut Teknologi Sumatera (ITERA)
Bandar Lampung, Indonesia
*Corresponding author
mfikri.120140077@student.itera.ac.id

3rd Nabila Muthia Putri
Informatics Engineering
Institut Teknologi Sumatera (ITERA)
Bandar Lampung, Indonesia
nabila.120140023@student.itera.ac.id

4th Ahmad Naim Che Pee
Information and Communication Technology
Universiti Teknikal Malaysia Melaka
Melaka, Malaysia
naim@utem.edu.my

5th Mohd Hafiz Zakaria
Information and Communication Technology
Universiti Teknikal Malaysia Melaka
Melaka, Malaysia
hafiz@utem.edu.my

6th Meida Cahyo Untoro
Informatics Engineering
Institut Teknologi Sumatera (ITERA)
Bandar Lampung, Indonesia
cahyo.untoro@if.itera.ac.id

2024 11th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI) | 979-8-3503-5531-4/24/$31.00 ©2024 IEEE | DOI: 10.1109/EECSI63442.2024.10776491

***Abstract*— Massive Open Online Course (MOOC) is a type of online course that has been designed and can be accessed by all individuals via the internet. The problem that is often found in MOOCs is the lack of a recommendation system provided by the algorithm of the MOOC. This research is conducted to analyze a recommendation system that applies the Content Based Filtering approach in order to solve the problems that occur. The recommendation system analyzed will function as a media that provides recommendations to users based on their preferences. By utilizing content-based methods, the recommendations given are expected to be exactly what the user wants. The level of explainability of the recommendation system is further emphasized by XAI using ELI5. By getting a concise explanation when a recommendation is given, the system will gain more trust from users for providing an appropriate recommendation. The assessment of the accuracy of the recommendation system model is measured using MAE. By researching this recommendation system using XAI, it is hoped that it can help future systems to improve the quality of the course recommendation system.**

Keywords—MOOC, content-based filtering, XAI, recommender system

I. INTRODUCTION

Massive Open Online Course (MOOC) is a type of online course that has been designed to be accessible to all individuals via the internet. MOOC is a development in the field of online education, which has gained significant spotlight and popularity in the early 2010s[1]. MOOC has several courses that do not charge any fees to all users from all walks of life which makes MOOC has more enthusiasts compared to other online learning platforms. The problem obtained in this MOOC is the lack of a recommendation system provided by the algorithm of the MOOC [2].

This is because the effectiveness of a recommendation system owned by MOOC depends on the quality of the data, the sophistication of the algorithm, and also the match of recommendations with the preferences of each user. Recommendation systems have become very popular lately and are used in various web-based applications. The recommendation system is a system used to provide suggestions related to the needs of users of the application or website. The advice given by the recommendation system is also not fixated with one aspect. A recommendation system usually focuses on a particular topic to issue recommendations that have been tailored to provide useful and appropriate suggestions to its users [3].

Recommender systems generally have three modeling categories that can be used, including Content Based Filtering (CBF), Hybrid Recommender System, and Collaborative Filtering. The CBF method is also based on information retrieval. Another name for CBF is Cognitive Filtering. This method will provide suggestions related to items based on the user's profile or user's item profile. The customer's profile will be created when the user turns on the system. The system will collect all items from the user and will recommend the items after analyzing the features of the items and users [4]. All items that have been recommended are items that have similarities to all items.

items that have been bookmarked by the user and have similarities with items that have been browsed by the user before. In CBF, item description and user profile have high importance. The CBF method also relies on the choices of the user [5]. The item description and orientation of the user profile have a very important role in the CBF method. This method will provide recommendations based on similarities with previously explored items. CBF has the ability to provide a personalized recommendation based on item characteristics and user preferences. CBF utilizes item features and user profiles in providing recommendations, so CBF has the ability to provide recommendations without relying on historical user data. CBF can also provide an explanation for a recommendation by highlighting features and attributes that contribute to the recommendation [6].

A similarity of two vectors will be measured using cosine similarity. This matrix will work by measuring the cos of an angle between the two documents in the vector. The angle between vectors will determine whether the vectors will point in the same direction or different directions [7]. If the vectors point in the same direction, then it can be interpreted that the documents are similar or similar, the closer they are expressed on the axes, the more similar they are. Conversely, the further away the axes are, the lower the level of similarity. Each document used for comparison will be presented in space as a plot, and has cosine similarity to capture the orientation of these plots [8].

By using cosine similarity, determining the orientation of a document can be done sequentially to determine the document quantity. Furthermore, all algorithms will be evaluated using a structured framework or methodology that is often used to assess, analyze, and measure the performance, effectiveness, or quality of a system, process, service, or other subject of interest. Model evaluation is usually used to make decisions and improvements based on objective data and criteria [9]. Model evaluation is also an important process that can help someone to understand more about the development of models and systems that have been made in order to produce predictions that are expected beforehand. Mean Absolute Error (MAE) is a method used to measure the accuracy of a model. MAE is also a measure of the average error in a set of predictions. It is measured as an average absolute difference between the predicted and actual values and will be used to assess the effectiveness of the regression model [10]. This evaluation method is easy to understand and is often used to assess the accuracy of predictions. The value that will be generated from MAE will show the average absolute or absolute error of the true value. MAE will later calculate the evaluation of the value generated by the modeling before it is given to the recommendation system [11].

II. Methodology

The entire process carried out in this research shown in Fig. 1. The research work is to implement a MOOC recommendation system by using the Content Based Filtering method and using Cosine Similarity to calculate the similarity between documents and then evaluated using Mean Absolute Error.

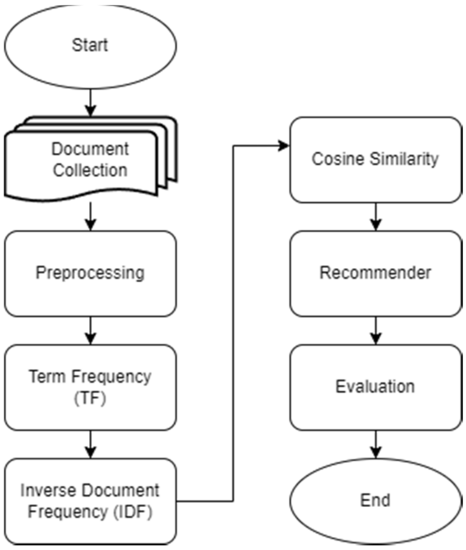

Fig. 1. Methodology flowchart

The model will perform several stages to reach the expected results, starting from collecting documents obtained from the Kaggle website with a dataset consisting of 3678 data, then the preprocessing will be carried out to sort the data to only use several attributes that will later be used in the model, after preprocessing is complete, the model will continue by calculating Term Frequency (TF) on the data that has been preprocessed, the purpose of this TF calculation is to give weighted values to each word contained in the dataset, If the TF calculation is complete, it will be continued with the calculation of Inverse Document Frequency (IDF) before continuing it to Cosine Similarity (CS), in the CS process all the data that has been given a weighted value by TF will be processed for the similarity values between words, the CS process occurred here is very important because the results will determine on how accurate the recommendations will be given by the model later, then after the model issues a recommendation, an evaluation will be carried out using Mean Absolute Error (MAE) to calculate how much error is contained in the model.

III. Results

The research results consist of Exploratory Data Analysis (EDA), MAE calculation results, TF-IDF - Cosine Similarity calculation and XAI explanation from the Udemy Courses dataset totaling 3678 course titles provided in the following link: https://www.kaggle.com/datasets/andrewmvd/udemy-courses . The whole process is carried out in stages to finally be issued by the recommendation system with the Content Based Filtering method.

The recommendation system experiment based on combined features will focus on several attributes contained in the dataset shown in Table I, namely *course_title, subject, level, num_subscribers, is_paid,* and *content_duration*.

- ***Course_title***

  This attribute will be used to measure the similarity of each course title contained in the dataset. For example, if the course title being viewed by the user contains the word "Python" then the recommendation system will provide the user with courses with titles that have other "Python" words.

- ***Subject***

  This attribute will be used to perform clustering of the titles contained in the dataset. There are four parameters in this indicator, namely Business Finance, Graphic Design, Musical Instruments, and Web Development.

- ***Level***

  In this attribute, there are four main parameters, namely Beginner Level, Intermediate Level, Expert Level, and All Levels.

- ***Num_subscribers***

  This attribute will be used in modeling to make it easier for the system to provide recommendations from the number of subscribers contained in each course.

- ***Is_paid***

  In this attribute there are two main parameters, TRUE and FALSE. TRUE means the course to be taken is paid and FALSE means the course is free.

- ***Content_duration***

  In this attribute there is data from the duration of the course that will be taken by the user. So the modeling will use the duration data to provide recommendations according to the duration that is often taken by users in choosing a course.

In this experiment, preprocessing will be carried out first and calculating TF - IDF and CS then the Recommendation

System based on Combined Features. The preprocessing in this experiment is combining the selected attributes namely *course_title, subject, level, num_subscribers, is_paid,* and *content_duration* into a new string and add it to a new column namely *combined_features* to the DataFrame

TABLE I. DATA ATTRIBUTES BEFORE PREPROCESSING

Before Preprocessing					
course_title	subject	level	num_subscribers	Is_paid	Content_duration
Python Programming for Beginners	Programming	Beginner	20000	True	6.5

TABLE II. DATA ATTRIBUTES AFTER PREPROCESSED

After Preprocessing
combined_features
Python Programming for Beginners Programming Beginner 20000 True 6.5

Table II, shows that the attributes are merged into a new attribute called combined_features. This process helps the model to give a better recommendation compared to using only one attribute at the time such as only the course_title or any other attributes.

After preprocessing the model will calculate the TF – IDF and the Cosine Similarity value, lastly the model will give an output for the course titles that have similarities with the titles inputted by the user based on the attributes shown in Fig. 2.

```
Enter a course title to search for similar courses (based on selected attributes):  Introduction to Web D
evelopment
Similar Courses based on Selected Attributes:
                                     course_title  similarity_score
2607              Introduction to Web Development          1.000000
3123    Beginners Introduction to Web Development          0.844288
2494        Introduction to Web Development: HTML          0.655557
2965              Introduction to CSS Development          0.600774
2670                             AJAX Development          0.514282
2686    WordPress Domination #2: Leads & Scarcity          0.508909
2753            Ruby On Rails For Web Development          0.504505
2599                 Introduction to Bootstrap 3           0.495893
3588  Bootstrap 4 Responsive Web Design and Development     0.483676
```

Fig. 2. Similarity score output for Combined Features

The output shows that the model provides 10 recommendations with the highest to lowest scores. Please note that the *similarity_score* has been calculated previously using TF - IDF and CS also based on the attributes that have been determined, and it can be seen that there is 1 title that has perfect similarity with the title "Introduction to Web Development", this is because the title "Introduction to Web Development" there are 2 in the dataset used, thus making the similarity score worth 1. And followed by the title "Beginners Introduction to Web Development" which has a *similarity_score* of 0.844288.

Model testing of each experiment will be carried out using the Mean Absolute Error (MAE) evaluation which serves to find the error value of a model. The value of MAE has an uncertain range, which is from 0 to infinity. The more the MAE value approaches 0, the better the model that has been assessed using MAE. MAE evaluation will be conducted for each recommendation system experiment, MAE will evaluate the similarity of course titles. To perform evaluation using MAE, a user interactions simulation must be created to simulate user interaction with the course. In this research, user interactions will randomly select a number of courses that interact with each user, then simulate the actual user behavior. After providing recommended courses to the user, the system will perform a comparison between the recommended course and the course that the user is currently interacting with. The model calculates the MAE value by comparing the binary indicator values, 1 for recommended courses, and 0 for non-recommended courses, for each course. If a recommended course is the same as a course that the user is currently taking, then the course will contribute 0 to the MAE because the user has already taken the course.

```
Mean Absolute Error (MAE) for combined features: 0.98066
```

Fig. 3. MAE output for Combined Features

And the output generated by MAE on modeling the course title-based recommendation system alone is 0.98 as shown in Fig. 3, which indicates that, on average, for each simulated user, the course recommendation provided differs from the course that the user is actually interacting with by about 0.98. After that, XAI integration will be done to determine the weight of a feature contained in the dataset used in making a prediction by showing the importance of each feature in the decision-making process using ELI5. This process is assisted by using *train_test_split* from the *sklearn.model_selection* module, *RandomForestRegressor* from the *sklearn.ensemble* module, *Pipeline* from the *sklearn.pipeline* module, and *eli5*.

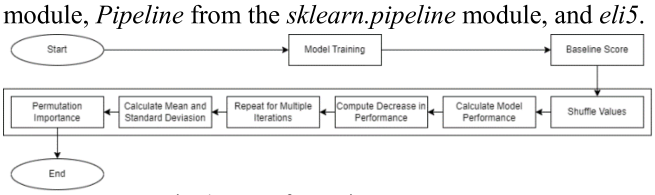

Fig. 4. ELI5 feature importance steps

The flowchart in Fig. 4 explains the process of XAI ELI5 to get the value of feature importance. The steps taken are:

- **Model Training**: Train a machine learning model on the dataset.
- **Baseline Score**: Calculate initial performance metrics of the model on the original dataset for each feature will be performed.
- **Shuffle Values**: Randomizes feature values while keeping other features unchanged.
- **Calculate Model Performance**: Train the model using the randomized dataset and calculate model performance metrics.
- **Compute Decrease in Performance**: Calculating the performance metric degradation caused by Shuffle Values.
- **Repeat for Multiple Iterations**: Repeating the process for several iterations to get the distribution of performance score reduction.
- **Calculate Mean and Standard Deviation**: Calculating the mean and standard deviation of the decrease in performance scores.
- **Permutation Importance**: Calculating permutation importance for features.

Weight	Feature
0.5264 ± 0.9914	development
0.4668 ± 0.9914	web
0.0024 ± 0.0080	instruments
0.0023 ± 0.0087	musical
0.0008 ± 0.0059	design
0.0005 ± 0.0035	graphic
0.0005 ± 0.0043	finance
0.0004 ± 0.0029	business

Fig. 5. ELI5 feature importance output

In the output that has been given in Fig. 5, it can be seen that the word "development" and followed by the word "web", the word "development" has the highest importance weight with a weight of 0.5264 and an uncertainty of 0.9914. Uncertainty in the XAI output represents the standard deviation associated with the weight of the feature. And for other features have a much smaller weight, indicating that these features have less importance in predicting the intended variable, namely the number of subscribers.

Result analysis is the last stage that must be done in a study. At this stage, a comparative analysis of the results of experiments that have been carried out on the recommendation system using the content-based filtering method will be carried out. The results obtained from the various experiments conducted are that each model has different *similarity_score* results in each experiment, based on the course inputted by the user, namely "Introduction to Web Development", the combined features experiment has the second largest recommendation result with a recommendation score of 0.844 which is greater than the course title experiment 0.811, and the course subject title 0.826. The difference in value in each experiment is due to the different attributes used in each experiment. For experiments that only use course title, the attribute used is only *course_title*, then for experiments that use course title and course subject, the attributes used are *course_title* and *subject*, and for combined features experiments, the attributes used are *course_title, subject, level, num_subscribers, is_paid,* and *content_duration*. The difference in the use of attributes is what causes the difference in the similarity_score value issued in each experiment. However, each experiment still has an error value or MAE which means that there are still errors that occur in each experiment conducted. This can be overcomed by several things, one of which is creating user interactions based on real user behavior, using historical data to get data containing actual user behavior, and others.

IV. Conclusion

In the research on Modeling MOOC Course Recommendation System with Explainable AI (XAI) Using Content Based Filtering Method, it can be concluded that based on experiments that have been carried out in several scenarios using a dataset of 3678 data, the similarity_score level of the combined features experiment is greater than the course title experiment, and the subject title. The difference in each experiment is caused by the use of attributes as a reference for the recommendation system experiments carried out, where in the combined features experiment, the attributes used include *course_title, subject, level, num_subscribers, is_paid,* and *content_duration*. Whereas in the course title and subject title process only uses *course_title*, and subject attributes. So that the value of the *similarity_score* issued by the combined features will be greater due to the wider use of attributes..

References

[1] Khalid, A., Lundqvist, K., & Yates, A. (2020). Recommender Systems for MOOCs: A Systematic Literature Survey (January 1, 2012 - July 12, 2019). International Review of Research in Open and Distance Learning, 21(4), 256–291. https://doi.org/10.19173/IRRODL.V21I4.4643J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[2] Assami, S., Daoudi, N., & Ajhoun, R. (2022). Implementation of a Machine Learning-Based MOOC Recommender System Using Learner Motivation Prediction. International Journal of Engineering Pedagogy, 12(5), 68–85. https://doi.org/10.3991/ijep.v12i5.30523

[3] Chalkiadakis, G., Ziogas, I., Koutsmanis, M., Streviniotis, E., Panagiotakis, C., & Papadakis, H. (2023). A Novel Hybrid Recommender System for the Tourism Domain. Algorithms, 16(4). https://doi.org/10.3390/a16040215

[4] Elektronika, J., & Komputer, D. (2023). Sistem Rekomendasi Dosen Pembimbing Tugas Akhir Menggunakan Content Based Filtering. 16(1), 64-71. https://journal.stekom.ac.id/index.php/elkom■page64.

[5] Salim, E., Pragantha, J., & Lauro, M. D. (n.d.). Perancangan Sistem Rekomendasi Film menggunakan metode Content-based Filtering.

[6] Jannach, D., Geyer, W., Freyne, J., Anand, S., Dugan, C., Mobasher, B., & Kobsa, A. (n.d.). Recommender Systems & the Social Web.M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

[7] G. Ferio, R. Intan, and S. Rostianingsih, "Sistem Rekomendasi Mata Kuliah Pilihan Menggunakan Metode User Based Collaborative Filtering Berbasis Algoritma Adjusted Cosine Similarity."

[8] T. Wahyuningsih, "Text Mining an Automatic Short Answer Grading (ASAG), Comparison of Three Methods of Cosine Similarity, Jaccard Similarity and Dice's Coefficient," *Journal of Applied Data Sciences*, vol. 2, no. 2, pp. 45–54, 2021.

[9] S. Al-Otaibi *et al.*, "Cosine similarity-based algorithm for social networking recommendation," *International Journal of Electrical and Computer Engineering*, vol. 12, no. 2, pp. 1881–1892, Apr. 2022, doi: 10.11591/ijece.v12i2.pp1881-1892.

[10] "Sistem Rekomendasi Sepatu Lokal Menggunakan Metode Collaborative Filtering Pada Toko Sepatu Tarsius Store".

[11] S. Devi Nurhayati and W. Widayani, "Sistem Rekomendasi Wisata Kuliner di Yogyakarta dengan Metode Item-Based Collaborative Filtering Yogyakarta Culinary Recommendation System with Item-Based Collaborative Filtering Method," 2021. [Online]. Available: https://manganenakyog.my.id/,