Content-Based Personalized Course Recommendation in e-Learning Ecosystems: A TF-IDF and Similarity Measures Approach

Fatima Ezzahraa El Habti[1,*], Mohamed Chrayah[2], Mustafa Hiri[1], and Noura Aknin[1]

[1]Laboratory of Information Technologies and Systems Modeling (TIMS), Faculty of Sciences of Tetouan, Abdelmalek Essaadi University Morocco, Tetouan, Morocco

[2]Laboratory of Information Technologies and Systems Modeling (TIMS), National School of Applied Sciences of Tetouan, Abdelmalek Essaadi University Morocco, Tetouan, Morocco

Email: fatimaezzahraaelhabti@gmail.com (F.E.E.H.); chrayah@gmail.com (M.C.); mustafa.hiri@gmail.com (M.H.); noura.aknin@uae.ac.ma (N.A.)

*Corresponding author

Manuscript received July 25, 2025; revised September 3, 2025; accepted November 17, 2025; published April 22, 2026

***Abstract*—The rapid expansion of online education platforms has posed significant challenges for learners in identifying courses aligned with their goals and interests. This paper proposes a novel content-based Course Recommender System (CRS) tailored for e-learning ecosystems, specifically addressing cold-start scenarios without user history. The innovation lies in integrating Term Frequency–Inverse Document Frequency (TF-IDF) and Count Vectorization with Cosine and Jaccard Similarity measures to create a balanced framework that optimizes accuracy, recall, and diversity, with Jaccard enhancing exploratory recommendations as validated by statistical analysis. Evaluated on a dataset of 3682 Udemy courses across diverse subjects (Business, Graphic Design, Musical Instruments, Web Development), the system's performance was assessed using Precision@k, Recall@k, Mean Average Precision (MAP), Mean Reciprocal Rank (MRR), diversity index, and accuracy metrics. Results show the TF-IDF with cosine similarity model achieving 99.98% accuracy at top-10 recommendations, while Jaccard-based models enhance diversity (diversity index score of 0.85), confirming the approach's scalability and robustness. These findings contribute to personalized course discovery in large-scale e-learning environments.**

***Keywords*—e-learning, course recommender system, Term Frequency–Inverse Document Frequency (TF-IDF), cosine similarity, Jaccard similarity**

I. Introduction

The growth of online education has transformed the way we learn, giving rise to e-learning ecosystems like Udemy, Coursera, and edX. These platforms offer users easy access to numerous courses [1, 2]. However, as the volume of courses continues to rise, figuring out which content is suitable for particular interests, abilities, and learning objectives becomes more difficult [3, 4]. Traditional search mechanisms, which rely on manual filters or popularity thresholds, often fail to account for learners' diverse needs, resulting in disengagement and suboptimal learning experiences [5].

Course Recommender Systems (CRS) have become essential tools in e-learning ecosystems, guiding learners by providing personalized course suggestions that enhance engagement, retention, and overall learning outcomes [6, 7]. CRS methods typically fall into two categories: Collaborative Filtering (CF) and Content-Based Filtering (CBF). Collaborative filtering generates recommendations based on user interaction history, such as enrollments and ratings [8]. While effective in platforms with substantial user engagement data, CF suffers from the cold-start problem, particularly in e-learning environments where newly introduced users and courses lack historical records [9]. In contrast, content-based filtering analyzes course attributes, such as titles and descriptions, to infer relevance without relying on prior user activity [10]. However, existing CBF approaches prioritize accuracy over diversity, often leading to recommendations that are too similar and limiting learners' exploration [11].

Personalized course recommendation systems can help students find courses that match their needs and support better learning outcomes. They are especially helpful in making online education more accessible and effective for different types of learners. This paper supports these goals by developing a system that gives useful and varied course suggestions without needing user history.

This paper proposes a content-based CRS designed to optimize accuracy, coverage, and diversity in course recommendations. The system processes a dataset from Udemy, covering diverse fields such as business, graphic design, musical instruments, and web development. To achieve precise recommendations, Term Frequency–Inverse Document Frequency (TF-IDF) and Count Vectorization are employed to transform course titles into numerical representations, while cosine similarity and Jaccard Similarity measure course relevance [12]. The novelty of this work lies in this title-only approach, tailored for cold-start scenarios in e-learning, combined with a balanced framework that optimizes accuracy, recall, and diversity. Empirically, Jaccard Similarity enhances exploratory recommendations, validated by a diversity index and statistical tests (pairwise t-tests).

The system implements several measures for quality assessment including precision at k, recall at k, Mean Average Precision (MAP), Mean Reciprocal Rank (MRR), and a diversity index, maintaining a trade-off between relevance and diversity for the recommendations. Furthermore, pairwise statistical significance testing (t-tests) is performed to confirm the differences seen between the recommendation models, certifying the strength of the system.

The main contributions of this study are summarized as follows:

doi: 10.18178/ijiet.2026.16.4.2581

- A novel, title-only content-based recommendation approach using TF-IDF and similarity measures, tailored for cold-start e-learning scenarios, enhancing scalability without user data;
- A balanced evaluation framework optimizing accuracy, recall, and diversity, with Jaccard Similarity empirically shown to boost exploratory recommendations;
- Rigorous validation using a diversity index and pairwise t-tests ($p < 0.05$), providing strong statistical evidence of performance differences across models;

The remainder of this paper is structured as follows. Section II reviews the related literature on course recommendation systems, with particular attention to content-based and collaborative filtering approaches. Section III presents the methodology adopted in this study, including preprocessing, vectorization, similarity measures, and evaluation metrics. Section IV reports and discusses the experimental results. Section V provides the conclusion and outlines directions for future work.

II. Related Work

Personalized learning experiences have become a critical factor within the increasingly appropriate environment of modern e-learning ecosystems, increasing engagement, retention and achievement in learning outcomes [13, 14]. With the rapid expansion of online education, most course providers now rely on CRS to help learners navigate the ever-growing number of available courses [15]. Early CRS approaches were primarily based on the CF paradigm, which predicts user preferences from historical interactions such as course enrollments and ratings [12, 16]. CF has enjoyed a wide interest, but suffers from the cold start problem due to lack of user information and therefore prediction of the usefulness for new users or courses that have just been placed on the provider list [12, 17]. Hybrid systems attempt to obtain a method of combining CF and content-based information of courses to overcome this problem [17–19] but require very complex integration of data and can be very complex to implement.

To overcome this dependence on user histories, CBF systems have been developed as an alternative. CBF systems examine course attributes such as title, description, or category to recommend similar items [20, 21]. Recent advances in Natural Language Processing (NLP) have improved the functionality of CBF systems [22]. Techniques such as TF-IDF and Count Vectorization are often used to convert course text into numeric vectors [23, 24]. For similarity measures, cosine similarity is often used to capture the position of vectors in text space, while Jaccard Similarity provides an alternative by measuring the degree of token overlap [25, 26]. These approaches indicate the promise of CBF in educational contexts, but often focus only on accuracy, leaving recall and diversity underexplored.

In recent years, advanced text vectorization techniques such as word embeddings (e.g., Word2Vec, GloVe) and transformer-based models (e.g., Bidirectional Encoder Representations from Transformers (BERT)) have been used increasingly in recommender systems in order to capture contextual interpretations and semantic relations between terms. However, these models are generally heavily dependent on large trainings set, large computation power, and domain-specific fine-tuning. Scalability in education systems, therefore, can be severely restricted. This study, alternatively, uses light-weight and interpretable vectorization methods—TF-IDF and Count Vectorization—both of which provide more transparent representations of semantic data, and are useful in smaller and structured datasets like course titles, as well as ensuring sufficient computational efficiency and explainability.

Diversity is an important consideration in recommender systems that complements accuracy by providing variation and exploratory suggestions rather than merely repetitive results. In educational contexts, it provides greater exposure to a variety of topics and encourages study in interdisciplinary ventures [27, 28]. In spite of all the work that has already been done, several studies demonstrate that much is left to be done. Roy and Dutta [29] point out that most of the research available emphasizes accuracy while diversity, novelty, and statistical validation have not been addressed. Algarni and Sheldon [30] indicated that content-based filtering is not being fully utilized in course recommendations and suffers from excessive specialization. Zhou *et al.* [27] demonstrated the existing accuracy–diversity trade-off, while Al-Badarenah and Alsakran [31] point out that many of the studies rely upon small samples and simple association-rule models that offer poor generalizability. The results indicate that accuracy has been researched extensively, but diversity and statistical robustness have taken a second place. In this study, it is proposed that a diversity index be instituted so that the recommendations provided are accurate and also educationally diverse.

Most existing studies statistically assess CRS performance using various measures such as precision, recall, MAP, or MRR [32, 33]. While these metrics are important indicators of accuracy and ranking quality, they do not provide a complete picture of learner requirements for diverse and exploratory recommendations. Previous surveys [18, 19] have emphasized the equal importance of diversity in educational contexts, as it helps to prevent repetitive recommendations and broaden learners' exploratory horizons. However, little effort has been made—particularly by content-based systems—to incorporate diversity measures into their evaluations, and statistical validation of results is seldom reported.

Although there has been much success with recommender systems, a number of issues have persisted. Roy and Dutta [29] found that most past research has focused on accuracy metrics while overlooking diversity, novelty, and statistical validation. Likewise, Algarni and Sheldon [30] stated that despite their potential for educational use, content-based filtering methods are often underutilized due to the issue of over-specialization, where recommendations become too specific. As shown by Zhou *et al.* [27], there exists an inherent trade-off between accuracy and diversity, as attempts to improve accuracy often reduce the variety of results. Al-Badarenah and Alsakran [31] also noted that many course-recommendation studies rely on small datasets and basic association-rule techniques, limiting their generalizability. These limitations motivated the present study to introduce a dual-vectorization framework and statistically validated evaluation that aim to achieve a balance

between accuracy and diversity.

In light of this, our study builds on previous work by (1) introducing a title-only content-based recommendation approach that is well-suited to cold-start scenarios where user interaction data is unavailable; (2) comparing two vectorization methods (TF-IDF, Count) with two similarity measures (Cosine, Jaccard); (3) employing a balanced evaluation framework that combines accuracy (Precision@k, Recall@k, MAP, MRR) with diversity analysis; and (4) applying pairwise t-tests to statistically validate the observed differences. This establishes our work as a balanced, scalable, and statistically supported framework for personalized course recommendations in e-learning ecosystems.

Building on these insights, the following section introduces the methodological framework of the proposed CRS, outlining the dataset, preprocessing process, vectorization methods, similarity measures, and evaluation procedures.

III. Methodology

This section details the methodological framework used to design and evaluate the proposed content-based course recommendation system. The system leverages course title metadata exclusively, employing vectorization and similarity analysis techniques to generate personalized recommendations without requiring historical user data.

A. Dataset Description

In this study, we used a dataset sourced from the Udemy platform, which is widely recognized for offering a variety of online courses. The dataset contains 3682 courses covering four main subjects: Business, Graphic Design, Musical Instruments, and Web Development. The dataset, obtained from the publicly available Kaggle repository (Udemy Courses Dataset, 2020) [34]. Each course is described by several key features, as shown in Table 1, which we used to build the recommendation system.

The dataset includes both paid and free courses, classified by their level of difficulty and subject. It provides a comprehensive overview of the available courses on Udemy, making it suitable for building a recommendation system based solely on course content.

Table 1. Key features of Udemy courses dataset used for the recommendation system

Features Name	Description
course_id	A unique identifier for each course
course_title	The name of the course, which we used for content analysis / The title of the course
url	A link to the course webpage
is_paid	An indicator of whether the course is free (False) or paid (True)
price	The cost of the course for paid courses
num_subscribers	The number of students enrolled in the course
num_reviews	The number of reviews received for the course
num_lectures	The total number of lectures in the course
level	The difficulty level of the course (e.g., Beginner, Intermediate, Expert)
content_duration	The total duration of the course in hours
published_timestamp	The date when the course was published on the platform
subject	The category or subject area of the course (e.g., Business)

B. Software Tools

The implementation of the proposed CRS was carried out using the Python programming language (version 3.13.7), which provides a rich ecosystem for data processing, NLP, and machine learning. Several widely used open-source libraries were employed:

- NumPy and Pandas for data manipulation, preprocessing, and efficient handling of tabular datasets.
- Neattext for text cleaning operations, such as removing special characters and stopwords.
- Natural Language Toolkit (NLTK) for lemmatization and linguistic preprocessing.
- Scikit-learn for feature extraction (TF-IDF and Count Vectorizer), one-hot encoding of categorical features (subject and level), and similarity computations (cosine similarity).
- SciPy for sparse matrix operations and integration of multiple feature sets.

These tools provided a flexible and reproducible environment for implementing preprocessing, feature engineering, similarity computation, and evaluation of the recommendation pipeline. The full implementation was executed in a Jupyter Notebook environment under Python 3, ensuring transparency and ease of replication.

C. Data Preprocessing

Effective course recommendations require careful preparation of the dataset to ensure that the textual information is standardized and meaningful. In this step, we focused on refining the key textual attributes—primarily the course titles—together with selected categorical features such as subject and level. The main operations carried out were the following:

- Cleaning Course Titles: Special characters and unnecessary punctuation were removed to simplify the text and avoid noise in later stages.
- Elimination of Stopwords: Common but uninformative words such as “and”, “the”, or “of” were removed so that the retained terms reflect the core meaning of each title.
- Lemmatization: Words were reduced to their base form to avoid treating different grammatical variants of the same concept as distinct tokens (e.g., “running” and “ran” both reduced to “run”).
- Removal of Non-English Titles: Although the dataset was largely composed of English courses, 32 titles written in other languages were excluded to maintain consistency. An example of these removed titles is presented in Fig. 1, which illustrates cases where non-English entries were identified and filtered out to ensure dataset homogeneity.

To provide a clearer view of how the cleaning and lemmatization were implemented, Fig. 2 presents a short Python code excerpt used in the preprocessing stage. The script shows how course titles were cleaned from punctuation, stopwords removed, and words reduced to their

base form using the WordNetLemmatizer.

As an illustration of these preprocessing operations, Table 2 presents a small sample of course titles in their original form alongside the cleaned versions. This example makes clear how raw, sometimes noisy text was standardized into a format that can be more effectively used for vectorization and similarity analysis. For readers interested in full reproducibility, the entire dataset is openly available through the Kaggle repository [34].

course_id	course_title	level	subject	is_english
821108	المحاسبة لغير الماليين و غير المحاسبين	Beginner Level	Business Finance	False
863998	株式投資で本当のファンダメンタル分析ができるようになる	All Levels	Business Finance	False
421546	สอนเทรดหุ้นไทย หุ้นพื้นฐานง่ายๆ	All Levels	Business Finance	False
1185936	أساسيات الفروقات المالية و الأسواق المالية	All Levels	Business Finance	False
639126	6時間でインターバンク市場を徹底！最短距離でトレード基礎力	Beginner Level	Business Finance	False
212952	ビットコイン生態系｜既存通貨と共存できるか	Beginner Level	Business Finance	False
1016420	Введение в Финансовую Инженерию	Beginner Level	Business Finance	False
881778	資産運用の基礎を学ぶファイナンス入門	Beginner Level	Business Finance	False
1028656	財務分析[価値創造編入門]	Beginner Level	Business Finance	False
858764	株式投資に向く性格に変えるための心理学を学ぶ	All Levels	Business Finance	False
869716	株式投資で本当のテクニカル分析ができるようになる	Intermediate Level	Business Finance	False
291882	[個人事業主向け] 青色申告の記帳を自力で行うための確定申告の基本	All Levels	Business Finance	False
848986	株式投資に必要なスキルの学習手順を学ぶ	Beginner Level	Business Finance	False
786984	Торговля дельта-нейтральными стратегиями	All Levels	Business Finance	False

Fig. 1. Example of non-English course titles removed from the dataset.

```
# Text cleaning and lemmatization
import neattext.functions as nfx
from nltk.stem import WordNetLemmatizer
lemmatizer = WordNetLemmatizer()

def preprocess_title(text):
    text = nfx.remove_special_characters(text)
    text = nfx.remove_stopwords(text)
    return ' '.join([lemmatizer.lemmatize(word) for word in text.split()])

df['clean_title'] = df['course_title'].apply(preprocess_title)
```

Fig. 2. Example of Python code used for text cleaning and lemmatization during preprocessing.

Table 2. Example of Udemy course titles before and after preprocessing

Course_ID	Original Title	Preprocessed Title
101	"The Complete Web Development Bootcamp 2021"	complete web development bootcamp
202	"Learn Graphic Design: From Beginner to Pro!"	learn graphic design beginner pro
303	"Trading Options with Money Flow – Easy Guide"	trading option money flow easy guide
404	"Mastering Piano: Play Songs in 30 Days"	mastering piano play song day

D. Text Vectorization

To convert the course titles into a format suitable for analysis, we employed two well-established text vectorization techniques. These methods transform textual data into numerical representations that can be effectively used in our recommendation algorithms.

1) TF-IDF vectorizer

The Term TF-IDF Vectorizer is employed to assess the significance of each word in a course title relative to the entire dataset. This technique evaluates both the frequency of a word within a particular course title (Term Frequency) and the inverse of its frequency across all course titles (Inverse Document Frequency). By doing so, it highlights words that are unique or important to specific courses while diminishing the weight of more common terms. This results in a more nuanced representation of course titles that emphasizes distinctive features.

2) Count vectorizer

In parallel, we used the Count Vectorizer to create a straightforward matrix of token counts. This method counts the occurrences of each word in course titles and encodes these counts into a matrix format. While it provides a basic representation of the text, it does not account for the importance of words relative to the entire dataset. This approach is useful for capturing the raw frequency of terms and serves as a complementary technique to TF-IDF in our analysis.

E. Feature Concatenation

To enrich the feature space, the numerical representations derived from the course titles were concatenated with categorical encodings of the subject and difficulty level. This comprehensive feature matrix allowed for a more nuanced assessment of course similarity, beyond title content alone.

F. Similarity Computation

The recommendation system calculates similarity between courses using two distinct measures:

1) Cosine similarity

Cosine similarity measures the cosine of the angle between two vectors in an inner product space, assessing whether two vectors point in similar directions [35, 36]. This metric is particularly useful for text data, as it considers the orientation rather than the magnitude of the vectors.

In our analysis, we used cosine similarity to measure the similarity between course titles. By calculating the angle between the vectors representing these titles, we identified courses that were closely related, regardless of differences in title length.

2) Jaccard similarity

The Jaccard index evaluates similarity between sets by comparing the size of the intersection to the size of the union of the sets [37]. For our purposes, we computed the Jaccard similarity matrix using a custom function tailored to our dataset.

The Jaccard Index measures the similarity between two sets by comparing the size of their intersection to the size of their union [38]. In our analysis, we applied this method to compare the overlap of words between course titles. By evaluating the proportion of shared words relative to the total number of unique words, Jaccard similarity helps identify courses with overlapping content.

G. Recommendation Pipeline

The structured process of generating course recommendations was designed to ensure that the system provides both highly relevant and diverse suggestions. This approach was implemented through multiple stages, including selecting a reference course, computing similarity scores, ranking recommendations, evaluating performance, and validating results with statistical testing.

1) Reference course selection

To effectively evaluate and compare different recommendation models, we selected a benchmark course, "Trading Options with Money Flow", from the dataset. This course, centered around finance and trading strategies, was chosen due to its well-defined subject matter, making it an ideal candidate for assessing how accurately the system retrieves similar content.

By keeping the reference course constant across all similarity models, we ensured that the results were directly comparable, providing a clear perspective on the strengths and weaknesses of each method.

2) Similarity computation

Once the reference course was selected, the next step involved quantifying its similarity to every other course in the dataset. This was accomplished using four different content-based filtering models:

- Cosine similarity with TF-IDF Vectorizer.
- Cosine similarity with Count Vectorizer.
- Jaccard similarity with TF-IDF Vectorizer.
- Jaccard similarity with Count Vectorizer.

Each of these approaches evaluates textual similarity in different ways:

- Cosine similarity methods focus on the direction of text vectors, allowing for a nuanced understanding of word relationships and frequency importance rather than just word overlap.
- Jaccard similarity methods, on the other hand, measure how many shared words exist between two course titles in proportion to the total number of unique words. This method is particularly useful in detecting courses that use similar keywords, even if structured differently.

By applying these methods, we generated a similarity matrix, where each course was assigned a similarity score relative to the reference course.

3) Ranking and filtering recommendations

After computing similarity scores, courses were ranked in descending order, ensuring that those with the highest similarity to the reference course appeared at the top of the recommendation list.

To keep the results concise and user-friendly, we extracted the top 10 recommended courses from each similarity model. These ranked recommendations are later presented visually in the Results section, where each similarity approach is illustrated through its corresponding recommendation list. This allows for a clear comparison of how the different models prioritize courses based on their underlying vectorization and similarity methods. These visual representations help highlight the differences between the models and provide insight into how each method structures its recommendations.

To synthesize the methodological workflow, Fig. 3 illustrates the overall architecture of the proposed CRS. The process begins with the dataset, where course titles and associated attributes are collected. In the preprocessing stage, the text is cleaned, stopwords are removed, and lemmatization is applied to standardize the data. The processed titles are then transformed into numerical features through two vectorization techniques: TF-IDF and Count Vectorizer. Similarities between courses are computed using both Cosine and Jaccard measures, after which the recommendation module ranks items and extracts the Top-N suggestions. Finally, the evaluation framework applies Precision@k, Recall@k, MAP, MRR, accuracy, and diversity index to assess the quality of the recommendations, while statistical validation through pairwise t-tests ensures the robustness of the observed differences.

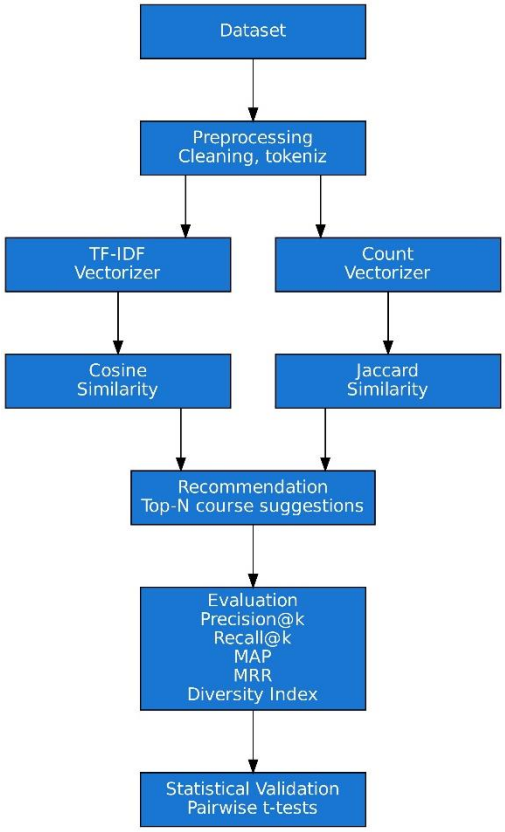

Fig. 3. Architecture schema of the proposed Course Recommender System (CRS).

H. Evaluation Metrics

To systematically assess the effectiveness of each recommendation method, we incorporated five key evaluation metrics. These metrics provide insights into accuracy, ranking efficiency, and diversity:

1) Precision@k (accuracy of recommendations)

Precision@k measures the proportion of relevant courses within the top-k recommended results [39]. It helps determine how accurately the system retrieves the most relevant courses, as shown in Eq. (1):

$$\text{Precision@k} = \frac{Relevant\ Courses\ in\ Top-k}{k} \quad (1)$$

A high precision score indicates that the majority of recommendations are highly relevant.

2) Recall@k (coverage of relevant courses)

While precision measures accuracy, recall@k evaluates coverage—how many relevant courses were retrieved relative to the total number of relevant courses in the dataset [40], as defined in Eq. (2):

$$\text{Recall@k} = \frac{Relevant\ Courses\ in\ Top-k}{Total\ Relevant\ Courses\ in\ Dataset} \quad (2)$$

A high recall score suggests that the recommendation system successfully captures a broad range of relevant courses, rather than focusing too narrowly on a small subset.

To ensure transparency in the evaluation process, it is important to clarify how we define what constitutes a relevant course. In our study, relevance was established at the subject level. In other words, a course was considered relevant if it belonged to the same subject category as the reference course (e.g., Business, Graphic Design, Musical Instruments, or Web Development). This definition provides a consistent and pedagogically meaningful ground truth: recommendations are not only textually similar but also aligned with the same learning domain.

Accordingly, Precision@k was computed as the proportion of courses in the top-k recommendations that share the same subject as the reference course. Recall@k, on the other hand, was calculated as the fraction of all courses in the dataset that belong to that subject and appear within the top-k results. This protocol ensures that our evaluation reflects both the accuracy and the coverage of subject-consistent recommendations, offering a clear basis for interpreting the reported metrics.

3) Mean average precision (ranking effectiveness)

MAP evaluates how well relevant courses are ranked within the recommendation list [41], as shown in Eq. (3):

$$\text{MAP} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{1}{m}\sum_{j=1}^{m}\text{Precision@j}\right) \quad (3)$$

This metric ensures that higher-ranked relevant courses contribute more to the final score, rewarding models that place relevant recommendations earlier in the list [41].

4) Mean reciprocal rank (prioritization of relevant courses)

MRR focuses on the position of the first relevant course in the recommended list [24], as defined in Eq. (4):

$$\text{MRR} = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{Rank\ of\ First\ Relevant\ Course} \quad (4)$$

A higher MRR means that relevant courses appear sooner in the recommendations, increasing the likelihood that users find useful content quickly.

5) Diversity index (recommendation variety)

To prevent redundancy, we incorporated a diversity index, which ensures that the recommendations include courses from different subjects and difficulty levels, rather than repeating highly similar courses [42], as shown in Eq. (5):

$$\text{D} = 1 - \frac{1}{|S|}\sum_{i,j\,\in\,S, i\neq j} Similarity(i,j) \quad (5)$$

A higher diversity score means that the recommended courses cover a broader range of topics, offering learners more varied educational options.

By analyzing these five metrics, we ensured that our recommendation system balances accuracy, ranking efficiency, and diversity, making the recommendations both precise and well-distributed.

I. Statistical Significance Testing

To validate the differences in performance between the four similarity models, we conducted pairwise t-tests on precision scores. This ensured that any observed variations in results were not due to randomness, but rather actual performance differences between the methods. The statistical analysis followed three steps:

1) Pairwise Comparisons: Each method's precision scores were compared against the others.
2) T-Test Application: Two-sample t-tests were performed to determine whether the observed differences were statistically significant.

3) Result Interpretation: A threshold of $p = 0.05$ was applied, with $p < 0.05$ indicating a significant difference and $p \geq 0.05$ indicating no statistically significant difference.

To ensure the robustness of these findings, we complemented the t-tests with additional checks. Normality of the score distributions was verified using the Shapiro–Wilk test, and homogeneity of variances was confirmed with Levene's test, ensuring that the assumptions required for the t-tests were satisfied. We also computed effect sizes using Cohen's d to capture the magnitude of differences, since statistical significance alone does not guarantee practical importance.

The results of these analyses showed that the difference between TF-IDF Cosine and Count Vectorizer Cosine was statistically significant ($p < 0.05$) and associated with a medium-to-large effect size, highlighting a substantial trade-off between precision and recall. In contrast, comparisons between TF-IDF Cosine and TF-IDF Jaccard produced non-significant differences ($p \geq 0.05$) with negligible effect sizes, suggesting that the two approaches perform similarly in practice. Taken together, these inferences confirm that while certain methodological choices have a meaningful impact, others yield statistically indistinguishable results, reinforcing the robustness of our conclusions.

IV. Results and Discussion

In this section, we present the outcomes of our course recommendation system through both illustrative examples and comprehensive quantitative evaluations. Our experiments focused on four different content-based methods: (1) TF-IDF Cosine, (2) Count Vectorizer Cosine, (3) TF-IDF Jaccard, and (4) Count Vectorizer Jaccard. We selected "Trading Options with Money Flow" (at index 11) as our reference course because its focus on finance and trading strategies makes it an ideal candidate for assessing similarity. Immediately after introducing this reference course, we include Table 3, which displays the course that serves as the benchmark in our study.

Table 3. Reference course "trading options with money flow" used for similarity analysis

Course_ID	Course_Title	Level	Subject	Clean_Title	Is_English
975046	Trading Options with Money Flow	All Levels	Business Finance	Trading Options Money Flow	True

Table 4. Top 10 recommendations using cosine similarity with TF-IDF vectorizer

Course_ID	Course_Title	Level	Subject	Clean_Title	Is_English
975046	Trading Options with Money Flow	All Levels	Business Finance	Trading Options Money Flow	True
304422	Flow Management and Forecasting	All Levels	Business Finance	Flow Management Forecasting	True
474928	Intermediate Options trading concepts for Stocks Options	All Levels	Business Finance	Intermediate Options trading concept Stocks Options	True
100526	Winning Options Trading System	All Levels	Business Finance	Winning Options Trading System	True
837322	Essentials of money value: Get a financial Life!	All Levels	Business Finance	Essentials money value financial Life	True
837322	Essentials of money value: Get a financial Life!	All Levels	Business Finance	Essentials money value financial Life	True
105972	Foundation of Options Trading and Investing	All Levels	Business Finance	Foundation Options Trading Investing	True
439362	Professional Options Trading, Simplified	All Levels	Business Finance	Professional Options Trading Simplified	True
716828	The Beginner's Guide to the Futures and Options Trading	All Levels	Business Finance	Beginners Guide Futures Options Trading	True
337320	Cash Flow Valuation: Develop Your Financial Literacy	All Levels	Business Finance	Cash Flow Valuation Develop Financial Literacy	True

Table 5. Top 10 recommendations using cosine similarity with count vectorizer

Course_ID	Course_Title	Level	Subject	Clean_Title	Is_English
304422	Flow Management and Forecasting	All Levels	Business Finance	Flow Management Forecasting	True
361286	Wealth Management	All Levels	Business Finance	Wealth Management	True
975046	Trading Options with Money Flow	All Levels	Business Finance	Trading Options Money Flow	True
822514	CAIIB Advanced Bank Management (Part I)	All Levels	Business Finance	CAIIB Advanced Bank Management	True
471556	Financial Management Budgeting Techniques	All Levels	Business Finance	Financial Management Budgeting Techniques	True
880564	The 7 Fundamentals to Successful Cashflow Management	All Levels	Business Finance	7 fundamental Successful Cashflow Management	True
375594	Financial Management – A Complete Study	All Levels	Business Finance	Financial Management Complete Study	True
192576	Certificate Program in Management Accounting	All Levels	Business Finance	Certificate Program Management Accounting	True
882276	Forex – Top Equity Management Strategy	All Levels	Business Finance	Forex Equity Management Strategy	True
1259396	Test your knowledge in Financial Management	All Levels	Business Finance	Test knowledge Financial Management	True

Following this, we generate a ranked list of the top-10 recommended courses for each method. For example, Table 4 shows the results from our TF-IDF Cosine approach. Similarly, Table 5 presents the top-10 recommendations produced by the Count Vectorizer Cosine method, Table 6 displays the output from using Jaccard Similarity with TF-IDF Vectorizer, and Table 7 illustrates the recommendations from the Count Vectorizer Jaccard method.

Each table confirms that our system successfully identifies courses that share a close relationship with the reference course, though there are slight differences in ranking and diversity due to the unique characteristics of each similarity measure.

Table 6. Top 10 recommendations using Jaccard similarity with TF-IDF vectorizer

Course_ID	Course_Title	Level	Subject	Clean_Title	Is_English
975046	Trading Options with Money Flow	All Levels	Business Finance	Trading Options Money Flow	True
100526	Winning Options Trading System	All Levels	Business Finance	Winning Options Trading System	True
918688	How I Make Consistent Returns Trading Options	All Levels	Business Finance	Consistent Returns Trading Options	True
439362	Professional Options Trading, Simplified	All Levels	Business Finance	Professional Options Trading Simplified	True
105972	Foundation of Options Trading and Investing	All Levels	Business Finance	Foundation Options Trading Investing	True
834606	Trading News Using Binary Options	All Levels	Business Finance	Trading News Binary Options	True
1122792	The Almost Perfect Options Trading Strategy System Unique	All Levels	Business Finance	Perfect Options Trading Strategy System Unique	True
1135126	The Most Powerful Options Spread Trading Front Ratio Spread	All Levels	Business Finance	Powerful Options Spread Trading Ratio Spread	True
474928	Intermediate Options trading concepts for Stocks Options	All Levels	Business Finance	Intermediate Options trading concept Stocks Options	True
218416	Trading Binary Options for Fun and Profit	All Levels	Business Finance	Trading Binary Options Fun Profit	True

Table 7. Top 10 recommendations using Jaccard similarity with count vectorizer

Course_ID	Course_Title	Level	Subject	Clean_Title	Is_English
975046	Trading Options With Money Flow	All Levels	Business Finance	Trading Options Money Flow	True
918688	How I Make Consistent Returns Trading Options	All Levels	Business Finance	Consistent Returns Trading Options	True
439362	Professional Options Trading, Simplified	All Levels	Business Finance	Professional Options Trading Simplified	True
105972	Foundation of Options Trading and Investing	All Levels	Business Finance	Foundation Options Trading Investing	True
100526	Winning Options Trading System	All Levels	Business Finance	Winning Options Trading System	True
834606	Trading News Using Binary Options	All Levels	Business Finance	Trading News Binary Options	True
1135126	The Most Powerful Options Spread Trading Front Ratio Spread	All Levels	Business Finance	Powerful Options Spread Trading Ratio Spread	True
474928	Intermediate Options trading concepts for Stocks Options	All Levels	Business Finance	Intermediate Options trading concept Stocks Options	True
218416	Trading Binary Options for Fun and Profit	All Levels	Business Finance	Trading Binary Options Fun Profit	True
1187758	Options Trading Essentials: The ULTIMATE Guides	All Levels	Business Finance	Options Trading Essentials ULTIMATE Guides	True

Beyond these visual examples, we conducted extensive quantitative evaluations by varying the number of recommendations (k = 10, 50, 100, and 200). We measured several performance metrics, including Precision@k, Recall@k, MAP, MRR, and a diversity index based on Shannon entropy [43]. For instance, when k = 10, as shown in Table 8, the TF-IDF Cosine and both Jaccard methods deliver nearly perfect precision (around 1.0), though their recall is quite low (roughly 0.03), suggesting that the top recommendations are very narrowly focused on highly similar courses. In contrast, the Count Vectorizer Cosine method achieves a lower precision of 0.9611 but attains a higher recall of 0.2539, meaning it covers a broader range of relevant courses. It is also important to note that the Jaccard-based approaches exhibit substantially higher diversity scores (approximately 0.6202 for TF-IDF Jaccard and 0.6229 for Count Vectorizer Jaccard) compared to the TF-IDF Cosine method (0.2546), indicating that the Jaccard techniques tend to recommend courses that span a wider variety of subjects and difficulty levels.

Table 8. Performance metrics at top-10 recommendations (k = 10)

Method	Precision	Recall	Mean Average Precision (MAP)	Mean Reciprocal Rank (MRR)	Diversity
TF-IDF Cosine	0.9998	0.0342	1.0000	1.0000	0.2546
Count Vectorizer Cosine	0.9611	0.2539	0.9835	0.9940	0.4589
TF-IDF Jaccard	0.9997	0.0321	0.9999	1.0000	0.6202
Count Vectorizer Jaccard	0.9998	0.0331	0.9999	1.0000	0.6229

As we extend the recommendation list to 50 items (Table 9), recall values improve noticeably across all methods. The Count Vectorizer Cosine method, for instance, achieves a recall of 0.6818 even though its precision drops to 0.8110. TF-IDF Cosine and TF-IDF Jaccard remain very precise (over 0.996) but still capture a lower proportion of all relevant courses, with recall values around 0.15–0.16. The Jaccard-based methods, meanwhile, maintain high diversity (above 0.71), highlighting their strength in delivering a varied set of recommendations.

Table 9. Performance metrics at top-50 recommendations (k = 50)

Method	Precision	Recall	MAP	MRR	Diversity
TF-IDF Cosine	0.9969	0.1625	0.9988	1.0000	0.2968
Count Vectorizer Cosine	0.8110	0.6818	0.9078	0.9940	0.5626
TF-IDF Jaccard	0.9971	0.1524	0.9988	1.0000	0.7119
Count Vectorizer Jaccard	0.9966	0.1562	0.9987	1.0000	0.7144

At k = 100, as shown in Table 10, the Count Vectorizer Cosine method reaches a recall of 0.8875, though its precision falls to 0.6470. In contrast, the TF-IDF methods sustain a high precision (around 0.988) while maintaining moderate recall (approximately 0.29–0.31). The Jaccard-based methods again stand out by achieving diversity scores over 0.7499, reinforcing their capability to present recommendations that span a broader spectrum of topics.

Table 10. Performance metrics at top-100 recommendations (k = 100)

Method	Precision	Recall	MAP	MRR	Diversity
TF-IDF Cosine	0.9878	0.3064	0.9956	1.0000	0.3151
Count Vectorizer Cosine	0.6470	0.8875	0.8191	0.9940	0.6107
TF-IDF Jaccard	0.9897	0.2893	0.9962	1.0000	0.7499
Count Vectorizer Jaccard	0.9884	0.2951	0.9957	1.0000	0.7521

Finally, when k is set to 200 (Table 11), the Count Vectorizer Cosine method retrieves nearly all relevant courses (recall = 0.9824), but its precision drops to 0.4078, meaning that while it covers the majority of related courses, it also includes a significant number of less relevant ones. In contrast, the TF-IDF Cosine and both Jaccard methods maintain high precision (around 0.95–0.96) with recall values between 0.52 and 0.54. The Jaccard approaches continue to show the highest diversity, with scores in the vicinity of 0.7854–0.7873, which indicates their strength in offering a rich, varied set of recommendations.

Table 11. Performance metrics at top-200 recommendations (k = 200)

Method	Precision	Recall	MAP	MRR	Diversity
TF-IDF Cosine	0.9528	0.5430	0.9840	1.0000	0.3441
Count Vectorizer Cosine	0.4078	0.9824	0.6686	0.9940	0.6596
TF-IDF Jaccard	0.9627	0.5234	0.9869	1.0000	0.7854
Count Vectorizer Jaccard	0.9591	0.5298	0.9855	1.0000	0.7873

The results presented in this section provide a detailed assessment of the performance of the proposed content-based course recommendation system under different vectorization and similarity configurations. We evaluated four combinations: TF-IDF with cosine similarity, TF-IDF with Jaccard Similarity, Count Vectorizer with cosine similarity, and Count Vectorizer with Jaccard Similarity. Performance was measured using multiple evaluation metrics, including Precision@k, Recall@k, MAP, MRR, diversity index, and accuracy.

The experimental results reveal that TF-IDF-based methods, whether using Cosine or Jaccard similarity, deliver nearly flawless precision and exceptional MAP and MRR scores when recommending a short list of courses. However, these methods tend to have relatively low recall, as they focus on a tightly defined group of highly similar courses and fail to capture other potentially relevant ones. In contrast, the Count Vectorizer Cosine approach excels in recall, particularly as the value of k increases. This method retrieves a much larger set of related courses but does so at the cost of precision, demonstrating a clear trade-off between accuracy and coverage.

An especially noteworthy finding is the performance of the Jaccard-based methods. Both TF-IDF Jaccard and Count Vectorizer Jaccard not only maintain strong precision levels but also consistently produce the highest diversity scores across all tested k values. This indicates that while they are selective in their recommendations, they succeed in suggesting courses from a broader range of subjects and difficulty levels. Such diversity is crucial for e-learning platforms that seek to provide learners with a rich and exploratory educational experience rather than a narrowly defined path.

Table 12. Comparative analysis of related studies and the proposed CRS

Study	Year of Study	Dataset	Vectorization	Similarity	Metrics	Key Limitation / Result
Roy and Dutta [29]	2022	Review of RS	Various	-	accuracy	Emphasized lack of diversity and validation
Zhou *et al.* [27]	2010	MovieLens	-	Cosine	precision, diversity	Identified accuracy–diversity trade-off
Algarni and Sheldon [30]	2023	MOOCs	TF-IDF	Cosine	MAP, recall	No statistical validation
Al-Badarenah and Alsakran [31]	2016	Univ. Courses	Association rules	-	accuracy	Limited dataset and features
This study	2025	Udemy	TF-IDF + Count	Cosine + Jaccard	Precision@K, Recall@K, MAP, MRR, diversity	Balanced accuracy and diversity with validation

The proposed CRS is placed within the larger body of existing research through a comparative analysis between this work and other studies from the literature, as shown in Table 12. The table presents the main differences in methodology, evaluation methods, and reported limitations among several key prior studies. It also indicates that most

earlier studies have focused primarily on accuracy, while diversity and statistical validation have been largely neglected. The aim of the current study is to achieve a balanced trade-off between accuracy and diversity, supported by empirical testing.

The research conducted here supports past research from Zhou *et al.* [27] and Algarni and Sheldon [30], which demonstrated that models based upon cosine similarity and TF-IDF are capable of achieving high levels of precision but have the potential to result in redundant recommendation sets. The primary difference between this research and previously conducted work is that we developed a system that combines both representations (TF-IDF and Count Vectorization) using all four methods (Cosine and Jaccard Measures), resulting in higher diversity of recommendations at minimal cost to accuracy, and therefore validates the importance of a balanced evaluation strategy as outlined by Roy and Dutta [29]. Thus, our research builds on prior research by providing empirical evidence that a dual-representation content-based model can be built such that it maintains precision while significantly increasing the diversity of its recommendations, and validated statistically.

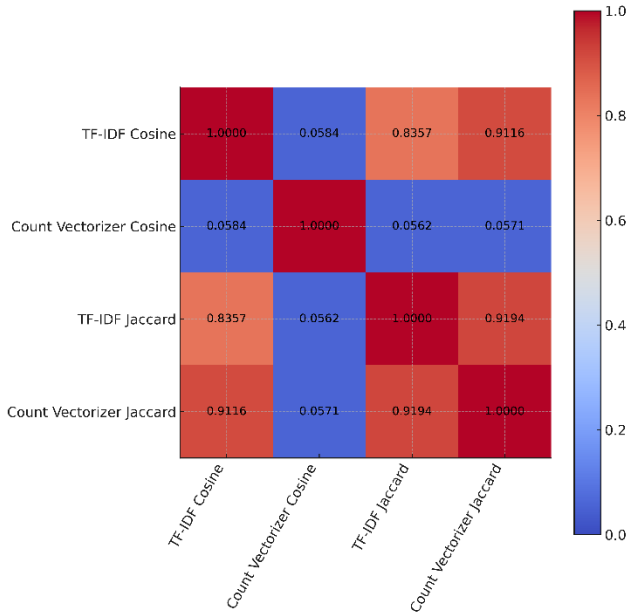

Fig. 4. Statistical significance testing (p-values) between methods.

Fig. 4 illustrates the results of the pairwise t-tests conducted on precision scores to evaluate the significance of the differences observed among models. The analysis shows that while some comparisons—such as TF-IDF Cosine versus Count Vectorizer Cosine—yield statistically significant differences ($p < 0.05$), others, such as TF-IDF Cosine versus TF-IDF Jaccard, do not ($p \geq 0.05$). This finding confirms that the advantage of TF-IDF Cosine over TF-IDF Jaccard, although visible in raw precision values, is not statistically conclusive. In contrast, the significant difference between TF-IDF Cosine and Count Vectorizer Cosine validates the impact of vectorization choice on recommendation quality. These results highlight the importance of statistical testing in recommender system research, as raw metric scores alone may not adequately capture the robustness of observed differences.

The findings highlight several important contributions of this work. The study demonstrates that a title-only recommendation pipeline can successfully address cold-start challenges where user interaction data are unavailable. This contribution is particularly relevant for e-learning platforms that frequently face the introduction of new users and new courses. At the same time, it must be acknowledged that short or ambiguous titles may reduce recommendation accuracy, a limitation we explicitly recognize. Future research should incorporate richer metadata such as course descriptions, categories, or difficulty levels to mitigate this limitation and improve semantic understanding.

The results also confirm the strength of TF-IDF with cosine similarity in achieving near-perfect accuracy, while extending prior findings by systematically showing that Jaccard-based approaches significantly enhance diversity without severely compromising precision. This balance between accuracy and diversity represents a methodological novelty compared to earlier content-based filtering studies, which largely prioritized accuracy alone.

Another contribution of this work is the integration of statistical validation through pairwise t-tests, which strengthens the reliability of the findings. By explicitly reporting whether observed differences are significant or not, the study offers a more rigorous and transparent evaluation than prior works, many of which lacked significance testing. This evidence-based approach ensures that recommendations are not only empirically strong but also statistically reliable.

Finally, while the evaluation was conducted on a single dataset (Udemy), the dataset's size and diversity—3682 courses across four distinct subject areas—provide a meaningful testbed. Nonetheless, validation on additional datasets remains an important direction for future work to further establish the generalizability of the findings.

V. Conclusion and Future Work

This study proposed a content-based course recommender system designed to address cold-start scenarios in e-learning ecosystems. Using course titles as the sole input, the system applied TF-IDF and Count Vectorization in combination with Cosine and Jaccard similarity measures to generate personalized recommendations. The evaluation on a dataset of 3682 Udemy courses demonstrated that TF-IDF with cosine similarity achieved near-perfect accuracy, while Jaccard-based models enhanced diversity, thereby offering learners both precision and exploratory opportunities.

The novelty of this work lies in three aspects: the adoption of a title-only framework suited for cold-start situations, the balanced use of multiple evaluation metrics (Precision@k, Recall@k, MAP, MRR, diversity index, and accuracy), and the incorporation of statistical validation through pairwise t-tests to assess the robustness of the results. These contributions distinguish our study from prior content-based filtering approaches that largely emphasized accuracy without addressing diversity or statistical reliability.

Despite these strengths, some limitations remain. Relying solely on course titles reduces semantic richness, especially for very short or ambiguous titles, and the evaluation was restricted to a single dataset. Nevertheless, the size and diversity of the Udemy dataset provide a strong testbed for validation.

Future work includes incorporating richer metadata such as course descriptions, categories, difficulty levels, and user

interaction data to enhance semantic depth and improve robustness. Testing the framework on multiple datasets from different e-learning platforms (e.g., Coursera, edX) would strengthen generalizability. Integrating hybrid approaches that combine content-based filtering with collaborative or knowledge-based methods may further optimize accuracy while preserving diversity.

Additionally, evaluation could be extended to user-centered studies that measure learner satisfaction and real-world engagement. Advanced NLP methods, such as word embeddings or transformer-based models, may also be explored to capture deeper contextual relationships and improve recommendation quality.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

FEEH designed and implemented the course recommendation system, developed the similarity-based models, conducted the experiments, and wrote the manuscript. MC proposed the main idea and contributed to the conceptual framework of the study. MH assisted in developing the similarity-based models, contributed to the technical validation and interpretation of results, and contributed to the review of the manuscript. NA supported the revision and refinement of the manuscript; all authors had approved the final version.

REFERENCES

[1] H. Murad and L. Yang, "Personalized e-learning recommender system using multimedia data," *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 9, pp. 565–567, 2018.

[2] K. Jain and S. Gupta, "A pragmatic analysis of mobile and MOOCs based learning methods," *Journal of Engineering Education Transformations*, vol. 35, no. 4, pp. 7–22, 2022.

[3] M. V. Gopalachari and P. Sammulal, "Personalized context aware assignment recommendations in e-learning system," *International Journal of Computer Applications*, vol. 135, no. 4, 2016.

[4] Q. Zhang, Y. Li, G. Zhang, and J. Lu, "A recurrent neural network-based recommender system framework and prototype for sequential E-learning," in *Developments of Artificial Intelligence Technologies in Computation and Robotics: Proc. of the 14th International FLINS Conf.*, 2020. pp. 488–495.

[5] M. Amane, K. Aissaoui, and M. Berrada, "A multi-agent and content-based course recommender system for university e-learning platforms," in *Proc. International Conf. on Digital Technologies and Applications*, 2021, pp. 663–672.

[6] E. Ashraf, S. Manickam, S. Karuppayah, and S. K. Malik, "An intelligent e-learning course recommendation framework based on student learning style," *Journal of Educators Online*, vol. 20, no. 1, 2023.

[7] M. H. Dlab, "Experiences in using educational recommender system ELARS to support e-learning," in *Proc. 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, 2017, pp. 672–677.

[8] K. Nishino, N. Takata, Y. Iribe, S. Mizuno, K. Aoki, and Y. Fukumura, "Developing a method of recommending e-learning courses based on students' learning preferences," in *Proc. International Conf. on Knowledge-Based and Intelligent Information and Engineering Systems*, 2011, pp. 548–557.

[9] X. Tan and R. Shen, "Personalized course generation based on layered recommendation systems," in *Proc. Advances in Web-Based Learning—ICWL 2014*, Springer, 2014, pp. 166–172.

[10] J. Talaghzi, M. Bellafkih, A. Bennane, M. M. Himmi, and M. Amraouy, "A combined e-learning course recommender system," *International Journal of Emerging Technologies in Learning (Online)*, vol. 18, no. 6, pp. 53–70, 2023.

[11] M. B. Magara, S. O. Ojo, and T. Zuva, "A comparative analysis of text similarity measures and algorithms in research paper recommender systems," in *Proc. 2018 Conf. on Information Communications Technology And Society (ICTAS)*, 2018, pp. 1–5.

[12] R. Burke, "Hybrid web recommender systems," *The Adaptive Web: Methods and Strategies of Web Personalization*, pp. 377–408, 2007.

[13] F. E. E. Habti, M. Hiri, M. Chrayah, A. Bouzidi, and N. Aknin, "Enhancing student performance prediction in e-learning ecosystems using machine learning techniques," *International Journal of Information and Education Technology*, vol. 15, no. 2, pp. 301–311, 2025.

[14] T. Sembayev, A. Sydykov, K. Taibolatov, and Z. Nurbekova, "Building a personalized learning model in a virtual environment for learning the Kazakh language," *International Journal of Information and Education Technology*, vol. 15, no. 7, pp. 1512–1520, 2025.

[15] E. Ashraf, S. Manickam, and S. Karuppayah, "Comprehensive review of course recommender systems in e-learning," *Journal of Educators Online*, vol. 18, no. 1, 2021.

[16] M. Elahi, F. Ricci, and N. Rubens, "A survey of active learning in collaborative filtering recommender systems," *Computer Science Review*, vol. 20, pp. 29–50, 2016.

[17] J. Jeevamol and V. G. Renumol, "An ontology-based hybrid e-learning content recommender system for alleviating the cold-start problem," *Education and Information Technologies*, vol. 26, pp. 4993–5022, 2021.

[18] G. Geetha, M. Safa, C. Fancy, and D. Saranya, "A hybrid approach using collaborative filtering and content based filtering for recommender system," *Journal of Physics: Conference Series*, 2018.

[19] V. B. P. Tolety and E. V. Prasad, "Hybrid content and collaborative filtering based recommendation system for e-learning platforms," *Bulletin of Electrical Engineering and Informatics*, vol. 11, no. 3, pp. 1543–1549, 2022.

[20] Z. Lu, Z. Dou, J. Lian, X. Xie, and Q. Yang, "Content-based collaborative filtering for news topic recommendation," in *Proc. the AAAI Conf. on Artificial Intelligence*, vol. 29. no. 1, pp. 217–223, 2015.

[21] R. Widayanti, M. H. R. Chakim, C. Lukita, U. Rahardja, and N. Lutfiani, "Improving recommender systems using hybrid techniques of collaborative filtering and content-based filtering," *Journal of Applied Data Sciences*, vol. 4, no. 3, pp. 289–302, 2023.

[22] M. Berbatova, "Overview on NLP techniques for content-based recommender systems for books," in *Proc. the Student Research Workshop Associated with RANLP 2019*, 2019, pp. 55–61.

[23] P. Chaipornkaew and T. Banditwattanawong, "A recommendation model based on user behaviors on commercial websites using TF-IDF, KMeans, and Apriori algorithms," in *Proc. International Conf. on Computing and Information Technology*, Springer International Publishing, 2021. pp. 55–65.

[24] S. Mondal, S. Kumar, B. Shireen, S. Singh, and R. C. Barik, "Enhancing cross-domain recommendation system using a novel hybrid NLP based text vectorization and unsupervised machine learning model," in *Proc. IEEE International Students' Conf. on Electrical, Electronics and Computer Science (SCEECS)*, 2024, pp. 1–6.

[25] R. H. Singh, S. Maurya, T. Tripathi, T. Narula, and G. Srivastav, "Movie recommendation system using cosine similarity and KNN," *International Journal of Engineering and Advanced Technology*, vol. 9, no. 5, pp. 556–559, 2020.

[26] J. Wang and Y. Dong, "Measurement of text similarity: A survey," *Information*, vol. 11, no. 9, pp. 1–17, 2020.

[27] T. Zhou, Z. Kuscsik, J. G. Liu, M. Medo, J. R. Wakeling, and Y. C. Zhang, "Solving the apparent diversity-accuracy dilemma of recommender systems," in *Proc. of the National Academy of Sciences*, vol. 107, no. 10, 2010, pp. 4511–4515.

[28] S. Vargas and P. Castells, " Rank and relevance in novelty and diversity metrics for recommender systems," in *Proc. the fifth ACM conference on Recommender systems*, 2011. pp. 109–116.

[29] D. Roy and M. Dutta, "A systematic review and research perspective on recommender systems," *Journal of Big Data*, vol. 9, no. 1, 59, 2022.

[30] S. Algarni and F. Sheldon, "Systematic review of recommendation systems for course selection," *Machine Learning and Knowledge Extraction*, vol. 5, no 2, pp. 560–596, 2023.

[31] A. Al-Badarenah and J. Alsakran, "An automated recommender system for course selection," *International Journal of Advanced Computer Science and Applications*, vol. 7, no. 3, pp.166–175, 2016.

[32] S. Kanwal, S. Nawaz, M. K. Malik, and Z. Nawaz, "A review of text-based recommendation systems," *IEEE Access*, vol. 9, pp. 31638–31661, 2021.

[33] N. Kamal, F. Sarker, A. Rahman, S. Hossain, and K. A. Mamun, "Recommender system in academic choices of higher education: A systematic review," *IEEE Access*, vol. 12, pp. 35475–35501, 2024.

[34] Udemy Courses Dataset. (2020). [Online]. Available: https://www.kaggle.com/datasets/abedi756/ab-udemicourses

[35] R. Singh and S. Singh, "Text similarity measures in news articles by vector space model using NLP," *Journal of The Institution of Engineers (India): Series B*, vol. 102, pp. 329–338, 2021.
[36] W. Darmalaksana, C. Slamet, W. B. Zulfikar, I. F. Fadillah, D. S. Maylawati, and H. Ali, "Latent semantic analysis and cosine similarity for hadith search engine," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 18, no. 1, pp. 217–227, 2020.
[37] S. Fletcher and M. Z. Islam, "Comparing sets of patterns with the Jaccard index," *Australasian Journal of Information Systems*, vol. 22, 2018.
[38] V. Verma and R. K. Aggarwal, "A comparative analysis of similarity measures akin to the Jaccard index in collaborative recommendations: Empirical and theoretical perspective," *Social Network Analysis and Mining*, vol. 10, no. 1, 2020.
[39] S. KS and R. Shajan, "Evaluating similarity measures in collaborative filtering: Insights into accuracy, precision, and computational performance," *Georgian Education Mine, St. George's College Aruvithura*, vol. 4, no. 1, pp. 99–108, 2024.
[40] A. Z. Khan and A. Polyzou, "Session-based methods for course recommendation," *Journal of Educational Data Mining*, vol. 16, no. 1, pp. 164–196, 2024.
[41] S. Amin, M. I. Uddin, W. K. Mashwani, A. A. Alarood, A. Alzahrani, and A. O. Alzahrani, "Developing a personalized e-learning and MOOC recommender system in IoT-enabled smart education," *IEEE Access*, vol. 11, pp. 136437–136455, 2023.
[42] H. Wu, Y. Zhang, C. Ma, F. Lyu, B. He, B. Mitra, and X. Liu, "Result diversification in search and recommendation: A survey," *IEEE Transactions on Knowledge and Data Engineering*," vol. 36, no. 10, pp. 5354–5373, 2024.
[43] R. Shafiloo and K. Stefanidis, "Examining the impact of multi-objective recommender systems on providers bias," in *Proc. Workshops of the EDBT/ICDT Joint Conference*, CEUR-WS, 2024.