Journal of Engineering and Technology for Industrial Applications

ISSN ONLINE: 2447-0228

ITEGAM-JETIA

Manaus, v.12 n.58, p. 273-283. March/April, 2026.
DOI: https://doi.org/10.5935/jetia. v12i58.3022

RESEARCH ARTICLE OPEN ACCESS

AN INTEGRATED RECOMMENDATION SYSTEM FOR CUSTOMISED E-LEARNING USING THE BERT MODEL

Jatin Singh[1], Nirbhay Gautam[2], Amul Bharti[3], Priyanka Sharma[4], Vibha Gaur[5]

[1, 2, 3,4,5]Department of Computer Science, Acharya Narendra Dev College, Delhi University, Govindpuri kalkaji-110019, New Delhi

[1]http://orcid.org/0009-0007-7703-5972, [2]http://orcid.org/0009-0007-8723-3657, [3]http://orcid.org/0009-0000-7095-4924, [4]http://orcid.org/0009-0009-3341-6823, [5]http://orcid.org/0000-0001-6668-9339

Email: jatin.ae-1224@andc.du.ac.in, nirbhay.ae-1263@andc.du.ac.in, amulbharti.ae-1253@andc.du.ac.in, priyankasharma@andc.du.ac.in, vibhagaur@andc.du.ac.in

ARTICLE INFO

Article History
Received: November 28, 2025
Reviewed: January 5, 2026
Accepted: January 12, 2026
Published: March 31, 2026

Keywords:
Case-Based Reasoning (CBR),
Collaborative Filtering (CF),
Content-Based Filtering (CBF),
Bidirectional Encoder Representations from Transformers (BERT) Model,
Customized Learning,
Rule-Based Reasoning (RBR).

ABSTRACT

Digitization of education has made education more accessible. With the growing accessibility, the primary challenge for e-learning is to customise the learning environment to the needs and preferences of the learners. The learning can be customised by considering features such as persona type, skill level, learning goal, learning style, educational background, past knowledge, and memory span of learners. The tailored learning environments enhance learner engagement and significantly increase the number of learners who successfully achieve their educational goal. This work presents a recommendation system using the Case-Based Reasoning (CBR) and the Rule-Based Reasoning (RBR) with the Bidirectional Encoder Representations from Transformers (BERT) model embedded for sentence classification. The proposed integrated system has an F1 score of 0.74, indicating the balance between making correct and useful recommendations, and a higher normalised Discounted Cumulative Gain (nDGC) of 0.81 shows that the system ranks the most relevant learning modules at the top of the recommendation list. The classification of learning objectives using the BERT model into predefined domains achieved an accuracy of 95%. The system results in a structured learning path, which is more organized and engaging. The system is beneficial to novice learners, as it reduces failure rates and improves completion time of the learning process.

I. INTRODUCTION

A Recommendation system is represented as a Decision Support System (DSS), which is used to support the users in decision-making by providing the recommendations based on the analysis of crucial features of items, users, or past experiences. There are three foundational methodologies used in developing a recommendation system: Collaborative Filtering (CF) [1], Content-Based Filtering (CBF) [2], and Knowledge-Based Recommendation System (KBRS). KBRS is a privileged decision-support tool that provides recommendations based on the knowledge base designed by domain experts [3]. RBR and CBR are two major reasoning approaches in the KBRS that are widely used for developing the recommender systems. RBR is a deductive reasoning process, which discovers new findings from a rule base consisting of logical 'if-then rules' based on domain-specific knowledge, and the rule engine executes and controls the problem-solving process by matching rules of the rule base and facts of the problem.

CBR, on the other hand, utilises the previous experiences gained from the similar users to make the deductions for the new learner. It may identify past learner profiles similar to a new learner and recommend learning paths that proved successful for past learners. There are some issues associated with CBR and RBR approaches. CBR faces problems such as overspecialization, cold-start and limited content analysis, e.g., The overspecialization issue in CBR occurs when a system repeatedly suggests items very similar to the learner that he has already interacted with. The cold-start problem represents a situation when CBR is unable to provide appropriate suggestions due to insufficient data. The limited content analysis issue in CBR means that the number and type of features related to learning modules are limited.

While this aspect ensures relevance, it reduces novelty and diversity in recommendations, limiting the ability of the system to introduce new or unexpected items to learners. RBR also suffers from a knowledge acquisition bottleneck that means the process of designing rules for the knowledge base requires intensive domain expertise, updating the rules as the learner's needs change over time, and scalability issues as more learners use the system, resulting in more data, which makes the rule-base more complex and the system expensive. To address the various issues with standalone CBR and RBR, the integrated form of the CBR engine and RBR engine is proposed. To develop Knowledge-Based Recommendation Systems, there are four possible approaches for combining the CBR and the RBR [4], [5] engines. This paper utilises the integrated approach where the method of hybridisation is cascade, in which one engine refines the recommendations given by the other engine.

Utilizing the combined advantages of both reasoning approaches, the proposed system offers the accuracy possessed by the RBR engine and the experience adaptability exhibited by the CBR engine. By allowing the system to iteratively improve recommendations, cascade hybridisation helps to mitigate frequent drawbacks, including modelling complexity, overspecialization, and cold-start problems. The integration makes the system's recommendation process more accurate and flexible. The first phase in the proposed approach is the CBR engine, which uses a customized composite similarity metric called Contextual Persona-Scaled Similarity with BERT (CPSS-BERT) to retrieve past cases. BERT is an open-source machine learning framework used in natural language processing (NLP). As BERT reads text in both directions, it understands language better and produces smoother output than one-direction models, resulting in accurate results.

The learner is asked to create the profile with all essential attributes, such as the earning goal, taken as input in sentence format, and the BERT model classifies the sentences to the predefined domains using the CPSS-BERT similarity metric.The profile of the learner is matched to a past successful learner's case in the case-base. The second phase, which uses RBR, employs a set of rules that take into account variables, including learner preferences, memory span, content complexity, and prerequisite completion rate. It refines the initial learning path from the CBR to guarantee that the suggested learning modules are cognitively acceptable, instructionally cohesive, and pertinent. By using the integrated CBR and RBR approach, the proposed system makes sure that the recommendations to the learner are systematically structured and customized according to the learner's preferences.

The integrated approach overcomes the drawbacks of the CBR and RBR by making the recommendations based on the past cases that were successful in achieving the learners' goals. In case there are no past learners in the system, some synthetic data is stored in the case-base database. With the help of a synthetic dataset and rules of the RBR engine, the proposed system generates appropriate customized recommendations even if there is no past learner's case in the case-base database to match with the new learners. Through the profile modelling and custom similarity metric, the system also solves problems such as cold start and data sparsity. Data sparsity refers to the problem that occurs when the user-item interaction matrix is mostly empty because learners interact with a small number of items. The proposed integrated recommendation system provides a reliable, flexible, and learner-centred solution for the e-learning platforms. Key contributions of the proposed recommendation system are as follows:

1. An integrated recommendation framework that combines a CBR and an RBR engines in a layered pipeline. In the proposed system, CBR first retrieves relevant learning paths based on past learners' profiles. The output is then refined by RBR with pedagogical and persona-specific rules to finalize the most suitable path.
2. A solution to the cold-start problem using persona-weighted feature prioritization and BERT-based learning goal similarity allows interpretation of free-text learner goals.
3. A comprehensive set of pedagogical rules are defined based on memory span, learning goal, learning style, and skill level of learners. These rules ensure that recommendations are coherent, manageable, and align educational content with the learner's capability.
4. A progressive profiling mechanism is employed that tracks the learner's activities over time through assessment scores after completing each session and every module. Session consistency represents the engagement of the learner, resulting in improved customization and enhanced quality of the case-base for future CBR retrieval.

The remaining paper presents the following sections: Section II discusses related work of the previous research that has been done in the field of Recommendation systems and e-learning; Section III outlines the methodology of the proposed system with a block diagram; Section IV discusses the implementation of the system using case study; Section V demonstrates the results of evaluation metrics used for validation and comparative analysis of CBR, RBR and the proposed integrated system while section VI concludes the paper.

II. LITERATURE REVIEW

Recommendation systems have gained importance in the field of research over the past few decades. E-learning platforms especially need these smart systems more than ever, mainly to create customized learning paths based on the performance and habits of individual learners. The most widely used methods in the field of e-learning are CF, CBF and KBRS, but these methods have several problems: the cold-start problem, limited control in teaching, and customization of the learning environment. CF systems work with a large amount of data that is presented in the form of learning items, which is difficult for new learners or matching of new items to the existing items. Four different approaches to developing hybrid systems are gaining significance as they ensure adaptability and customization [4], [5].

Hybrid approaches such as weighted, switching, mixed, and cascade to combine the power of multiple recommendations are outlined [6]. The objective is to enhance the knowledge retention capacity of the learners through the recommended solutions. But numerous hybrid models still lack transparency and pedagogical correctness. CBR has been proposed as a suitable approach due to its ability to retrieve and adapt past learner cases that share similar objectives and learning paths. Foundational work outlines the impactfulness of CBR in problem-solving scenarios, while later studies [7] demonstrated its applicability in areas like tourism recommendation. However, when applied to education, particularly in structured domains such as cybersecurity, CBR by itself falls short of providing the necessary pedagogical dimensions.

To overcome this issue, RBR has been integrated with CBR to incorporate domain-specific rules to solve this problem. A hybrid system in medical decision-making was reported [8]. A multi-layered architecture [9] in the field of e-learning is proposed, while a hybrid adaptive system utilising expert rules and ontology matching is presented [10]. The recommendation systems are based on semantic intelligence, persona awareness, and adaptive path control [11-13], whereas MOOCs and intelligent teaching systems have been outlined, in which the number of current Recommendation systems uses various features like framework design, learner memory retention, and the objective of the learning [14], [15].

As the representation of data of knowledge database is crucial to recommend solutions, the study [16] proposed a framework for a knowledge graph that is multidimensional, in which it models the progress of the learner, prerequisite modules, and content structure and similarly. ERSDO, which is an ontology-based system [17], is presented that dynamically customise learning modules and their connections. A Crop Recommendation System [18] using machine learning is proposed that generates recommendations about suitable crops based on the climate and the soil. The data-driven approach of the work presents how adaptability is enhanced through predictive modelling. A trust-aware model using Deep Matrix Factorization [19] is proposed that presents the accuracy and reliability of the recommendation that can be improved by integrating the trust level of the user.

A movie Recommendation system is outlined [20], showing that the adaptability and customization can be improved by using the hybrid and deep learning approaches. The hybrid recommendation systems using the sentimental analysis is presented [21], which demonstrates the importance of adaptive profiling and ranking of learning modules in the field of e-learning. The above-mentioned studies describe that the most common recommendation systems in e-learning are driven by CF, CBF and KBRS, and some of them are semantic-based, ontology-based or hybrid as well, but they fail to make a recommendation system that is adaptable, transparent, pedagogically precise and customised and solves the issues such as cold start, data sparsity, overspecialization, etc.

This paper proposes a system that combines CPSS-BERT for semantic learning goal similarity enhanced by persona-weighted feature mapping, RBR for expert logic enforcement, and CBR for historical learner case retrieval. The BERT model [22], [23] has been utilised for analysing the learning goal, as it works well with modelling, classification and relevance of textual information. This multi-layered pipeline ensures that the learners are precisely matched to past successful cases, pedagogically correct, and adaptive to the learners' preferences. The proposed system is designed to use an integrated approach that tightly couples the CBR and RBR and ensures that it is adaptable to any type of domain where accuracy, pedagogy, and adaptiveness are significant.

III. RESEARCH METHODOLOGY

With education shifting from conventional classrooms to online platforms, there is a growing need for intelligent systems to help students reach their goals by providing them with customized educational content based on their individual profiles, including background, past knowledge, skill level, and so forth. The growing demand for customized learning also presents a significant challenge in the field of e-learning. This paper outlines an integrated recommendation system for customized e-learning that tightly couples the RBR engine within the CBR cycle. The proposed system helps the learners to achieve the learning goal through appropriate recommendations according to the learner's preferences, goals, and receptiveness. It uses two types of knowledge-based reasoning: CBR and RBR. The proposed system is deployed within a learning management platform designed to support customized learning paths for each learner.

It provides learner-specific profiles that include information such as persona type, learning goal, prior knowledge, skill level, memory span, preferred learning style, and content format. After the profile modelling phase, the profile of the learner is given as the input in the CBR engine, which has four processes. The first process uses a customized similarity function, CPSS-BERT; the system identifies past learner cases that are similar to the target learner. The similarity function encompasses both the profile's attribute matching and the learning goal comparison to the predefined domains using BERT embeddings, which is often used for quality e-learning through assessment, relevance of domains to the learning goal, and textual analysis [24]. The CBR engine generates initial learning path recommendations for the RBR engine. After the initial recommendations are made, the initial learning path is given as input to RBR.

A set of rules are applied to validate, refine, and organize the learning path pedagogically. These rules are designed to implement content prerequisites, align instructional difficulty with the learner's receptiveness, and ensure a pace that matches the capacity of the learner. The rules ensure that the learning path is aligned with the learner's preferences. The system supports adaptive content delivery by recommending multiple learning styles, such as text, video, labs, and projects, which is based on the learner's given preference of style and performance history. It also provides time-based estimates of course completion by measuring study sessions on a daily basis, which help learners to plan lessons more effectively.

With the combined strength of CBR for finding the past similar learner's case and the RBR engine for rules filtration, which refines the learning path recommendation, the proposed recommendation system provides a customized, adaptive, and pedagogically suitable e-learning environment. The integrated approach of CBR and RBR engines improves learner involvement, ensures efficient time utilization, and improves knowledge acquisition in e-learning environments. The proposed recommendation system is designed to develop customized learning paths and modules by analysing the learner's receptiveness, preferences, goal, content duration, session duration, and session consistency, as illustrated in Figure 1. A detailed explanation of each phase is provided below.

III.1 REGISTRATION AND LOGIN

Learners use the proposed system through an e-learning platform that works for both new and existing learners. Existing learners can directly log in to the e-learning system and get access to the customized dashboard, which contains progress reports of the previously chosen learning paths. New learners are asked to complete a registration process before logging in. Login data contains details of the learner, such as email ID, password, age, gender, and so on. Once the successful authentication is completed, new learners are redirected to the profile modelling phase.

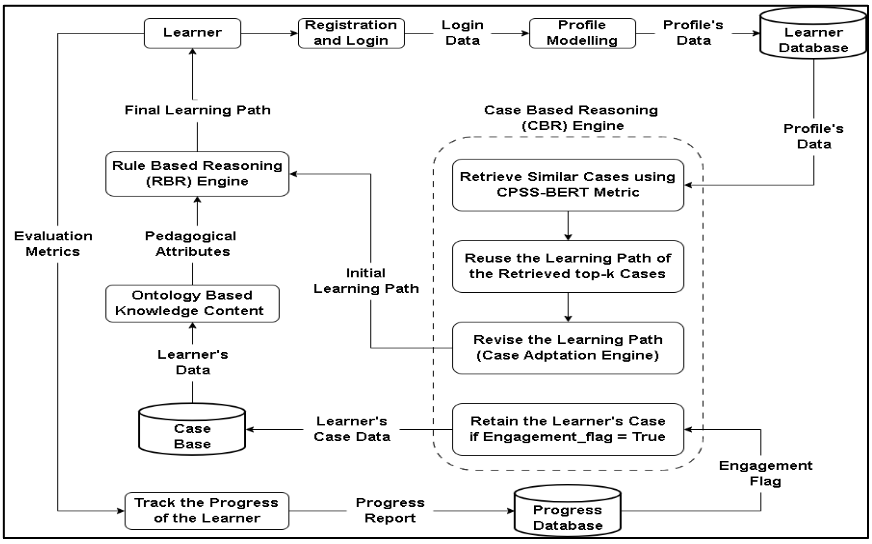

Figure 1: Block Diagram of the Proposed Integrated Recommendation System for E-Learning.
Source: Authors (2026).

III.2 PROFILE MODELLING

Following the completion of authentication, new learners are directed to the profile modelling phase, where attributes of the learner that are most crucial for an online education, such as the learner's persona type, learning goal, skill level, background, memory span, learning style, and prior knowledge, are gathered through a questionnaire. A quiz is designed to gauge the previous knowledge of the learner using the learning goal, skill level and background. The content recommendations and a personalised learning path are guided by the learner's profile. The next phase involves the CBR engine, which is the core of the proposed system.

III.3 CASE-BASED REASONING (CBR) ENGINE

Once the profile modelling phase is complete, then the attributes of the learner’s profile are given as the input to the CBR Engine as the profile data, as illustrated in Figure 1. The profile data contains the learning goal, the learning preferences, and receptive capacity. CBR is employed for retrieving and adapting learning paths from previously retained learner cases in the case-base database that are similar to the new learner. The retrieval in the CBR engine is done by using a customized composite similarity function called Contextual Persona-Scaled Similarity with BERT Fusion (CPSS-BERT).The CBR engine operates through the CBR cycle, consisting of four steps—Retrieval, Reuse, Revision and Retain—that systematically align and adapt leaner profiles to generate customized learning path recommendations. Algorithm 1 for a CBR, which returns the final adapted case, is described below. This helps the system to learn from past learner experiences while adjusting to the needs of each new user. Each phase of the CBR engine is discussed below in detail.

Algorithm 1

1. *Initialize pre-trained* ***BERT-based similarity model****,* $CPSS_{BERT}$.
2. *Initialize empty list:*
$Scored\_Cases \leftarrow \emptyset$
3. For each case $c \in KB$:
a. *If* $c.description$ *is missing, set*
$c_{desc} \leftarrow concat(c.module\_titles)$
b. *Compute* ***BERT-based semantic similarity*** *between* u *and* c:
$$Sim_{BERT}(u,c) = 1 - \frac{g_u \cdot g_c}{\|g_u\|\|g_c\|}$$
where g_u *and* g_c *are embedding vectors of* u *and* c.
c. *Compute* ***attribute-level similarities (if available)****:*
(i) Categorical attribute:
$$Sim_f(u,c) = 1 - \frac{|index(u) - index(c)|}{max\_distance}$$
(ii) Set-based (multi-select) attribute:
$$Sim_f(u,c) = \frac{|u \cap c|}{|u \cup c|}$$
(iii) Numerical score attribute:
$$Sim_f(u,c) = 1 - \frac{|u-c|}{max\_score}$$
d. *Compute* ***overall similarity*** *by combining structured and semantic similarities:*
$$Sim_{total}(u,c) = \alpha \sum_{f=1}^{n} w_f \cdot Sim_f(u,c) + (1-\alpha) \cdot Sim_{BERT}(u,c)$$
e. *Append cases and its score:*
$Scored\_Cases \leftarrow Scored\_Cases \cup \{(c, Sim_{total}(u,c))\}$
4. *Sort* $Scored_{Cases}$ *in descending order of* $Sim_{total}(u,c)$.
5. *Select top-k similar cases:*
$Top_k = Sorted(Scored\_Cases)[1:k]$
6. *Return* Top_k.

Source: Authors, (2026).

III.3.1 Retrieve: CPSS-BERT Similarity Matching

On the completion of the profile modelling phase, the learner's profile is matched against a case-base of prior learners using a composite similarity function called CPSS-BERT. Each structured attribute f in the learner profile such as skill level, memory span, background, learning style, prior knowledge and assessment score, are compared with the corresponding attribute in a previous case $\boldsymbol{c}$, using tailored similarity functions that return a normalised similarity score $\boldsymbol{Sim_f(u,c)}$ in the range [0,1], where:

i) $\boldsymbol{u}$: the new (target) learner's profile.
ii) $\boldsymbol{c}$: a past learner case from the case-base.
iii) $\boldsymbol{f}$: a specific feature/attribute.
iv) $\boldsymbol{w_f}$**:** the persona-specific weight assigned to the feature $\boldsymbol{f}$, based on its relevance to the learner's motivational type.

For different types of attributes, this study employs different metrics such as categorical distance, set-based Jaccard, and numerical score distance [25], [26]. For ordinal features such as skill level, the similarity is computed using a normalized distance metric, as given in (1) as,

$$Sim_f(u,c) = 1 - \frac{|index(u) - index(c)|}{\max\_distance} \tag{1}$$

In the ordinal similarity function, $index(u)$ and $index(c)$ represent the respective index positions of the learner and the historical case for a given ordinal feature. The variable max_distance denotes the maximum possible index difference in this ordered list, which ensures normalisation of the similarity score between 0 and 1. When the new learner and past learner case have multi-select categorical preferences, the Jaccard similarity shown in (2), which works on categorical data, is computed.

$$Sim_f(u,c) = \frac{|u \cap c|}{|u \cup c|} \tag{2}$$

The Jaccard similarity is used for comparing categorical set-based attributes such as learning styles based on the intersection and union of preferences between the new learner and a past learner case. In this context, $|u \cap c|$ denotes the count of shared styles of learning between the learner u and the case c. Conversely, $|u \cup c|$ denotes the total number of distinct learning styles present across both profiles, capturing the full diversity of preferences expressed by either case. For numerical attributes such as assessment scores obtained from prior knowledge quizzes and module test scores, similarity is calculated by normalizing the absolute difference between learner and case scores using (3).

$$Sim_f(u,c) = 1 - \frac{|u-c|}{\max\_score} \tag{3}$$

The numerical similarity is calculated using assessment scores, where u represents the quiz score of the learner, c is the score from a past learner case, and max_score represents the maximum score that is possible (commonly 100). This approach normalizes the similarity between the two scores on a scale from 0 to 1. Once similarity scores are calculated for features like skill level, learning style, and memory span, they are scaled using feature-specific weights w_f, that are assigned dynamically based on the learner's persona. For example, a beginner or new learner may benefit more from features like past knowledge and memory span, while professional may focus on learning objectives and preferred content type. The retrieve step comprehends the meaning of the learning goal in sentence form given by the learner, and the similarity between the learner's and the new learner's goal is calculated using (4), which is based on the cosine similarity measure used in Sentence BERT [27].

$$Sim_f(u,c) = 1 - \frac{g_u \rightarrow .\ g_c \rightarrow}{\| g_u \rightarrow \| .\ \| g_c \rightarrow \|} \tag{4}$$

In (4), $g_u \rightarrow$ and $g_c \rightarrow$ represent the semantic embedding vectors of the learner's and the past learner's case goals, respectively, $\| g_u \rightarrow \|$ and $\| g_c \rightarrow \|$ denote the Euclidean magnitude of vectors. This component ensures adjustment between the semantic intent of the learner and previous learners, even if text given by the learner differs significantly. The final overall similarity score $Sim_{total}(u,c)$ is derived in (5) by combining the weighted sum of all structured similarities with the semantic similarity:

$$Sim_{total}s(u,c) = \alpha \sum_{f=1}^{n} w_f . Sim_f(u,c) + (1-\alpha) .\ Sim_{BERT}(u,c) \tag{5}$$

Where the parameter α ∈ [0,1] is a tuneable weight that controls the comparative importance of the structured feature-based resemblance and the semantic correlation derived from BERT embeddings. A higher value of α places more emphasis on the structured attributes such as skill level, memory span, and learning style, while a lower value shifts the focus towards semantic alignment of learning goals. Here, n represents the total number of structured features, w_f is the persona-weighted importance of feature f, and $Sim_f(u,c)$ is the similarity score for feature f between the current learner u and a historical case c. The term $Sim_{BERT}(u,c)$ quantifies the semantic similarity between the new learner's stated goal and of the past learner's case, using cosine similarity on BERT-generated embeddings. The top-k past learner's cases with the highest total similarity scores are selected using the CPSS-BERT similarity function for generating the learning path of the new learner.

III.3.2 Reuse: Learning Path Retrieval

The learning paths are extracted from the top-k retrieved past learners' cases. These paths consist of sequences of modules, for example, videos, texts, labs, etc., associated with the target learner's goal. Modules frequently appearing among the top-k cases are prioritized, and their success outcomes, such as completion rate and failure rate, are factored into an initial learning path for the new learner. Once the initial learning path is generated, it is transferred as input to the next phase—the case-adaptation engine.

III.3.3 Revise: Case Adaptation Engine

After the retrieval and reuse process completed by the CBR Engine, the proposed system presents a Case Adaptation Engine, which implements the Revise step. The learning paths are adapted according to the new learner profile, which are retrieved from the previous past cases. The adaptation process considers various major personal characteristics to customize the learned path based on the current learner's environment, such as the learner's daily study slots available, as they represent typical learning habits that are used to determine pace and estimated completion time for each module. The learner's memory span, whether short, medium, or long, impacts the intervals of repetition content and review modules to facilitate efficient learning.

The learner's learning style preferences are included in the learning path given by the recommendation system. Each module on the recommended path is thoroughly analysed and tailored to fit the unique attributes of the learner.The duration of each module is adjusted based on the daily hours offered by the learner. The level of educational content is customised based on the learner's previous knowledge and capability; individuals with lower evaluation grades or shorter retention capacity are offered basic modules and supplementary revision-based material. The content format is tailored to the learner's preferred style; for example, text modules may be replaced with videos or interactive labs.

Adaptation rules ensure that the retrieved learning path from the case-base is adapted according to the learner's needs and preferences, which results in the initial learning path. This step is executed before the RBR engine phase, which takes the initial learning path as input and refines it more. The Case Adaptation Engine follows Algorithm 2 described below. The customized recommendation contains a studied set of learning modules consisting of theoretical material, videos, labs, and evaluations customized to the ability of the learner, memory span, learning style, and declared learning goals. It is a basic learning path, maximized for relevance and engagement, that will be further tested and finalized by the RBR engine to provide pedagogical integration and lesson alignment.

III.3.4 Retain: Case Storage and Update

After the complete representation of the case-base data, rule refinement, learner interaction, and progress tracking, the system monitors the outcomes and retains the updated learner case, including profile, learning path, and results in the case-base database. This enables continuous improvement and adaptability of future recommendations. If a learner successfully completes the learning path, the system stores that learner's case in the case-base database with the learning path for future case-based retrieval, as shown in Figure 1 above.

III.4 ONTOLOGY-BASED KNOWLEDGE MODELLING

The ontology model organizes the knowledge content into a set of interconnected entities that collectively define the pedagogical and learner-centred aspects of the system. E-learning course strategy is represented as the central entity that connects the four major components, such as learner's profile, learner's behavior, learner performance, and content structure. The entities and their attributes are stored in the data dictionary that serve as the semantic input source for the RBR engine in the system. The integration of the ontology model with the RBR engine forms the semantic bridge between structured data and intelligent decision-making. During execution, the ontology retrieves relevant pedagogical and learner-specific information from the data dictionary and provides it as semantically structured input to the RBR engine. The RBR engine then applies predefined instructional and pedagogical rules to match ontology attributes with suitable learning modules. The working of the RBR engine is explained in detail in the following phase.

Algorithm 2

```
1. Initialize Initial_Learning_Path ← Ø (empty)
2. IF learning_Styles is a single string, THEN convert it into a list.
3. For each (Case, Score) in Extracted_Learning_Paths:
   a. Set Modules_List ← Case.Modules
   b. Initialize Filtered_Modules ← Ø (empty)
   c. For each Module in Modules_List:
        For each Style in Preferred_Styles:
           IF Style partially matches Module.Format (case-insensitive),
              THEN Add Module to Filtered_Modules and break the inner loop.
   d. IF Filtered_Modules is empty, THEN assign original Modules_List as fallback.
   e. Update Case.Modules ← Filtered_Modules
   f. Append updated Case to Initial_Learning_Path
   g. IF optional case limit k is reached, THEN exit loop.
4. Return Initial_Learning_Path
End
```

Source: Authors, (2026).

III.5 RULE-BASED REASONING (RBR) ENGINE

The initial learning path that is generated by the CBR engine based on the learner's profile is fed to the RBR engine as input that further enhances the learning path through pedagogical logic. The RBR engine follows Algorithm 3 described below that fine-tunes the learning sequence. The role of the RBR engine is to enhance the instructional validity of the learning path by applying domain-specific rules developed by a domain expert of curriculum principles and prerequisite modules. The refinement process begins with the retrieval of pedagogical metadata from the case-base database. This metadata covers aspects such as module difficulty level, prerequisite relationships, learning modules, and memory span indicators. These pedagogical descriptors are organised in the learning modules in a smart way so that the order of recommended modules matches both the learner's personal needs and proper educational design principles. The set of rules defined by the domain expert are applied to the initial learning path generated by the system to refine it iteratively.

III.6 PROGRESS TRACKING MODULE

The proposed system tracks the progress of the learner through learning activities, which helps to observe the performance and engagement level of the learner. It also suggests areas where the learner can improve. The data of the progress tracking is stored in the progress database that acts as a centralized hub for containing and maintaining the actions performed by the learners during the engagement with the proposed system. Data in the progress database contains session-specific metrics like session duration and session consistency and performance metrics like completion rate, failure rate, and recurrence rate of the learner. Learning preferences are tracked by observing the formats that the learner interacted with most, such as videos, labs, text, or projects. The database provides validation checks for the learner's initially declared preferences. The system also takes into account content structure, such as verifying whether learners are engaging] well with the structured, and pedagogically accurate modules and whether the content is up to date.

This may flag any system inconsistencies for subsequent revision. Based on these factors, the system calculates an overall engagement assessment called Engagement Flag. This flag indicates various states of the learners: *Success*, denoting completion of the learning path, and *Recurrence*, reflecting repeated revisits of modules, or *Failure*: showing the low engagement or incomplete modules. The results, along with session metrics, are stored in the Progress Database. If a learner successfully completes the learning path, the system stores that learner's case in the case-base database with the learning path for future case-based retrieval, as shown in Figure 1 above. The refining of the learning path makes sure that the proposed recommendation system evolves as real learner data increases and ensures the adaptability and correctness of the system.

Algorithm 3

1. Initialize Final_Learning_Path ← Ø
2. Retrieve Memory_Span, Learning_Style, Skill_Level from Learner_Profile
3. For each Case in Adapted_Cases:
a. Set Modules_List ← Case.Modules
b. Initialize Refined_Modules ← Ø
***Rule** 1 - Memory Retention:*
IF Memory Span is short, THEN add a learning model M with revision frequently.
IF Memory Span is long, THEN add a long duration learning modules.
***Rule** 2 – Define the difficulty level of the Module:*
For each learning module M in Refined Modules,
IF Module.Difficulty > Skill_Level of learner, Then add basic modules in the learning path.
***Rule** 3 – Structure the content according to the preferred Learning style:*
For each M in Refined Modules,
IF Module.Format ≠ Learning Style of the learner, Then insert a different style of modules.
4. Assign Case.Modules ← Refined Modules
5. Add the learner's Case to Final Learning Path
6. Return Final Learning Path
End

Source: Authors, (2026).

IV. IMPLEMENTATION OF THE PROPOSED SYSTEM

To demonstrate the application of the proposed integrated recommendation system, a detailed walkthrough is presented for a new learner interacting with the e-learning platform offering cybersecurity education. This case study illustrates how each phase of the methodology is applied to construct a customized and pedagogically precise learning path for the learner, using structured profile modelling, integrated case-based rule-based reasoning-driven recommendation, and intelligent content sequencing. Python's Transformers and TensorFlow 2.0 libraries were used to implement the BERT model along with the Hugging Face library of Python transformers. The proposed recommendation system is implemented using Python 3.13.0 and React.js is used for the front-end framework. The case study for the new learner is mentioned below.

IV.1 CASE STUDY

A new learner, *Riya Sharma*, visited the e-learning platform for the first time. Upon accessing the registration page, Riya created an account by entering her name, email, and basic credentials. Once Riya registered and logged in to the system with the valid login credentials successfully, she was redirected to the profile modelling phase.

It is the first phase of the proposed system, where the learner gives inputs and answers the various questions like the persona of the learner; that is, which category the learner belongs to, such as college student, job seeker or professional looking for a career change. Riya selected the college student as her persona. The next set of questions include the learning goal in the form of a sentence. The system used the BERT model to classify the learning goal to the predefined domains. For the cybersecurity e-learning, she entered the learning goal as “I want to become an ethical hacker”, which was then classified to the ‘ethical hacking’ domain of the proposed system. This classification helps the system to ensure that the learning path only contains the learning modules related to the given learning goal of the learner. Furthermore, she selected her skill level, which had the three options such as beginner, intermediate or advanced. She selected the beginner level as her level of skill. She was also asked to choose from which background she belonged, such as tech or non-tech. She selected *tech* as her background, as she knew about the computer fundamentals and could complete basic modules without any difficulty.

She was also asked for her preferred learning style, such as text, video, lab or projects. The system presented her a quick test on the principles of general computing and logic to evaluate her past knowledge. Riya scored 30%, indicating limited baseline knowledge. This score was recorded and used to tailor the learning path, ensuring she begins with foundational concepts. Subsequently, Riya answered behavior-based questions to estimate her memory span. Based on her responses, she was classified as having a *medium* memory span, suggesting an average retention capacity and a need for moderately paced content review. Upon the completion of the profile modelling phase, all collected data was stored in the learner profile database. The structured profile was input to the CBR engine, which started the recommendation pipeline by finding similar past learner cases using Algorithm 1 that employs the CPSS-BERT similarity function using (5) to combine the similarities of each attribute individually as obtained using (1-4). Overall similarity of learners was obtained using (5) depending on the weights given by the academicians to the attributes for persona types as shown in Table 1.

Table 1: Weight assigned to attributes according to the Persona of the learner.

Feature	Student	Career Switcher	Enthusiast	Professional
Skill Level	0.25	0.30	0.30	0.20
Prior Knowledge (Quiz)	0.30	0.25	0.25	0.20
Memory Span	0.20	0.10	0.15	0.15
Learning Style	0.10	0.15	0.10	0.15
Content Preference	0.10	0.10	0.10	0.20
Learning Goal (BERT)	0.05	0.10	0.10	0.10

Source: Authors, (2026).

As there were 3 past cases stored in the case-base database, namely, Case A, Case B, and Case C. The values of similarity for each case attribute are given below:

Case A: Skill = 1.00, memory span = 0.50, learning style = 0.50, prior knowledge = 0.98, learning goal = 0.98, background = non-tech (0),
Case B: Skill = 0.50, memory span = 0.50, learning style = 0.25, prior knowledge = 0.65, learning goal = 0.225, background = non-tech (0),
Case C: Skill = 1.00, memory span = 1.00, learning style = 0.25, prior knowledge = 0.95, learning goal = 0.99, background = tech(1).

The system first calculated the similarity of the new learner to Case A, where each attribute was computed and multiplied by the weights. And the overall similarity score was obtained as 0.795. Similarly, the similarity of Case B and Case C was computed, where the achieved similarity scores for Cases B and C were obtained as 0.481 and 0.834, respectively. The top similar past learners were Case C (highest) and Case A (the second highest). From these retrieved cases, top similar cases such as A and C were passed to Algorithm 2, which extracted the learning paths from the top similar past learners and shortlisted the desired learning modules, which resulted in the initial learning path as depicted in Figure 2. The resulting initial learning path was fed as input to the RBR refinement engine, where pedagogical rules were applied. The RBR engine used Algorithm 3 for refining the learning path. The rules of RBR filtered out modules as per Riya’s current skill level, reordered the content based on prerequisites, and matched content formats to her preferred style.

After rule refinement, a final customized learning path was generated and displayed to Riya on her dashboard as shown in Figure 3. The dashboard also included an introductory message highlighting the estimated duration, content types included, and next phases to begin the learning journey. As Riya engaged with the system and interacted with learning modules, the system tracked the learner’s activity and behavior, which helps in computing her performance metrics. The system analysed Riya Sharma's progress data to calculate the performance metrics, such as completion rate, failure rate, and recurrence rate. Out of 10 learning modules in her ethical hacking learning path, Riya successfully completed 8, failed 1, and left 1 module incomplete. Furthermore, the system calculated a completion rate of 80%, indicating a successfully achieved learning goal and strong engagement of the learner.

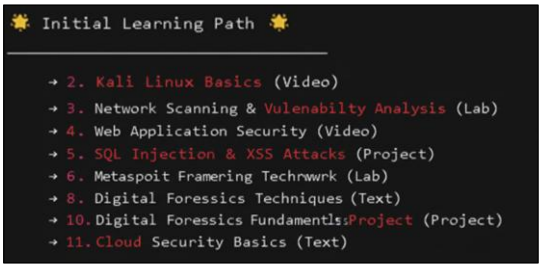

Figure 2: The initial learning path extracted and adapted to learner’s profile using Algorithm 2.
Source: Authors, (2026).

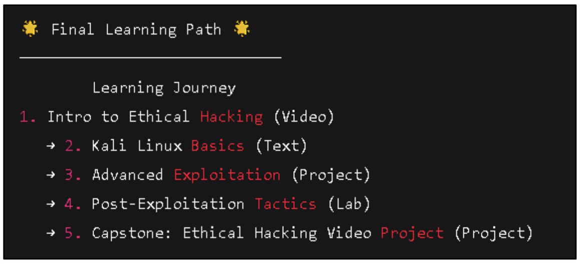

Figure 3: The final learning path generated according to the learner's need using Algorithm 3.
Source: Authors, (2026).

The failure rate of the learner was obtained as 20%, representing challenges in advanced topics and leaving the learning journey in the middle. Revisiting three modules, Riya exhibited a 37.5% recurrence rate, which corresponded to her moderate memory span and need for reinforcement.

V. RESULTS AND ANALYSIS

In this section, evaluation of the proposed system was performed. A total of 100 participants were considered for the study. The participant's learning goals were classified in six predefined cybersecurity domains: web security, bug bounty hunter, network security, penetration testing, ethical hacking, and Android security, The study may be extended to other domains of different subjects. The integrated system makes the profiles of a learner, and uses the BERT model to analyse the learning goal to classify it. The evaluation of the BERT model is given below.

V.1 BERT MODEL PERFORMANCE

The BERT model used the publicly available "Attack Dataset" from Kaggle.com [28]. The dataset was processed to handle inconsistencies, archive class balance, and ensure relevance for learning goal classification. The performance of the BERT model was evaluated using the standard classification metrics—Accuracy, Precision, Recall, and F1-score. Here, True Positives (TP) represent sentences correctly classified into their intended category, True Negatives (TN) denote sentences not belonging to the desired category, False Positives (FP) indicate sentences incorrectly assigned to a chosen category, and False Negatives (FN) correspond to sentences belonging to a category but were not correctly classified. The BERT model was trained on a total dataset of 14,134 labelled sentences, where the values of TP, TN, FP and FN were obtained as 11,800, 1,650, 344 and 340, respectively. The resulting system achieved an accuracy of 0.95, a precision value of 0.97, a Recall value of 0.971, and an F1-score of 0.971. These results confirm the BERT model's robust contextual understanding, enabling accurate and intelligent domain classification. The subsequent section computes the metrics used for validating the proposed recommender system.

V.2 VALIDATION OF THE PROPOSED SYSTEM

The performance of the proposed system was evaluated through the metrics mentioned below. Metrics such as Precision, Recall, F1-score, Normalized Discounted Cumulative Gain (nDCG) [24], the Instructional Coherence Rate (ICR) [25], Learning Path Effectiveness (LPE) [29], and the Engagement Score [30] were utilised to validate the projected system. The three different approaches of KBRS, such as CBR-only, RBR-only and the proposed integrated system, were compared. The resultant values of the validation metrics are given in Table 2, which outlines that the proposed integrated system outperforms the standalone CBR and RBR systems. The integrated system, for instance, obtained the highest precision (top 5 recommendations) score of 0.78, indicating that the top five suggestions were pertinent. It presented a higher percentage of all appropriate resources for learners, as shown by Recall (Top 5 recommendations) score of 0.70. The value of the F1-score (top 5 recommendations) was obtained as 0.74, which strikes a compromise between Recall and Precision, representing a balance between suitable recommendations and identifying all useful learning modules.

Table 2: Comparison of CBR, RBR and Integrated System.

Metric	CBR Only	RBR Only	Proposed Integrated Recommendation System
Precision	0.68	0.70	0.78
Recall	0.60	0.58	0.70
F1-Score	0.64	0.63	0.74
nDCG	0.69	0.68	0.81
Engagement Score (/10)	6.2	6.5	8.4
ICR	0.72	0.76	0.89
LPE	0.75	0.77	0.91

Source: Authors, (2026).

With a nDCG of 0.81 (Top 5 recommendations), the integrated system had the highest ranking, indicating the most relevant items were placed higher in the list of recommendations. The comparison of CBR-based, RBR-based and integrated CBR-RBR, using the set of metrics is illustrated in Figure 4. The results show that both individual models were surpassed by the integrated proposed approach across the evaluation metrics.

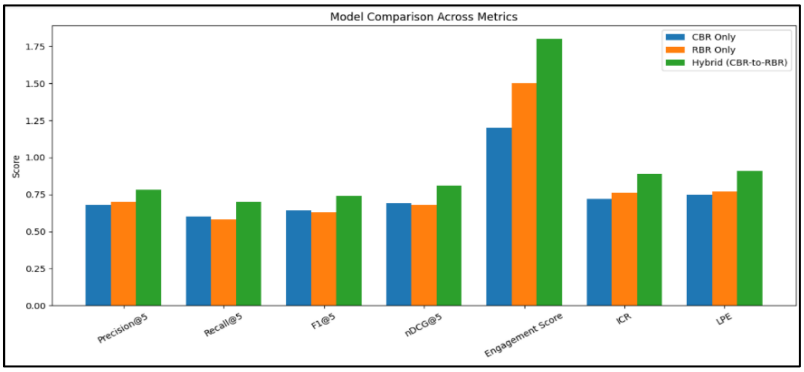

Figure 4: Comparison of CBR, RBR and Integrated System.
Source: Authors, (2026).

VI. CONCLUSIONS

This study proposed an integrated recommendation system using the BERT model that outperforms the standalone CBR-only and RBR-only systems. The integrated system overcomes the issues of the traditional recommendation systems like CF, CBF, and KBRS. The proposed system solves the cold-start problem by integrating the pedagogically and expert-designed rules of the RBR engine in the CBR cycle for non-existing past learners' cases. The CBR engine addresses the issue of data sparsity by using the past cases only if explicit data of the learner was insufficient, and in that case, RBR provides a fallback expert recommendation. The custom CPSS-BERT metrics used the BERT model to classify the learning goals of the learners, which helped the system to recommend more customised learning paths that resolved the overspecialization to some extent.

The integrated recommendation system achieved an engagement score of 8.4 out of 10, suggesting that learners showed the highest engagement and completed most of the modules in the suggested learning path. With an Instructional Coherence Rate (ICR) of 0.89, most of the recommended learning paths followed the right order and fulfilled the necessary requirements of learners. The Learning Path Effectiveness (LPE) value of 0.91 indicated that a majority of the learners effectively completed the prescribed paths and achieved high scores in the post-assessment phase. The proposed system provides a consistent and learner-centred solution for the e-learning platforms.

VII. AUTHOR'S CONTRIBUTION

Conceptualization: Vibha Gaur.
Methodology: Jatin Singh, Priyanka Sharma.
Investigation: Amul Bharti and Jatin Singh.
Discussion of results: Amul Bharti, Nirbhay, Jatin Singh, Vibha Gaur
Writing—Original Draft: Jatin Singh.
Writing—Review and Editing: Amul Bharti and Nirbhay.
Resources: Amul Bharti.
Supervision: Vibha Gaur, Priyanka Sharma.
Approval of the final text: Vibha Gaur .

VIII. ACKNOWLEDGMENTS

The authors are grateful to the Acharya Narendra Dev College, University of Delhi, for carrying out the work.

IX. REFERENCES

[1] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," in Proc. 10th Int. Conf. on World Wide Web, 2001, pp. 285–295, DOI:10.1145/371920.372071.

[2] Q. Yang, Z. Chen, and Y. Yu, "Explainable recommendation: A survey and new perspectives," Foundations and Trends in Information Retrieval, vol. 14, no. 1, pp. 1–101, 2019, DOI:10.1561/1500000066.

[3] A. Aamodt and E. Plaza, "Case-based reasoning: Foundational issues, methodological variations, and system approaches," AI Communications, vol. 7, no. 1, pp. 39–59, 1994, DOI:10.3233/AIC-1994-7104.

[4] W. S. Admass and Y. Y. Munaye, "Integrating case-based and rule-based reasoning for diagnosis and treatment of mango disease using data mining techniques," International Journal of Information Technology, 2023. DOI: 10.1007/s41870-023-01587-y

[5] R. Saraiva, M. Perkusich, L. Silva, H. Almeida, C. Siebra, and A. Perkusich, "Early diagnosis of gastrointestinal cancer by using case-based and rule-based reasoning," Expert Systems with Applications, 2016, DOI:10.1016/j.eswa.2016.05.026.

[6] R. Burke, "Hybrid recommendation systems: Survey and experiments," User Modelling and User-Adapted Interaction, vol. 12, no. 4, pp. 331–370, 2002, DOI:10.1023/A:1021240730564.

[7] T. A. Alemu, A. K. Tegene, and A. N. Tarekegn, "Recommender system in tourism using case based reasoning approach," International Journal of Information Engineering and Electronic Business, vol. 9, no. 5, pp. 34–43, 2017, DOI:10.5815/ijieeb.2017.05.05.

[8] K. A. Kumar, Y. Singh, and S. Sanyal, "Hybrid approach using case-based reasoning and rule-based reasoning for domain independent clinical decision support in ICU," Expert Systems with Applications, vol. 34, no. 1, pp. 492–502, 2008. DOI:10.1016/j.eswa.2007.09.054.

[9] L. Roux, P. Dagorret, P. Etcheverry, T. Nodenot, C. Marquesuzaà, and P. Lopistéguy, "A multi-layers architecture for an e-learning hybrid recommendation system," Procedia Computer Science, vol. 203, pp. 210–217, 2021, DOI:10.1016/j.procs.2022.07.029.

[10] V. Demertzi and K. Demertzis, "A hybrid adaptive educational eLearning project based on ontologies matching and recommendation system," arXiv preprint, 2020, DOI:10.48550/arXiv.2007.14771.

[11] O. Maryam, H. Ashraf, T. Amjad, and N. Jhanjhi, "Improved e-learning based recommendation systems: A survey," Preprints.org, 2023, DOI:10.20944/preprints202307.0350.v2.

[12] D. Poornima and D. Karthika, "E-learning recommendation system and classification techniques: A survey," Naturalista Campano, vol. 28, no. 1, 2024.

[13] S. Souabi, A. Retbi, M. K. Khalidi Idrissi, and S. Bennani, "Recommendation systems on e-learning and social learning: A systematic review," Electronic Journal of e-Learning, vol. 19, no. 5, pp. 1–20, 2021, DOI:10.34190/ejel.19.5.2482

[14] Y. Lin, S. Feng, F. Lin, W. Zeng, Y. Liu, and P. Wu, "Adaptive course recommendation in MOOCs," Knowledge-Based Systems, vol. 224, p. 107085, 2021, DOI:10.1016/j.knosys.2021.107085.

[15] Y. Madani, H. Ezzikouri, M. Erritali, and B. Hssina, "Finding optimal pedagogical content in an adaptive e-learning platform using a new recommendation approach and reinforcement learning," Journal of Ambient Intelligence and Humanized Computing, vol. 11, no. 10, pp. 3921–3936, 2020, DOI:10.1007/s12652-020-01771-y.

[16] D. Shi, T. Wang, H. Xing, and H. Xu, "A learning path recommendation model based on a multidimensional knowledge graph framework for e-learning," Knowledge-Based Systems, vol. 195, p. 105618, 2020, DOI:10.1016/j.knosys.2020.105618.

[17] M. Amane, K. Aissaoui, and M. Berrada, "ERSDO: E-learning recommendation system based on dynamic ontology," Education and Information Technologies, vol. 27, pp. 7549–7561, 2022, DOI:10.1007/s10639-022-10960-4.

[18] B. N. Mohapatra and V. Kale, "Crop recommendation system using machine learning," Journal of Emerging Technologies and Innovative Applications (JETIA), vol. 10, no. 48, 2024, DOI:10.5935/jetia.v10i48.1186.

[19] P. M. Paithane, "Trust aware recommendation using deep matrix factorization model," Journal of Emerging Technologies and Innovative Applications (JETIA), vol. 10, no. 48, 2024, DOI:10.5935/jetia.v10i48.941.

[20] Y. Ariyanto and T. Widiyaningtyas, "A systematic review of movie recommendation systems," Journal of Emerging Technologies and Innovative Applications (JETIA), vol. 10, no. 47, 2024, DOI:10.5935/jetia.v10i47.1052.

[21] I. Uddin, A. S. Imran, K. Muhammad, N. Fayyaz, and M. Sajjad, "A hybrid e-learning recommendation integrating adaptive profiling and sentiment analysis," Web Semantics, vol. 72, p. 100700, 2022, DOI:10.1016/j.websem.2021.100700.

[22] P. Singh, R. Pachisiya, S. Nehra, and V. Gaur, "Quantifying the credibility of e-learning systems using the BERT model," ITEGAM-JETIA, vol. 12, no. 57, pp. 519–528, Jan. 2026, DOI:10.5935/jetia.v12i57.3166.

[23] A. Varshney, Y. Kapoor, V. Chawla, and V. Gaur, "A novel framework for assessing the criticality of retrieved information", International Journal of Computing and Digital System, Vol. 11(1), pp. 1229-1244, 2022, DOI:10.12785/ijcds/1101100

[24] R. Kaur, R. Sharma, A. A. Jha, and V. Gaur, "A quantitative approach for appraising quality of online education," Journal of Engineering Education Transformations, vol. 38, no. 2, pp. 164–177, 2024, DOI:10.16920/jeet/2024/v38i2/24187.

[25] S. Jadon and P. Patil, "A comprehensive survey of evaluation techniques for recommendation systems," arXiv preprint, 2023, DOI:10.48550/arXiv.2312.16015.

[26] N. Villeneuve, C. Perreault, D. Bédard, and G. Samson, "An instructional coherence ratio as a performance indicator for improving the quality of teaching practices," in Proc. 16th Int. Technology, Education and Development Conf. (INTED), 2022.

[27] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using Siamese BERT-networks," arXiv preprint, 2019, DOI:10.48550/arXiv.1908.10084.

[28] TeamInCribo, "Cyber security attacks dataset," Kaggle, 2023. [Online]. https://www.kaggle.com/datasets/teamincribo/cyber-security-attacks.

[29] Y. Lin and Z. Wu, "Learning path recommendation enhanced by knowledge tracing and large language model," Electronics, vol. 14, no. 22, p. 4385, 2025, DOI:10.3390/electronics14224385.

[30] X. Solé-Beteta, J. Navarro, B. Gajšek, A. Guadagni, and A. Zaballos, "A data-driven approach to quantify and measure students' engagement in synchronous virtual learning environments," Sensors, vol. 22, no. 9, p. 3294, 2022, DOI:10.3390/s22093294.