REVOLUTIONIZING E-LEARNING: PERSONALIZED LEARNING PATHS WITH LSTM AND COLLABORATIVE FILTERING

1st Dr.A.Karunamurthy , 2nd R.Taruna

Associate Professor,Department of computer Applications,

Sri Manakula Vinayagar Engineering College(Autonomous), Puducherry 605008,India

Karunamurthy26@gmail.com

Post Graduate student,Department of computer Applications,

Sri Manakula Vinayagar Engineering College (Autonomous),Puducherry 605011 Tarunat453@gmail.com

ABSTRACT:

E-learning, rooted in formalized teaching methods and supported by electronic resources, has revolutionized education by integrating computers and the Internet to provide flexible, accessible learning opportunities. Initially met with skepticism due to its perceived lack of human interaction, e-learning has significantly evolved, particularly with the adoption of dynamic recommendation systems. Traditional recommendation systems often relied on algorithms like k-nearest neighbors (K-NN), which, while simple and intuitive, struggle with scalability and adaptability in handling complex, evolving user data. To address these limitations, a more advanced, dynamic e-learning system is proposed, combining Natural Language Processing (NLP) and Long Short Term Memory (LSTM) networks. NLP plays a crucial role in interpreting and understanding user queries expressed in natural language, enabling the system to comprehend the learner's intent more effectively. Meanwhile, LSTMs, a type of Recurrent Neural Network (RNN), are capable of learning from sequential data, making them well-suited for tracking user behavior and predicting future learning needs based on past interactions. This synergy between NLP and LSTMs enables the system to dynamically adapt its content recommendations to suit individual learning preferences and goals. Unlike traditional models, the proposed system continuously learns and evolves, enhancing its predictive accuracy and user engagement over time. By leveraging these advanced technologies, the proposed e-learning model not only overcomes the constraints of earlier systems but also offers a more personalized, efficient, and engaging learning experience. This marks a significant step in the technological advancement of e-learning, aligning digital education with the needs of modern learners.

Keywords: the k-nearest neighbors (K-NN), Long Short Tear (LSTM s), BERT, e-learning

I. INTRODUCTION:

E-learning, short for electronic learning, is an educational approach that leverages electronic technologies and digital resources to facilitate the acquisition of knowledge and skills. It encompasses a diverse range of instructional methods delivered through digital platforms, such as the internet, computers, and mobile devices. Unlike traditional classroom-based education, e-learning provides flexibility in terms of time, location, and pace of learning. This method allows learners to access educational content from virtually anywhere, fostering a globalized and accessible approach to education. E-learning platforms often feature multimedia elements, interactive exercises, and dynamic assessments, enhancing the engagement and interactivity of the learning experience. With the evolution of technology, e-learning has become a prominent and indispensable component of contemporary education, offering a versatile and adaptable means of acquiring knowledge across various subjects and disciplines.

a. COLLABORATIVE FILTERING:

Collaborative filtering algorithms recommend items (this is the filtering part) based on preference information from many users (this is the collaborative part). This approach uses similarity of user preference behaviour, given previous interactions between users and items, recommender algorithms learn to predict future interaction. These recommender systems build a model from a user's past behaviour, such as items purchased previously or ratings given to those items and similar decisions by other users. The idea is that if some people have made similar decisions and purchases in the past, like a movie choice, then there is a high probability they will agree on additional future selections.

II. Literature survey:

Enabling recommendation system architecture in virtualized environment for e-learning E-learning sites are useful for improving the skills and awareness of the academic backbone, such as instructors, students, administrative staff, and those who are searching for current information about various educational institutes. Despite all the benefits of an online learning platform, users face some challenges and complexities, such as selecting appropriate learning material and courses based on their needs and preferences. Hence, the provision of quality resources during the training phases is their central responsibility, the lack of online assistance offered by service providers is known to be the key cause of many difficulties. There is a need to create a system that can intelligently propose courses while considering a variety of viewpoints to enhance the learners' skills and knowledge. This research proposes an architecture that builds semantic recommendations with the aid of virtual agents based on user requirements and preferences, assisting academia in seeking appropriate courses in a real-world setting. The experimental and statistical results show that, when compared with existing techniques, the virtualized agent-based recommendation system not only improved user learning skills but also made course selection easier, depending on users' interests and preferences.**[1]** **An Efficient Approach of Product Recommendation System using NLP Technique** As we are moving toward an age of digital globalization and online shopping, there is an increasing need for an efficient and reliable system that can help the consumers and the visitors to find their suitable products. Currently, various websites display the searched product when a visitor comes to their website. What we need is a system, which can recommend the products which are like the searched products. This will help the consumer to find out another product in case the item is unavailable, or the searched product is not good enough, or when they would like to look through different similar products. A good recommendation system has been found out to be financially beneficial for the companies also. It is found out that consumer is 35% more likely to buy a product if the recommendation is good enough for consumers.**[2]** **An ecommerce recommendation algorithm based on link prediction** In the field of ecommerce, most recommendation algorithms are based on user-item bipartite graph network (BGN). But this kind of recommendation algorithm is severely lacking in accuracy and diversity. In this paper, a novel ecommerce recommendation algorithm is proposed based on BGN link prediction. Firstly, all the user-item data were imported into distance formula to calculate the similarity between the attributes. Then, the BGN was projected into a single-mode network (SMN), making it more efficient to extract potential links from the BGN. On this basis, the potential links were predicted based on similarity. Through experiments on real ecommerce datasets, it was proved that our algorithm has a higher accuracy and coverage than typical recommendation algorithms.[**3]** **Adaptive Personalized Recommendation System Using Learning Automata and Item Clustering** the personalized recommender systems provide user-related services based on user preferences; these preferences are recorded in an individual profile. Therefore, the more complete and precise each user profile leads more successful the recommendation process. The people's interests change over time though traditional researches do not follow these changes regularly. Under such circumstances, designing an efficient user model to track users' interests is greatly important. In the current study, we suggest an algorithm to create the learning automata-based user profiling. Due to many items and the commonality of features between them, we clustered items. In this technique, a learning automaton is assigned to the active user. The learning automaton adjusts the amount of user interest in each cluster based on user feedback. As the user interactions with the system increase, the internal state of the learning automaton converges towards the user's genuine interests in the item clusters. The experimental results demonstrate that our algorithm outperforms compared approaches in precision, recall, RMSE, and MAE. In addition, the proposed algorithm for new users has acceptable performance.[4]

III. *Implementation:*

a. *Data Collection:*

The first step involves sourcing course-related data from Kaggle, a popular platform for open datasets. These datasets may include user interactions, course content, ratings, and preferences. The data is downloaded in structured formats such as CSV or JSON. This information forms the foundation for building an intelligent course recommendation system. It provides insights into user behavior, course popularity, and completion rates. Collecting diverse and rich data ensures the system can generalize well to different learning needs. Proper selection and understanding of the dataset are crucial for model performance, as the quality and relevance of the data directly impact the effectiveness of the recommendation engine.

b. *Pre-processing:*

Pre-processing is essential to clean and structure the raw data collected from Kaggle. This step includes handling missing values, removing duplicates, normalizing text (e.g., lowercasing, removing stop words), and encoding categorical variables. For time-series or sequence-based data, it may involve aligning timestamps or sorting interactions. Textual course descriptions and user reviews may also undergo tokenization and lemmatization for further analysis. Pre-processing ensures that the data is in a consistent and analyzable format, reducing noise and redundancy. It improves model accuracy and ensures efficient training by preparing data inputs suitable for machine learning and deep learning models like LSTM.

c. *Feature Extraction:*

Feature extraction involves identifying key attributes from the pre-processed data that are most relevant to learning preferences and behaviors. For text data, techniques such as TF-IDF, word embeddings (e.g., Word2Vec or GloVe), or contextual embeddings (e.g., BERT) may be used. From user interaction logs, features like course views, ratings, time spent, and sequence patterns are derived. These features capture both the content relevance and user engagement levels, which are crucial for effective recommendations. Extracted features are transformed into numerical vectors that serve as inputs for the LSTM model, helping it learn temporal patterns and contextual relationships in user-course interactions.

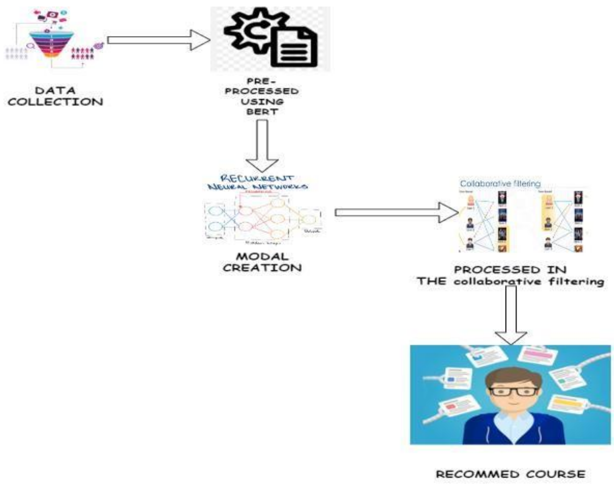
d. Model Creation Using LSTM Algorithm:

Long Short Term Memory (LSTM) networks, a type of recurrent neural network (RNN), are employed to model sequential data such as a user's interaction history. The LSTM processes input sequences (e.g., courses visited over time) and captures long-range dependencies to understand a learner's preferences and evolving interests. The model is trained using features extracted in the previous step and optimized using loss functions like categorical cross-entropy. By learning from historical behavior patterns, the LSTM predicts future preferences with high accuracy. This predictive capability enables the system to anticipate what kind of content a user is likely to engage with next.

e. Recommend Course Using Collaborative Filtering:

Collaborative filtering suggests courses based on user similarity or item similarity, leveraging user ratings and interaction data. It identifies patterns where users with similar tastes preferred certain courses and recommends them to others with similar profiles. There are two main types: user-based and item-based collaborative filtering. When combined with LSTM predictions, collaborative filtering enhances personalization by cross-referencing predicted interests with community trends. This hybrid approach improves recommendation diversity and accuracy, especially in sparse datasets. It helps recommend relevant courses even for users with limited interaction history, ensuring a tailored and engaging e-learning experience.

f. ARCHITECTURE DIAGRAM:



The architecture diagram for the Intelligent E-learning Recommendation System illustrates the high-level structure and components of the system. It provides an overview of how different modules interact and work together to deliver the desired functionality. At the core of the architecture is the Recommendation Engine, which serves as the central component responsible for generating personalized recommendations for users. It utilizes various algorithms, including the LSTM (***Long Short Term Memory***) algorithm and collaborative filtering techniques, to analyze user preferences, historical data, and course information to generate accurate and relevant recommendations. The Data Preprocessing module plays a crucial role in preparing and organizing the data required for recommendation generation. It handles tasks such as data cleaning, normalization, and feature extraction. This module ensures that the data used by the Recommendation Engine is of high quality and suitable for accurate analysis and prediction.

IV. RESULT AND DISCUSSION:

In a LSTM (Long Short-Term Memory) , the architecture consists of several layers, each playing a distinct role in processing sequential data. The primary layers include the input layer, hidden layers, and output layer. Input Layer: The input layer receives sequential data inputs, such as words in a sentence or time-series data points. Each input is typically represented as a vector or embedding, which serves as the starting point for information processing.

$$h_t = \sigma(W_{ih} \cdot x_t + W_{hh} \cdot h_{t-1} + b_h)$$

Where:

- h_t is the hidden state at time step t.
- x_t is the input at time step t.
- W_{ih} is the weight matrix connecting input to hidden layer.
- W_{hh} is the weight matrix connecting hidden state to hidden state.
- b_h is the bias term for the hidden layer.
- σ is the activation function (commonly a sigmoid or tanh function).

a. *Hidden Layers:*

The hidden layers are where the computation occurs in an LSTM . These layers maintain a hidden state that captures information from previous time steps and propagates it to subsequent ones. Each time step involves updating the hidden state based on the current input and the previous hidden state, allowing the network to capture temporal dependencies in the data. In a basic LSTM , there is typically only one hidden layer, but in more complex architectures like LSTM (Long Short-Term Memory) or GRU (Gated Recurrent Unit), multiple hidden layers with specialized gating mechanisms are employed to mitigate the vanishing gradient problem and capture long-term dependencies more effectively.

$$h_t = \text{activation}(W_{ih} \cdot x_t + b_{ih} + W_{hh} \cdot h_{t-1} + b_{hh})$$

- h_t: Hidden state at time t.
- x_t: Input at time t.
- W_{ih}: Input-to-hidden weight matrix.
- b_{ih}: Input-to-hidden bias vector.
- W_{hh}: Hidden-to-hidden weight matrix.
- h_{t-1}: Hidden state at time $t-1$.
- b_{hh}: Hidden-to-hidden bias vector.

b. *Output Layer:*

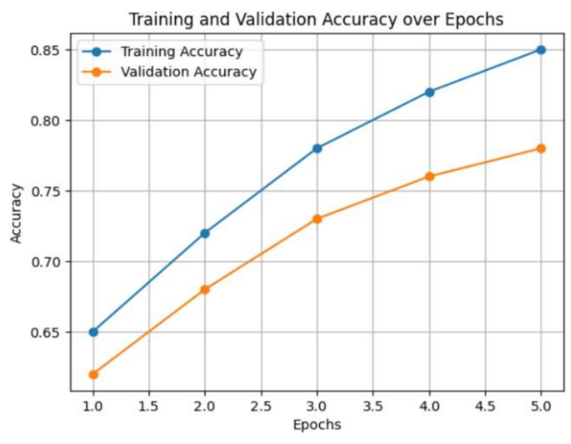
The output layer receives the final hidden state or sequence of hidden states from the hidden layers and produces the network's output. Depending on the task, the output layer may consist of a single unit for binary classification, multiple units for multi-class classification, or a sequence of outputs for sequence prediction tasks.

$$y_t = W_{ho} \cdot h_t + b_{ho}$$

- y_t: Output at time t.
- W_{ho}: Hidden-to-output weight matrix.
- b_{ho}: Output bias vector.

These layers work together to process sequential data, with information flowing from the input layer through the hidden layers to generate an output. The recurrent connections in the hidden layers allow the network to maintain memory of past inputs, enabling it to learn patterns and dependencies in sequential data effectively.

ACCURACY GRAPH:



An accuracy graph illustrates the performance of a machine learning model over training epochs by plotting accuracy against the number of iterations. It provides a visual representation of how the model's accuracy improves or stabilizes during training. Rising accuracy curves signify the model's ability to classify data correctly, while plateaus or declines may indicate issues like overfitting. By monitoring accuracy graphs, researchers can optimize model parameters and architecture to enhance performance. Additionally, accuracy graphs facilitate comparison between different models or configurations, aiding in informed decision-making for model selection and deployment. Overall, accuracy graphs are indispensable tools for evaluating and refining machine learning models.

V. conclusion and future work:

In the proposed model for e-learning, a dynamic recommendation system has been integrated into the platform to enhance the learning experience. Leveraging the capabilities of electronic learning (eLearning), which involves the delivery of educational content through digital resources, this model employs the LSTM (Recurrent Neural Network) algorithm for dynamic prediction and integrates collaborative filtering models for the recommendation process. The LSTM algorithm, known for its ability to capture long-term dependencies in sequential data, is applied to predict user preferences and behavior over time. This predictive capability is then complemented by collaborative filtering models, which analyze user interactions and similarities to provide personalized course recommendations. By incorporating LSTM for dynamic prediction and collaborative filtering for tailored recommendations, the proposed model adapts to individual learning patterns, creating a more responsive and personalized e-learning experience. This approach aligns with the essence of eLearning, allowing users to access educational content seamlessly through digital devices connected to the internet, fostering flexibility and convenience in the learning process.

VI. REFERENCE:

[1] Dahdouh K, Dakkak A, Oughdir L, Ibriz A. Large-scale e-learning recommender system based on Spark and Hadoop. J Big Data 2019;6:2. doi: https://doi.org/ 10.1186/s40537-019-0169-4.

[2] Fu Q-K, Lin C-J, Hwang G-J, Zhang L. Impacts of a mind mapping-based contextual gaming approach on EFL students’ writing performance, learning perceptions and generative uses in an English course. Comput Educ 2019;137:59–77. doi: https://doi.org/10.1016/j.compedu.2019.04.005.

[3] Humayun M. Blockchain-Based secure framework for e-learning during COVID-19. Indian J Sci Technol 2020;13(12):1328–41.

[4] Baloian N, Zurita G. Achieving better usability of software supporting learning activities of large groups. Inf Syst Front 2016;18(1):125–44. doi: https://doi. org/10.1007/s10796-015-9580-3.

[5] Chavarriaga O, Florian-Gaviria B, Solarte O. A recommender system for students based on social knowledge and assessment data of competences. In: Rensing C, de Freitas S, Ley T, Muñoz-Merino PJ, editors. Open learning and teaching in educational communities. Cham: Springer International Publishing; 2014. p. 56–69.

[6] Qiu L, Qi L. E-learning assessment for tourism education LISREL assisted intercultural tourism perception and data integrated satisfaction perspectives. J Comput High Educ 2020;32(1):89–108. doi: https://doi.org/10.1007/s12528- 019-09223-0.

[7] Sarwar S, Qayyum ZU, García-Castro R, Safyan M, Munir RF. Ontology based Elearning framework: A personalized, adaptive and context aware model. Multimed Tools Appl 2019;78(24):34745–71. doi: https://doi.org/10.1007/ s11042-019-08125-8.

[8] Choi C-R, Jeong H-Y. Quality evaluation for multimedia contents of e-learning systems using the ANP approach on high speed network. Multimed Tools Appl 2019;78(20):28853–75. doi: https://doi.org/10.1007/s11042-019-7351-8