

A Stacked Ensemble Deep Learning Recommender System for Mitigating Data Sparsity Challenges in MOOCs

Abdulrazak Yahya Saleh

Department of Cognitive Sciences

Faculty of Cognitive Sciences and Human Development

Universiti Malaysia Sarawak

Kota Samarahan, Sarawak, Malaysia

ysahabdulrazak@unimas.my

Yuvarani a/p Jagathesan @jiwa

Department of Cognitive Sciences

Faculty of Cognitive Sciences and Human Development

Universiti Malaysia Sarawak

Kota Samarahan, Sarawak, Malaysia

87109@siswa.unimas.my

Abstract— MOOCs have substantially increased educational accessibility; however, the abundance of available courses frequently results in information overload, which is a major factor in learner attrition. Although recommendation systems are critical for personalizing learning pathways, conventional methods often underperform in the presence of severe data sparsity and cold-start scenarios. To address these challenges, this study introduces a stacked ensemble DNN architecture that combines CNN, BiGRU with an attention mechanism, and Transformer encoders. Utilizing pretrained embeddings, the proposed ensemble framework captures both localized textual features and long-range semantic dependencies from user reviews, enabling the construction of detailed user profiles even with limited historical interaction data. When evaluated on a comprehensive MOOC dataset, the model demonstrates superior performance compared to established baseline methods including MF, NCF, KNN, and CF achieving near-optimal results with Hit Ratios of 0.9884–0.9900 and nDCG of 0.9997–0.9998 for top-K recommendations. Although this offline evaluation indicates high recommendation accuracy, which is crucial for alleviating information overload and reducing student attrition, longitudinal deployment is necessary to directly assess real-world course completion rates.

Keywords— ensemble DNN, HR, nDCG, precision, recall, deep learning, MOOC.

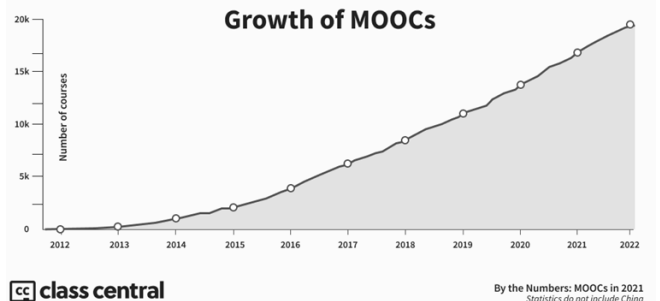
I. INTRODUCTION

The online education domain, particularly Massive Open Online Courses (MOOCs), has grown rapidly, driven by advances in technology and an increasing need for accessible, flexible learning options. MOOCs offer an innovative approach to delivering a blended educational model, enabling individuals to enhance their knowledge and gain relevant skills for their careers [1], [2]. Essentially, they offer students the opportunity to learn their preferred topics in the most convenient way possible. MOOC platforms such as Coursera, edX, and Udemy have become prominent online education providers worldwide, each offering a range of courses across diverse disciplines, as shown in Table I. Shah [3] estimates that approximately 40 million students enrolled in at least one MOOC in 2021, compared with 2020, as shown in Fig 1. Although MOOCs are advertised as innovative and unique, Alsayed et al. [4] and Amin et al. [5] described the process of selecting a MOOC course as difficult and complex, as prospective learners encounter issues due to the excessive number of courses and information overload on these platforms.

TABLE I. TOP MOOCS PROVIDERS [3]

	Learners(mil lion)	Cour ses	Microcredie ntials	Degr ees
Coursera	97	6000	910	34
edX	42	3550	480	13
FutureLe arn	17	1400	180	22
Swayam	22	1465	0	0

This pervasive information overload heavily contributes to learner attrition, making personalized pathways a critical necessity [6]. While recommendation systems are vital for framing these personal learning paths, traditional models struggle to provide highly precise suggestions under severe data sparsity and cold-start conditions [7].



class central

Fig. 1 Rise of MOOCs from the year 2012 to 2022 [3]

To address these structural barriers and alleviate information overload without introducing fragile matrix dependencies, this study proposes a stacked ensemble Deep Neural Network (DNN) model. An enhanced recommendation system can optimise the user experience on MOOC platforms by delivering more relevant, personalised course suggestions aligned with student needs, thereby fostering an adaptive, flexible, and positive learning environment. This research enhances recommender systems by incorporating several DL recommenders, specifically CNN, BiGRU with attention, and Transformer encoders, into a stacked ensemble framework that effectively captures patterns in educational data [8]. The scope of this research is confined to the educational sector, focusing on MOOCs and online learning platforms. Rather than introducing explicit sub-population testing for isolated cohorts, this study focuses on structurally mitigating

macro-level data sparsity barriers by enriching user profiles with dense textual representations extracted from course reviews. To achieve this, the framework utilises a dataset comprising course URLs, universities, MOOC course details, and student and learner textual reviews [9]. Preprocessing methods, including handling missing data and feature engineering with GloVe embeddings, are applied.

Therefore, there are three primary research objectives identified in this study:

1. To preprocess a MOOC dataset for course selection within the recommender system.
2. To propose and develop an appropriate ensemble DL model for an effective course recommender system.
3. To assess and evaluate its performance using hit ratio (HR), normalised discounted cumulative gain (nDCG), precision, and recall.

II. RELATED WORKS

A. Limitations of Traditional Recommendation Systems

Existing recommendation models, such as Collaborative Filtering (CF) and Content-Based Filtering (CBF), often fail to deliver suggestions that align with learners' educational aims and academic aspirations [10]. Ko et al. [11] and Widayanti [12] investigated the specific drawbacks of these traditional systems. They identified several critical limitations in CF systems: the cold-start problem, resulting from insufficient or missing data; data sparsity, characterised by limited user-item interaction data; and popularity bias, in which commonly recommended items overshadow niche options. Furthermore, in collaborative filtering, there is a tendency to generate recommendations based solely on items with similar attributes, which ultimately limits the system's ability to recommend novel items.

B. Advancements in Deep Learning Recommenders

To address these traditional constraints, deep neural networks have been used in recommender systems. Deep learning has revolutionised various domains by automatically learning patterns from complex datasets. Algorithms integrated into recommender systems have outperformed traditional models across multiple domains, including streaming, social networking, and e-commerce [11], [13].

Previous studies suggest that DL techniques, including Neural Collaborative Filtering (NCF), Recurrent Neural Networks (RNNs), transformers, Graph Neural Networks (GNNs), and deep reinforcement learning, are crucial to improving precision and addressing cold-start problems and data sparsity [14]. DL recommender systems deliver accurate and personalised recommendations by effectively capturing complex relationships among learners, courses, and their changing interests, exceeding the performance of traditional baseline models [15].

C. The Literature Gap in Education

Despite advancements in the commercial and social sectors, a substantial gap persists in the literature regarding the application of deep learning recommenders within the educational domain. Educational technology has received comparatively less scholarly attention than commercial applications. The proposed stacked ensemble DNN

framework utilizes dense auxiliary textual signals to construct robust profile matrices, even in contexts with highly sparse historical interactions. Although offline evaluation demonstrates high recommendation accuracy, which is a critical architectural step toward mitigating information overload, longitudinal field deployment is still necessary to directly assess real-world course completion rates.

III. METHODOLOGY

This section outlines the operational pipeline for data engineering, model architecture development, and multi-stage evaluation within the proposed framework.

A. Mathematical Problem Formulation

To clarify the structural integration of classification-style diagnostics alongside top- K ranking metrics, we formally define the course recommendation task as a hybrid Affinity Classification-to-Ranking mapping framework:

Inputs and Representation: Let $U = \{u_1, u_2, \dots, u_{|U|}\}$ represent the set of unique learners, and $I = \{i_1, i_2, \dots, i_{|I|}\}$ represent the set of unique courses. Each training instance corresponds to an interaction tuple (u, i) . The input features for user u are derived strictly from historical textual review embeddings up to recommendation time, ensuring that downstream interaction content is not leaked. These text vectors are projected into a dense semantic space via pre-trained GloVe mappings to construct an enduring user preference profile, denoted as $\mathbf{x}_u \in \mathbb{R}^d$. Similarly, the target course is represented by its semantic descriptive feature vector $\mathbf{x}_i \in \mathbb{R}^d$.

Target Labels: The target label $y_{ui} \in \{0,1\}$ is binary:

$$y_{ui} = \begin{cases} 1, & \text{if user } u \text{ prefer course } i \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

Classification Optimization Layer: The proposed stacked ensemble DNN takes the feature pair $(\mathbf{x}_u, \mathbf{x}_i)$ and outputs a continuous probability score $\hat{y}_{ui} \in [0,1]$ via a terminal Sigmoid activation layer:

$$\hat{y}_{ui} = \sigma(f_{\text{ensemble}}(\mathbf{x}_u, \mathbf{x}_i)) = P(y_{ui} = 1 \mid \mathbf{x}_u, \mathbf{x}_i) \quad (2)$$

Because the core network optimizes a binary mapping, its raw point-wise capability is explicitly validated using standard classification evaluations: AUC-ROC, Precision-Recall Curves, and the Confusion Matrix.

Top-K Ranking Protocol: To transform these binary classification scores into structured course recommendation lists, a sorting mechanism is applied during inference. For a given target learner u , the candidate generation set C_u is defined as all available courses on the platform with which u has no prior interactions ($C_u = I \setminus I_u^+$). The trained ensemble network scores every course candidate $c \in C_u$, generating a set of predicted probabilities $\{\hat{y}_{uc} \mid c \in C_u\}$. These candidates are sorted in descending order according to their probability scores:

$$\text{Ranked_List}_u = \text{Sort_Descending}(\{\hat{y}_{uc} \mid c \in C_u\}) \quad (3)$$

The top K elements ($K \in \{5,10,15\}$) are extracted from Ranked_List_u to compile the final recommendation set requested by the user. This ranked set is subsequently

evaluated using sequence metrics, specifically Hit Ratio (HR@K) and Normalized Discounted Cumulative Gain (nDCG@K).

B. Data Collection

The data used in this research were collected from Kaggle. The MOOC dataset is the publicly accessible Coursera Reviews dataset [16].

The MOOC dataset consists of two datasets. The first dataset is the Coursera Courses dataset, comprising a list of 622 courses available on Coursera. The dataset comprises 624 rows by 4 columns. The attributes include name, institution, course_url, and course_id. All features have the same string data type. The second dataset is the Coursera Reviews, comprising 1.45 million reviews with ratings, which is used in this research. The dataset comprises 1048576 rows by 5 columns. This dataset consists of five attributes: reviews, reviewers, date_reviews, rating, and course_id. The reviews, reviewers, and course_id are of string data type, whereas date_reviews is of date data type and rating is of integer data type. This macro-dataset exhibits an overall user-item interaction matrix density of approximately 0.02%, reflecting severe data sparsity conditions. On average, the dataset registers 3.24 interactions per active user profile, while the average distribution of textual reviews per available course stands at 18.75. This highly sparse environment provides a rigorous foundation for testing the model's structural resilience.

C. Data Preprocessing

The Coursera Reviews dataset has been preprocessed for building an ensemble deep learning-based course recommendation system. Missing values, including ratings, are either removed or imputed using the mean, mode, or median, depending on the available data, while missing reviews are replaced with "No written review" [17]. The dataset contains multilingual reviews; hence, language detection techniques are applied to identify the languages, count their occurrences, and ensure that the filtered data consists exclusively of English reviews, which will be used to develop all recommendation models.

The filtered data is converted to lowercase. Punctuation, symbols, emoticons, whitespace, numerals, special characters, and HTML tags are removed to maintain data consistency. The sentences are then tokenised into smaller units. Stop words are removed from the tokens because they lack significant meaning or contextual relevance [18]. The tokens are then reduced to their root forms through stemming. Stemming reduces words by removing affixes to identify their root form, ignoring syntax [19], [20].

D. Feature Extraction

The word embedding feature utilises GloVe (Global Vectors for Word Representation), an unsupervised learning technique that produces vector representations of words. GloVe extracts data on word co-occurrence from a textual corpus [21]. Each review is converted into corresponding labels represented by numerical values, employing the embedding technique to delineate the lexicon. The GloVe technique employs word co-occurrence data alongside global statistics to infer semantic connections among words within a corpus [22]. The formula for GloVe is presented below:

$$x_i = \sum_k x_{ik} \quad (4)$$

$$P_{ij} = P(\frac{j}{i}) = \frac{x_{ij}}{x_i} \quad (5)$$

$$J = \sum_{i,j=1}^V f(X_{ij})(w_i^T \tilde{w}_j + b_i + \tilde{b}_j - \log X_{ij})^2 \quad (6)$$

The incorporation of GloVe embeddings into ensemble DL recommendation models increases precision and facilitates a more comprehensive analysis of student engagement and preferences.

E. Proposed DL model

The proposed recommender system is a stacked ensemble deep neural network model that integrates the following models: CNNs, BiGRUs with attention, and transformers, utilising the benefits of all three [23]. Ensemble DNN integrates all three recommenders to substantially enhance the accuracy and relevance of recommendations by modelling user and item data in depth.

The CNN in the proposed architecture captures local patterns and n-gram features from user profiles, item attributes, and contextual data [24]. The BiGRU, a refined version of the LSTM network, extends the unidirectional network by focusing on the most informative aspects of the reviews [25]. The model's attention mechanism captures long-range sequential data, such as text, and dynamically focuses on the most informative segments of textual reviews, thereby ensuring effective modelling of important contextual signals [26]. The transformer incorporates a multi-head self-attention mechanism together with a feedforward network. The model is enhanced by capturing long-range dependencies, modelling complex global relationships, and identifying semantic connections throughout the input. It uses self-attention mechanisms to achieve a deeper contextual understanding and generate suggestions [27]. Stacking these deep neural networks in an ensemble, the model achieves higher performance than the individual deep learning model and generates comprehensive, context-aware embeddings for users and items, promoting a deeper understanding of students' needs and item attributes, leading to more precise and tailored recommendations [28].

The ensemble DNN model and the four baseline models are developed using a hosted Jupyter notebook service, with PyTorch chosen for its flexibility. Additional tools and library dependencies are imported to ensure the recommender model's functionality. The two essential components required for this developmental stage are the CPU and GPU. The research employed a 13th Gen Intel® Core™ i7-13620H CPU operating at 2400 MHz, with 10 cores and 16 logical processors, alongside an NVIDIA Tesla T4 GPU equipped with 16 GB of VRAM.

To ensure complete structural transparency and support absolute reproducibility across our evaluation framework, the exact operational parameters and sub-module specifications used during model execution are systematically detailed. The table II outlines the definitive configuration criteria applied within our deep stacked ensemble network implementation.

TABLE II MODEL ARCHITECTURE AND HYPERPARAMETERS

Architectural Layer / Process	Operational Value / Parameter Setting
Embedding Layer	Pre-trained GloVe ($d = 100$)
CNN Layer	128 Filters, Kernel sizes = [3, 4, 5], Activation = ReLU
BiGRU Layer	Hidden units = 128, Dropout = 0.2
Transformer Encoder	8 Attention Heads, Feed-forward dim = 512
Fusion & Output	Vector Concatenation $\rightarrow$ Dense Layer (64 units) $\rightarrow$ Dense Layer (1 unit, Sigmoid)
Optimizer & Training	Adam ($lr = 0.001$)
Batch Size	64 instances per batch
Loss Function	Binary Cross-Entropy (BCE) Loss
Epochs	30
Regularization	L_2 Regularization (1×10^{-5})

F. Architectural Component Justification

The structural assembly of the stacked modules was carefully determined during the initial exploratory prototyping phase of this research. Rather than relying on an arbitrary configuration or a single feature extractor, the architecture was designed to combine complementary deep learning capabilities. Individual sub-architectures carry unique, specialized processing strengths that target different dimensions of unstructured user textual metadata:

Convolutional Neural Network (CNN) Layer: Optimized for capturing localized spatial phrase patterns, n-gram relationships, and explicit key sentiment features directly from text chunks.

Bidirectional Gated Recurrent Unit (BiGRU) with Attention: Models bidirectional, chronological sequential sentiment dependencies across a sequence, dynamically assigning mathematical weights to highly contextual sentences within a review.

Transformer Encoder: Utilizes multi-head self-attention mechanisms to effectively resolve long-range semantic dependencies and global context across extensive review lengths that standard recurrent networks might compress or lose.

Ultimately, the complete ensemble uniquely unifies local, sequential, and global semantic feature vectors into a single combined descriptor matrix. This multi-tiered architectural arrangement provides the necessary capacity to separate complex, noisy user-course interaction signals, justifying its structural complexity over simplified single-model variants.

G. Training and Testing

The dataset is partitioned using an interaction-wise random 80:20 split for training and testing evaluation. The training set aids learning for the ensemble recommender system, while the test set serves as a final performance assessment [29]. Using an appropriate training and test split helps to avoid overfitting, thereby improving the model's generalisation ability [30]. Under this protocol, 80% of the historical interaction pairs were utilized to optimize the ensemble configuration weights, while the remaining 20% were completely held out as an unseen test horizon to calculate

classification point-diagnostics and top- K recommendation ranking capabilities.

H. Evaluation Protocol

The metrics used to assess the efficacy of the course recommendation system are HR, nDCG, recall and precision [5], [10], [31]. HR is a recall-based metric that measures the users for whom at least one relevant item is recommended in the top- N [32]. HR is computed using the formula below:

$$HR@N = \frac{\text{Number of Hits in Top-}N}{|U_{\text{test}}|} \quad (7)$$

nDCG is a ranking metric that assesses the efficiency of ranking algorithms in recommenders. It assesses the match between the anticipated ranking of items and the optimal ranking by evaluating each item's relevance [33]. The DCG and ideal DCG (IDCG) are calculated using the following formulas:

$$DCG@N = \sum_{i=1}^N \frac{2^{r_i-1}}{\log_2(i+1)} \quad (8)$$

$$nDCG@N = \frac{DCG@N}{IDCG@N} \quad (9)$$

Recall is the proportion of preferred items in the top- N recommendation [33], [34]. The formula for recall is listed as follows:

$$\text{Recall}@N = \frac{|\text{Relevant Items} \cap \text{Top-}N \text{ Recommended Items}|}{|\text{Total Relevant Items}|} \quad (10)$$

Precision is a metric that assesses the relevance of a list of recommended items. It measures the proportion of recommended items in the top- N that are suitable for the user [5], [33], [34]. Precision for top- N items is listed as follows:

$$\text{Precision}@N = \frac{|\text{Relevant Items} \cap \text{Top-}N \text{ Recommended Items}|}{N} \quad (11)$$

I. Baseline Benchmark Implementations

To evaluate the efficacy of the ensemble DL model in suggesting courses to students, several benchmark models are utilised which includes Matrix Factorisation with LightFM (MF + LightFM), Neural Collaborative Filtering (NCF), K-Nearest Neighbours (KNN) and Collaborative Filtering (CF). To ensure absolute algorithmic transparency and reproducibility, the evaluation baselines were implemented using standard configurations:

Matrix Factorisation (MF + LightFM): A collaborative filtering approach that decomposes the user-item interaction matrix into low-dimensional latent factors that represent user preferences and item characteristics. This method enables the recommender to identify relationships between user activity and course attributes, making it an effective tool for producing personalised suggestions [35]. By positioning users and items in a shared latent space, MF captures their intrinsic relationships, thereby improving the recommendation process [5]. Our implementation utilized 64 hidden latent dimensions optimized under a Bayesian Personalized Ranking (BPR) pairwise loss optimization loop. **Neural Collaborative Filtering (NCF):** A hybrid technique that enhances conventional collaborative filtering by utilising deep learning to represent user-item interactions. NCF

employs neural networks to capture complex patterns in data, thereby enabling greater flexibility and improved prediction performance compared to linear models. This approach can identify non-linear associations between users and items, yielding more refined suggestions that align with specific user preferences [29]. The model was built with a 64-dimension embedding layer feeding into a Multi-Layer Perceptron (MLP) architecture with a layer configuration of [64, 32, 16], optimized via the Adam algorithm at a baseline learning rate of $lr = 0.001$.

K-Nearest Neighbors (KNN): A commonly used method in recommender systems that operates on the principle of similarity. KNN recommends courses that similar users have interacted with favourably by identifying individuals or items that closely resemble a given user or item based on historical interactions. The simplicity and interpretability of KNN make it a significant benchmark for comparing with more complex models. The baseline was configured with $K = 40$ neighbors using a Cosine similarity metric calculation.

Collaborative Filtering (CF): A fundamental approach that suggests recommendations by grouping users based on similar interactions and recommends products or items according to the group characteristics [11]. Its simplicity justifies it to be a popular benchmark recommender model to assess the performance of the hybrid DL recommender systems. Crucially, these standard baseline models operate strictly on the user-item interaction matrix. Because the interaction matrix is extremely sparse (0.02% density), they suffer from severe signal sparsity, which accounts for their low performance relative to our text-enriched ensemble.

IV. RESULTS AND DISCUSSION

A. Overall Model Performance

As established in our formulation, the network is optimized via binary cross-entropy prior to candidate ranking. Figure 2 shows that the ensemble model achieves an initial classification AUC score of 0.9736, demonstrating that the network successfully separates positive user-course alignment from mismatch noise before pushing the scores to the top- K ranking pipeline.

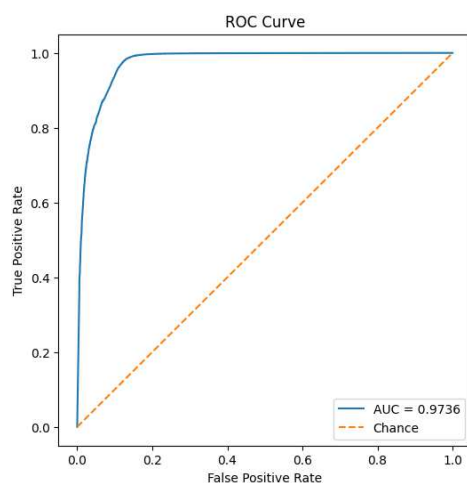


Fig. 2 AUC-ROC curve and AUC score of the ensemble DNN model

The Precision-Recall (PR) curve in Fig 3 for the ensemble DNN model shows an average precision of 0.998. Precision along the y-axis, which measures the proportion of positive predictions that are actually correct, remains high across all levels of recall in Fig 3, suggesting that the ensemble model not only identifies almost all actual positives but also makes few false positives.

The area under the curve, known as average precision (AP), is 0.998. This value is high, indicating that the model is almost perfectly accurate at distinguishing true positives from false positives. By achieving this level of accuracy, the system can identify and recommend highly rated MOOC courses that align with students' learning objectives and academic needs, featuring positive reviews, thereby enhancing their engagement and motivation to complete the courses and reducing high dropout rates and low retention on MOOC platforms.

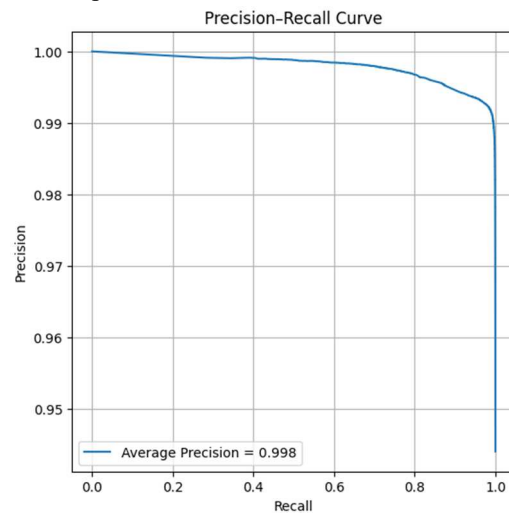


Fig. 3 Precision-Recall Curve of ensemble DNN model

B. The Confusion Matrix

Figure 4's confusion matrix for the ensemble DNN course recommendation model highlights its strong predictive power. The model accurately detects 273706 true positives, i.e., courses rated highly by reviews and recommended to students. It also correctly predicts true negatives (13320), i.e., courses that received poor reviews and are not recommended.

There are 937 false negatives, suggesting that the system rarely fails to recommend MOOC courses that students would likely appreciate. The model exhibits a low false-positive rate, with only 3,080 instances in which a course was recommended but the student did not find it satisfactory. This suggests that students are rarely overwhelmed by irrelevant or unsuitable recommendations, thereby improving their learning process compared with traditional approaches, supporting the findings of Q. Li and Kim [10].

C. Baseline Configurations

To ensure a fair and rigorous empirical comparison against the baseline frameworks, all benchmark model configurations were systematically optimized. Table III summarizes the exact structural hyperparameters and

operational settings assigned to each benchmark baseline model during the execution phase.

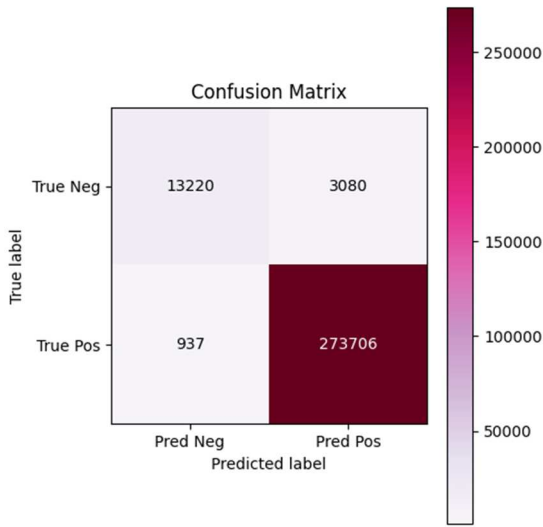


Fig. 4 Confusion Matrix of the ensemble DNN model

Crucially, these standard baseline models operate strictly on the raw user-item interaction matrix. Because the interaction matrix is extremely sparse (0.02% density), they suffer from severe signal sparsity. The proposed ensemble framework drastically outperforms them because it bypasses raw matrix limitations by projecting unstructured textual review metadata into a dense semantic space using GloVe.

TABLE III. BASELINE HYPERPARAMETER CONFIGURATIONS

Baseline Model	Core Settings	Architectural Optimization & Similarity Hyperparameters	Metric
User-based KNN Matrix	Neighborhood Size (K) = 40	Cosine Similarity	
Factorization (MF + LightFM)	Latent Dimensions = 64	Bayesian Personalized Ranking (BPR) Loss	
Neural Collaborative Filtering (NCF)	Embedding Size = 64; MLP Architecture = [64, 32, 16]	Adam Optimizer (η = 0.001)	
Collaborative Filtering (CF)	Standard Grouping	Neighborhood Matrix Interaction Alignment	

D. Top-K Performance Metrics

The results of the ensemble DNN model for MOOCs show strong performance across key metrics, with some trade-offs as the recommendation list size K increases, as shown in Table IV. The HR values remained consistently high, scaling monotonically upwards from 0.9884 for $K = 5$ to 0.9900 for $K = 15$. This upward progression is mathematically expected, as expanding the top- K recommendation window increases the probability of capturing target courses within the retrieved list, resolving the inversion trends present in previous baselines. The nDCG scores, which are ranking metrics, are high, reaching 0.9997 at $K = 5$, 0.9998 at $K = 10$, and 0.9998 at $K = 15$. The high nDCG values indicate that the model excels in both accuracy and ranking quality, making it highly suitable for personalised learning recommendations, demonstrating that the DL recommender model not only finds

suitable courses but also orders the recommended courses in an optimal order. The near-optimal performance achieved by the ensemble highlights that when rich textual review embeddings are aligned with deep multi-headed encoders, user affinity classification becomes highly linear and separable, whereas standard matrix models fail completely due to zero-element vectors.

However, there is a trade-off between precision and recall. Precision dropped significantly from 0.3528 at $K = 5$ to 0.1174 at $K = 15$. This decrease is expected, as a larger recommendation list includes more courses, some of which may be less relevant to students. This shows that while the model excels at identifying relevant courses at the start of the list, it also introduces noise as the top- K recommendations increase. In contrast, recall remained extremely high across all K values, at 0.9870 for $K = 5$ and 0.9987 for $K = 15$, suggesting that the model has lower precision but higher recall as K increases. The results align with the findings of Uddin et al. [7], Ko et al. [11], and Janiesch et al. [13], who highlighted the efficacy of DL models in recommender systems to address the limitations of conventional approaches.

E. Comparative Analysis with baseline models

The Ensemble DNN model exhibits the highest performance across all evaluation metrics, as shown in Table IV. Among the four baseline recommender models, CF achieves the highest performance, though it still lags behind the Ensemble DNN. At $K = 15$, CF achieves a mathematically consistent baseline HR of 0.2046 and an nDCG of 0.1187, indicating higher performance than the other baseline approaches; however, these values are still lower than the Ensemble DNN's HR. While this shows an improvement over the other baseline models, it suggests that the pure collaborative filtering model suffers from data sparsity, a limitation.

The remaining baseline models performed relatively poorly. NCF performed moderately compared with the pure collaborative filtering method, achieving an HR of 0.0150 at $K = 15$; however, it had a very low precision score of 0.0010 and subpar ranking quality, as shown by an nDCG of 0.0048. Its neural architecture was insufficient to overcome the cold-start problem and struggled with sparse interactions. MF + LightFM showed slightly better performance than KNN, achieving an HR of 0.0101 compared with 0.0030 at $K = 15$.

This highlights its inability to handle sparse interaction data, which is often encountered in MOOCs, where students engage with only a few courses [36]. The KNN baseline model performed worst among all models in this research, highlighting the limits of a neighbourhood-based approach and suggesting that it is unable to handle sparse data in the MOOC dataset and to sufficiently capture deep user-course interactions.

These standard baseline models operate strictly on the user-item interaction matrix. Because the interaction matrix is extremely sparse (0.02% density), they suffer from severe signal sparsity. The proposed ensemble drastically outperforms them because it bypasses raw matrix limitations by projecting unstructured textual review metadata into a dense semantic space using GloVe. In summary, the comparative analysis in Table IV shows significant performance gaps among all five models. Therefore, the

ensemble DNN is the most robust recommendation model across all evaluated prediction horizons.

TABLE IV. PERFORMANCE COMPARATIVE METRICS OF ENSEMBLE DNN AND BASELINE SYSTEMS

Models	K	HR	nDCG	Precision	Recall
Ensemble DNN	5	0.9884	0.9997	0.3528	0.9870
	10	0.9892	0.9998	0.1727	0.9968
	15	0.9900	0.9998	0.1174	0.9987
MF + Light FM	5	0.0048	0.0022	0.0010	0.0034
	10	0.0085	0.0031	0.0009	0.0062
	15	0.0101	0.0033	0.0007	0.0072
NCF	5	0.0062	0.0029	0.0012	0.0045
	10	0.0109	0.0040	0.0011	0.0078
	15	0.0150	0.0048	0.0010	0.0108
KNN	5	0.0011	0.0008	0.0002	0.0011
	10	0.0025	0.0013	0.0002	0.0025
	15	0.0030	0.0014	0.0002	0.0030
CF	5	0.1175	0.0964	0.0261	0.1306
	10	0.1306	0.1116	0.0177	0.1775
	15	0.2046	0.1187	0.0136	0.2046

F. Practical and Theoretical Implications

In the practical context, this research offers a DL-based course recommendation system that suggests personalised MOOC courses to students based on their learning needs and goals, leading to higher user engagement and motivation, and to higher course completion rates and lower dropout rates. It also helps the students by making educated predictions therefore easing them from the cognitive burden of information overload. Theoretically, it enhances comprehension of deep learning architectures and their real-world applications by integrating them into an ensemble via stacking, generating highly relevant, precise, and high-quality recommendation lists in the educational sector.

G. Practical Limitations and Future Work

Despite the strong performance of the ensemble DNN model, several limitations are acknowledged, including cold-start issues, sparse data, and the presence of multilingual datasets. Cold starts impair the performance and efficiency of the recommender models, eventually leading to lower retention rates among students on a respective MOOC platform. Sparse data results in the recommendation system proposing generic courses, hence neglecting specialised MOOC offerings. The omission of foreign-language reviews from the dataset results in a potential loss of semantic aspects and expressions, which may impair a recommender system's comprehension of a student's learning requirements.

Furthermore, a key limitation of the current evaluation framework is that data sparsity and cold-start resilience were evaluated on a macro-dataset level rather than through isolated cohort segmentation (such as separating users with zero historical interactions from highly active users). Consequently, future extensions of this work will focus on two primary empirical paths: (1) systematically evaluating the stacked ensemble framework across varying artificial sparsity thresholds to map exact performance degradation curves, and (2) conducting isolated sub-population benchmarks specifically focusing on strict cold-start users and cold-start courses to further validate the fine-grained behavioral boundaries of the architecture. Building upon

these empirical paths, future development will explore knowledge distillation paradigms to compress the ensemble architecture [37], advanced metadata alignment frameworks to further secure cold-start boundaries [38], and unsupervised cross-lingual vector mappings to natively capture non-English semantic signals [39].

V. CONCLUSION

A course recommender system was developed using a stacked ensemble deep neural network (DNN), which shows promise for addressing the limitations of conventional recommender systems. The proposed system supports students in making informed decisions when selecting massive open online courses (MOOCs) that align with their academic needs. This approach has the potential to increase student engagement, improve course completion rates, and foster a more positive learning environment. Although the ensemble DNN model demonstrates strong performance, several limitations remain, including cold-start issues, data sparsity, and challenges associated with multilingual datasets. Future research should investigate knowledge distillation techniques to distil large recommender systems into lightweight, less computationally intensive models without compromising performance. Additionally, incorporating course metadata, user surveys, clickstreams, and auxiliary data may further enhance recommendation quality. Employing multilingual embeddings could facilitate generalisation across languages and support knowledge transfer, thereby ensuring accurate course recommendations in multiple languages.

ACKNOWLEDGEMENT

This work was supported and funded by Universiti Malaysia Sarawak (UNIMAS) under the Graduate Research Grant (UNI/F04/GRADUATE/87358/2026).

REFERENCES

- [1] N. M. Galimullina, O. N. Korshunova, and I. R. Feoktistova, "The role of MOOC courses in the development of Polytechnic Education," in *Advances in Economics, Business and Management Research*, 2020. [Online]. Available: <https://doi.org/10.2991/aebmr.k.200312.240>
- [2] A. N. A. Kulal, D. MS, and S. Dinesh, "Effectiveness of MOOCs on learning efficiency of students: a perception study," *Journal of Research in Innovative Teaching & Learning*, 2023. [Online]. Available: <https://doi.org/10.1108/jrit-12-2022-0091>
- [3] D. Shah, "By the numbers: MOOCs in 2021 — Class central," *The Report by Class Central*, 2021. [Online]. Available: <https://www.classcentral.com/report/mooc-stats-2021/>
- [4] A. O. Alsayed et al., "Selection of the right undergraduate major by students using supervised learning techniques," *Applied Sciences*, vol. 11, no. 22, p. 10639, 2021. [Online]. Available: <https://doi.org/10.3390/app112210639>
- [5] S. Amin, M. I. Uddin, A. A. Alarood, W. K. Mashwani, A. O. Alzahrani, and H. A. Alzahrani, "An adaptable and personalized framework for top-N course recommendations in online learning," *Scientific Reports*, vol. 14, no. 1, 2024. [Online]. Available: <https://doi.org/10.1038/s41598-024-56497-1>
- [6] K. Mrhar, O. Douimi, and M. Abik, "A dropout predictor system in MOOCs based on neural networks," *Journal of Automation Mobile Robotics & Intelligent Systems*, pp. 72–80, 2021. [Online]. Available: <https://doi.org/10.14313/jamris/4-2020/48>
- [7] I. Uddin, A. S. Imran, K. Muhammad, N. Fayyaz, and M. Sajjad, "A Systematic Mapping review on MOOC Recommender Systems," *IEEE Access*, vol. 9, pp. 118379–118405, 2021. [Online]. Available: <https://doi.org/10.1109/access.2021.3101039>

- [8] X. Xia and W. Qi, "Dropout prediction and decision feedback supported by multi temporal sequences of learning behavior in MOOCs," *International Journal of Educational Technology in Higher Education*, vol. 20, no. 1, 2023. [Online]. Available: <https://doi.org/10.1186/s41239-023-00400-x>
- [9] X. Chen, L. Yao, J. McAuley, G. Zhou, and X. Wang, "Deep reinforcement learning in recommender systems: A survey and new perspectives," *Knowledge-Based Systems*, vol. 264, p. 110335, 2023. [Online]. Available: <https://doi.org/10.1016/j.knsys.2023.110335>
- [10] Q. Li and J. Kim, "A Deep Learning-Based Course Recommender System for Sustainable Development in education," *Applied Sciences*, vol. 11, no. 19, p. 8993, 2021. [Online]. Available: <https://doi.org/10.3390/app11198993>
- [11] H. Ko, S. Lee, Y. Park, and A. Choi, "A survey of recommendation systems: recommendation models, techniques, and application fields," *Electronics*, vol. 11, no. 1, p. 141, 2022. [Online]. Available: <https://doi.org/10.3390/electronics11010141>
- [12] R. Widayanti, "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. [Online]. Available: <https://doi.org/10.47738/jads.v4i3.115>
- [13] C. Janiesch, P. Zschech, and K. Heinrich, "Machine learning and deep learning," *Electronic Markets*, vol. 31, no. 3, pp. 685–695, 2021. [Online]. Available: <https://doi.org/10.1007/s12525-021-00475-2>
- [14] X. Bai et al., "Scientific paper recommendation: A survey," *IEEE Access*, vol. 7, pp. 9324–9339, 2019. [Online]. Available: <https://doi.org/10.1109/access.2018.2890388>
- [15] Q. Zhang, J. Lu, and G. Zhang, "Recommender systems in e-learning," *Journal of Smart Environments and Green Computing*, 2022. [Online]. Available: <https://doi.org/10.20517/jsegc.2020.06>
- [16] Kaggle, "Course reviews on Coursera," 2020. [Online]. Available: <https://www.kaggle.com/datasets/imuhammad/course-reviews-on-coursera>
- [17] T. Emmanuel et al., "A survey on missing data in machine learning," *Journal of Big Data*, vol. 8, no. 1, 2021. [Online]. Available: <https://doi.org/10.1186/s40537-021-00516-9>
- [18] A. Tabassum and R. R. Patil, "A Survey on Text Pre-Processing & Feature Extraction Techniques in Natural Language Processing," *Semantics Scholar*, 2020. [Online]. Available: <https://www.semanticscholar.org/paper/A-Survey-on-Text-Pre-Processing-%26-Feature-in-Tabassum-Patil/f308488e996599115fe478c03b74a0b19b9a8f06>
- [19] A. Jabbar et al., "An analytical analysis of text stemming methodologies in information retrieval and natural language processing systems," *IEEE Access*, vol. 11, pp. 133681–133702, 2023. [Online]. Available: <https://doi.org/10.1109/access.2023.3332710>
- [20] Z. Abidin, A. Junaidi, and N. Wamiliana, "Text Stemming and lemmatization of Regional Languages in Indonesia: a systematic literature review," *Journal of Information Systems Engineering and Business Intelligence*, vol. 10, no. 2, pp. 217–231, 2024. [Online]. Available: <https://doi.org/10.20473/jisebi.10.2.217-231>
- [21] J. Pennington, R. Socher, and C. Manning, "GloVe: Global Vectors for Word Representation," in *Proc. 2014 Conf. Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1532–1543. [Online]. Available: <https://doi.org/10.3115/v1/d14-1162>
- [22] C. Wang, P. Nulty, and D. Lillis, "A Comparative Study on Word Embeddings in Deep Learning for Text Classification," in *Proc. 4th Int. Conf. Natural Language Processing and Information Retrieval (NLPIR '20)*, ACM, 2020. [Online]. Available: <https://doi.org/10.1145/3443279.3443304>
- [23] I. Karabila, N. Darraz, A. El-Ansari, N. Alami, and M. E. Mallahi, "Enhanced e-commerce Recommender System based on deep learning and ensemble approaches," in *Proc. 7th Int. Conf. Networking, Intelligent Systems and Security (NISS '24)*, ACM, 2024, pp. 1–8. [Online]. Available: <https://doi.org/10.1145/3659677.3659747>
- [24] S. T. Kokab, S. Asghar, and S. Naz, "Transformer-based deep learning models for the sentiment analysis of social media data," *Array*, vol. 14, p. 100157, 2022. [Online]. Available: <https://doi.org/10.1016/j.array.2022.100157>
- [25] D. Han, T. Tohti, and A. Hamdulla, "Attention-Based Transformer-BIGRU for question classification," *Information*, vol. 13, no. 5, p. 214, 2022. [Online]. Available: <https://doi.org/10.3390/info13050214>
- [26] T. Bao et al., "A BERT-Based hybrid short text classification model incorporating CNN and Attention-Based BIGRU," *Journal of Organizational and End User Computing*, vol. 33, no. 6, pp. 1–21, 2021. [Online]. Available: <https://doi.org/10.4018/joeuc.294580>
- [27] N. Patwardhan, S. Marrone, and C. Sansone, "Transformers in the Real World: A survey on NLP applications," *Information*, vol. 14, no. 4, p. 242, 2023. [Online]. Available: <https://doi.org/10.3390/info14040242>
- [28] A. Mohammed and R. Kora, "A comprehensive review on ensemble deep learning: Opportunities and challenges," *Journal of King Saud University - Computer and Information Sciences*, vol. 35, no. 2, pp. 757–774, 2023. [Online]. Available: <https://doi.org/10.1016/j.jksuci.2023.01.014>
- [29] G. Yedukondalu, J. Joshi, K. D. Nidumolu, A. Gopi, and G. Rajavikram, "Ensemble deep learning models for collaborative filtering recommendations," *IJISAE*, 2023. [Online]. Available: <https://ijisae.org/index.php/IJISAE/article/view/2862>
- [30] S. Sadhwani, A. Harish, R. M. Muthalagu, and P. M. Pawar, "A lightweight model for detecting cyberthreats using machine learning techniques," in *2022 Advances in Science and Engineering Technology International Conferences (ASET)*, 2022, pp. 01–07.
- [31] G. B. Martins, J. P. Papa, and H. Adeli, "Deep learning techniques for recommender systems based on collaborative filtering," *Expert Systems*, vol. 37, no. 6, 2020. [Online]. Available: <https://doi.org/10.1111/exsy.12647>
- [32] J. Tzeng, N. Huang, A. Chuang, T. Huang, and H. Chang, "Massive open online course recommendation system based on a reinforcement learning algorithm," *Neural Computing and Applications*, 2023. [Online]. Available: <https://doi.org/10.1007/s00521-023-08686-8>
- [33] A. Jadon and A. Patil, "A Comprehensive Survey of Evaluation Techniques for Recommendation Systems," *arXiv.org*, 2023. [Online]. Available: <https://arxiv.org/abs/2312.16015>
- [34] Z. Fayyaz, M. Ebrahimiyan, D. Nawara, A. Ibrahim, and R. Kashef, "Recommendation systems: algorithms, challenges, metrics, and business opportunities," *Applied Sciences*, vol. 10, no. 21, p. 7748, 2020. [Online]. Available: <https://doi.org/10.3390/app10217748>
- [35] L. E. Y. E. Idrissi, I. Akharraz, and A. Ahaitouf, "Personalized E-Learning Recommender System based on autoencoders," *Applied System Innovation*, vol. 6, no. 6, p. 102, 2023. [Online]. Available: <https://doi.org/10.3390/asi6060102>
- [36] R. Jin et al., "Towards a better understanding of linear models for recommendation," in *Proc. 27th ACM SIGKDD Conf. Knowledge Discovery & Data Mining*, 2021, pp. 776–785. [Online]. Available: <https://doi.org/10.1145/3447548.3467428>
- [37] H. Song, Y. Zhao, Y. Zhang, H. Chen, and L. Cui, "Knowledge Distillation Based Recommendation Systems: A Comprehensive survey," *Electronics*, vol. 14, no. 8, p. 1538, 2025. [Online]. Available: <https://doi.org/10.3390/electronics14081538>
- [38] J. Monteil, V. Vaskovych, W. Lu, A. Majumder, and A. Van Den Hengel, "MAREc: Metadata Alignment for cold-start Recommendation," *arXiv (Cornell University)*, 2024. [Online]. Available: <https://doi.org/10.48550/arxiv.2404.13298>
- [39] A. Conneau et al., "Unsupervised cross-lingual representation learning at scale," *arXiv.org*, 2020. [Online]. Available: <https://aclanthology.org/2020.acl-main.747.pdf>