République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Chadli Bendjedid - El Tarf Département
d'Informatique
Faculté des Sciences et Technologies
Laboratoire d'informatique et de mathématiques appliquées (LIMA)

THÈSE

Pour l'obtention du diplôme de docteur 3`eme cycle

Une Approche Intelligente pour l'apprentissage distant

Filière : Informatique
Spécialité : Systèmes d'Information et E-learning
Préparé par :

SOUNDES OUMAIMA BOUFAIDA

Soutenue le : **30/06/2026**
Devant le jury composé de :

Présidente:	Dr.	Amina Makhlouf	Université Chadli Bendjedid - El Tarf
Rapporteur:	Prof.	Abdelmajid Benmachiche	Université Chadli Bendjedid - El Tarf
Co-Rapporteur:	Dr.	Majda Maatallah	Université Chadli Bendjedid - El Tarf
Examinateur:	Dr.	Zoubir Talai	Badji Mokhtar University - Annaba
Examinateur:	Dr.	Ala Djeddai	Université Chadli Bendjedid - El Tarf

Academic year :2025/2026

Democratic and Popular Republic of Algeria

Ministry of Higher Education and Scientific Research

University of Chadli Bendjedid

Faculty of Sciences Technology

Department of Computer Science

Laboratory of Computer Science and Applied Mathematics (LIMA)

THESIS

For the award of the Doctoral Degree (Third Cycle)

A Smart Approach to Distance Learning

Field: Computer Science

Specialization: Information Systems and E-learning

Prepared by:

SOUNDES OUMAIMA BOUFAIDA

Defended on: 30/05/2026

Before the jury composed of:

President:	*Dr.*	*Amina Makhlouf*	*Chadli Bendjedid University - El Tarf*
Supervisor:	*Prof.*	*Abdelmajid Benmachiche*	*Chadli Bendjedid University - El Tarf*
Co-Supervisor:	*Dr.*	*Majda Maatallah*	*Chadli Bendjedid University - El Tarf*
Examiner:	*Dr.*	*Zoubir Talai*	*Badji Mokhtar University - Annaba*
Examiner:	*Dr.*	*Ala Djeddai*	*Chadli Bendjedid University - El Tarf*

Academic year :2025/2026

Dedication

I dedicate this work:

To my grandparents

I wish to pay heartfelt tribute to my grandparents, whose values, kindness, and presence have always been a source of strength and inspiration.

*I especially dedicate this work to the memory of my **grandfather**, who recently passed away. His encouragement and pride in my academic journey particularly before and after I succeeded in the doctoral entrance examination will remain forever etched in my heart. His unwavering belief in me continues to light my way.*

To my father

For his deep love, endless patience, strong sense of duty, and the countless sacrifices he has made to ensure my success.

To my mother

For her sincere affection, constant support, gentle care, and silent prayers that have guided and protected me throughout every stage of this journey.

To my beloved sisters

For their love, unconditional support, and comforting words that have lifted me during moments of doubt and difficulty.

To my uncles and aunts

For their warm words, their faith in me, and their encouragement which have accompanied me from the beginning.

To all those who love me and whom I love,

And to everyone who, from near or far, has contributed in any way to the realization of this work.

Acknowledgements

I would like to begin by expressing my deepest gratitude to God, whose guidance, strength, and blessings have accompanied me throughout every stage of my academic journey. Without His divine support, reaching this important milestone would not have been possible.

I am profoundly thankful to my thesis supervisor, ***Pr. Abdelmadjid Benmachiche*,** for his exceptional guidance, constructive feedback, and steadfast support. His expertise, mentorship, and encouragement have been pivotal in shaping both the direction and depth of this research.

I am deeply grateful to my thesis co-supervisor, *Dr. Majda Maatallah*, for her invaluable guidance and insightful mentorship throughout the course of this research.

To my beloved ***family***, especially my ***mother***, my ***father***, my ***sisters*** thank you for your unwavering love, patience, and belief in me. Your constant encouragement and emotional support have been the foundation upon which I've built my perseverance and motivation.

I would also like to express my heartfelt appreciation to my close friends and my research team (*Rais Khadidja, Boutabia Ines and Sedraoui Brahim khalil*), my friends whose presence, advice, and encouragement have made this journey not only possible but also meaningful and fulfilling. Your companionship has been a true source of strength, both academically and personally. To everyone who has supported me in any way during this journey, thank you for being part of it and for helping me turn this vision into reality.

Abstract

This research provides one cohesive and comprehensive deep learning framework adaptable to many of the technical intelligence and adaptability enhancements required by Massive Open Online Course (MOOC) platforms across four key aspects, classification, personalized course recommendation, learner behaviour modeling and semantic content retrieval. Given the exponential growth and diversity of MOOCs and shortcomings of traditional systems' limitations to pedagogical relevance and learner context, we present new architectures that incorporate neural networks with an attention mechanism and bio-inspired optimization. My first contribution is CNN-LOA, a convolutional neural network optimized with the Lion Optimization Algorithm that makes dynamic hyperparameter adjustments that can enhance course classification which results in improvements in course categorization. The next architecture, HMHA-C-BERT, which utilizes contextual embeddings from BERT, BiLSTM encodings and multiple head attention provides highly personalized course recommendations based on learner interaction patterns. To better model the temporal evolution of learners' engagement, we develop TSA-GRU, a GRU-based architecture using temporal sparse attention that allows the model to selectively attend to important time steps with respect to educational significance. The last model we contribute is a Multi-Head Bi-Encoder, a semantic course retrieval model that demonstrates competitive performance in matching learners' queries to content from large-scale course repositories. The models are tested, using authentic datasets from MOOCs, established experimental protocols including cross-validation and statistical tests on results, and evaluated by accuracy, F1-score, precision, and semantic ranking. Results show the feasibility and scalability of the models that can be integrated into building intelligent, adaptive, personalized, and pedagogically intelligent MOOC systems for learners. This work offers strong groundwork for a foundation for the next generation of adaptive education platforms about design for the learner, semantic enrichment, and continual engagement over time in online learning environments.

Keywords: MOOC platforms, Course Classification, Personalized Recommendation, Learner Behavior Prediction, Semantic Search, Deep Learning, Hybrid Models, Attention Mechanisms, Optimization Algorithms.

Résumé

Ce travail propose un cadre unifié et complet au service de l'apprentissage profond, pour améliorer l'intelligence et l'adaptabilité des plateformes de MOOCs en quatre principales directions : le besoin de classifier les cours, le besoin de recommandations personnalisées de cours, le besoin de modélisation des comportements des apprenants, et le besoin de faire de la recherche sémantique de contenus. Face à l'ampleur et la diversité croissantes des MOOCs, le cadre classique de recommandations personnalisées des MOOC ne semble plus suffisant pour capter la richesse et la diversité de la pertinence des contenus pédagogiques qui seront et devront être proposés aux apprenants selon les caractéristiques de leur contexte d'apprentissage, nous proposons de nouvelles architectures intégrant réseaux neuronaux, mécanismes d'attention, et optimisation bio-inspirée. La première contribution porte sur CNN-LOA, un réseau de neurones convolutifs dont l'optimisation par Lion Optimization permet d'améliorer la classification des cours par ajustement dynamique des hyperparamètres. La seconde architecture HMHA-C-BERT exploite les représentations contextuelles et procédurales de BERT, d'un encodage BiLSTM et d'une attention multi-têtes, afin de fournir des recommandations des cours fortement personnalisées, intégrant les interactions des apprenants. Pour modéliser le comportement temporel des apprenants, nous avons proposé TSA-GRU, une architecture reposant sur GRU, enrichie d'un mécanisme d'attention temporelle clairsemée pour permettre au modèle de se focaliser sur des moments pédagogiques pertinents. Enfin, un modèle Multi-Head Bi-Encoder a été développé pour la recherche sémantique des cours, montrant de robustes performances dans l'appariement des requêtes des apprenants avec les contenus pertinents au sein de grands corpus. Chaque modèle a fait l'objet d'une évaluation rigoureuse via des jeux de données réels issus de MOOCs selon des protocoles expérimentaux stricts (validation croisée, tests de significativité statistique, etc.) en ayant recours à des métriques (précision, F1-score, rappel, mesure du classement sémantique). Les résultats obtenus montrent l'efficacité et la scalabilité des modèles proposés qui peuvent être intégrés pour construire des systèmes MOOC intelligents, personnalisés et pédagogiquement pertinents. Globalement, ce travail constitue une méthodologie cible pour les futures plateformes éducatives adaptatives, avec des retombées directes pour les approches centrées sur l'apprenant, l'enrichissement sémantique, et l'adhésion au processus d'apprentissage en ligne.

Mots-clés : plateformes MOOCs, classification des cours, recommandation personnalisée, prédiction du comportement des apprenants, recherche sémantique, apprentissage profond, modèles hybrides, mécanismes d'attention, algorithmes d'optimisation.

الملخص

يعرض هذا العمل إطارًا موحدًا وشاملًا يعتمد على تقنيات التعلم العميق، يهدف إلى تعزيز الذكاء والتكيف في منصات التعليم المفتوح عبر الإنترنت (MOOCs) عبر أربعة أبعاد رئيسية: تصنيف الدورات، التوصية الشخصية بالدورات، نمذجة سلوك المتعلم، واسترجاع المحتوى الدلالي. واستجابةً للنمو السريع والتنوع المتزايد في محتوىMOOCs ، إضافةً إلى القيود التي تعاني منها الأنظمة التقليدية في فهم السياقات التعليمية واحتياجات المتعلمين، نقترح معماريات جديدة تدمج بين الشبكات العصبية وآليات الانتباه وتقنيات التحسين المستوحاة من الطبيعة. يتمثل المساهمة الأولى في نموذجCNN-LOA ، وهو شبكة عصبية التفافية محسّنة باستخدام خوارزمية الأسد(Lion Optimization Algorithm) ، ما يؤدي إلى تحسين كبير في دقة تصنيف الدورات بفضل التعديل الديناميكي للمعاملات الفائقة. أما النموذج الثانيHMHA-C-BERT ، فيدمج بين تمثيلات BERT السياقية، وتشفيرBiLSTM ، والانتباه متعدد الرؤوس لتقديم توصيات تعليمية مخصصة للغاية بناءً على أنماط تفاعل المتعلم. ولتمثيل التطور الزمني في سلوك التعلم، نقترحTSA-GRU ، وهو نموذج مبني على وحدات GRU مدعوم بآلية انتباه زمني متفرق، ما يمكّن النموذج من التركيز على اللحظات الزمنية الأكثر أهمية تربويًا. وأخيرًا، تم تطوير نموذج ثنائي التشفير متعدد الرؤوس لاسترجاع الدورات دلاليًا، مما يُظهر أداءً قويًا في مطابقة استفسارات المتعلمين مع المحتوى المناسب ضمن مستودعات واسعة النطاق. تم تقييم كل نموذج بدقة باستخدام مجموعات بيانات حقيقية من منصات MOOCs، ووفق بروتوكولات تجريبية صارمة تشمل التحقق المتقاطع، واختبارات الدلالة الإحصائية، ومقاييس مثل الدقة، وF1-score، والتصنيف الدلالي. وتُظهر النتائج فعالية وقابلية توسّع النماذج المقترحة في بناء نظم تعليمية ذكية وشخصية ومدركة للسياق التربوي. وبصورة شاملة، يقدم هذا العمل إطارًا منهجيًا أساسيًا للجيل القادم من المنصات التعليمية التكيفية، مع تأثيرات مباشرة على تصميم تعلّم يتمحور حول المتعلم، وتعزيز الدلالات، وتحقيق استدامة في التفاعل مع بيئات التعلم الإلكتروني.

الكلمات المفتاحية:

منصات التعليم المفتوح(MOOCs) ، تصنيف الدورات، التوصية الشخصية، التنبؤ بسلوك المتعلم، البحث الدلالي، التعلم العميق، النماذج الهجينة، آليات الانتباه، خوارزميات التحسين.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

For clarity and ease of reading, the full meaning of an abbreviation is usually given only when it occurs for the first time in the text of each chapter.

MOOC	**M**assive **O**pen **O**nline **C**ourse.
LMS	**L**earning **M**anagement **S**ystem.
AI	**A**rtificial **I**ntelligence.
ML	**M**achine **L**earning.
DL	**D**eep **L**earning.
NLP	**N**atural **L**anguage **P**rocessing.
CNN	**C**onvolutional **N**eural **N**etwork.
LSTM	**L**ong **S**hort-**T**erm **M**emory.
BiLSTM	**B**idirectional **L**ong **S**hort-**T**erm **M**emory.
GRU	**G**ated **R**ecurrent **U**nit.
TSA	**T**emporal **S**parse **A**ttention.
MHA	**M**ulti-**H**ead **A**ttention.
BERT	**B**idirectional **E**ncoder **R**epresentations from **T**ransformers.
HMHA	**H**ybrid **M**ulti-**H**ead **A**ttention.
LOA	**L**ion **O**ptimization **A**lgorithm
ReLU	**R**ectified **L**inear **U**nit
TF-IDF	**T**erm **F**requency–**I**nverse **D**ocument **F**requency
CSV	**C**omma-**S**eparated **V**alues

General Introduction

The digital transformation of the 21st century has created a revolution in nearly all elements of modern life, with education as a leader in the transformation. With information technologies incorporated into everyday life, they set the stage for some transformative changes in how learning is delivered, accessed, and engaged with. As a result of this transformative change, e-learning systems have emerged and began to grow rapidly, representing fundamental changes in practice. These systems provide flexible, scalable, and personalized learning opportunities to beyond traditional constraints of time, geography, and accessibility. The importance of the transformation is a result of providing learners with an incredibly high level of access to knowledge and resources and providing educators advanced tools for content materials, interaction, and outcomes assessment.

E-learning platforms have experienced a range of developmental stages, from basic computer-assisted learning to advanced AI-based learning environments. Today, e-learning platforms include Learning Management Systems (LMS), collaborative digital tools, mobile learning apps, and adaptive content systems, all designed with a single underlying aim: a learner-centered experience that adjusts based on individual learners' needs, wants, and prior knowledge. Studies in cognitive science, pedagogy, and IT have increasingly shown how digital platforms can support different learning styles and learning trajectories. In these environments, Artificial Intelligence (AI) has enabled new functionality, providing e-learning platforms with adaptive learning, automated feedback, predictive modeling, and personalized recommendations.

Among the most influential and disruptive innovations in this space include MOOCs (Massive Open Online Courses). In many ways, MOOCs are democratized education, as you can provide high-quality learning to millions of learners globally, based on ability and where one lives, and if they have money or support. The scale and openness of courses like the MOOC have offered new models of professional development, lifelong learning and education knowledge mobilization. Despite the utopia, scale raises many issues, particularly disengagement of learners (e.g., high dropout rates) and no course path personalization. The standardized content for a highly varied learner cohort, and with no human contact with the instructor involved makes the challenge to engage students continuously more challenging.

Given these challenges, the investigation of intelligent systems in the context of MOOCs is drawing substantial focus from both a research and innovation perspective. AI techniques including deep learning, natural language processing, and attention-based neural networks enable responsive learning environments. This allows MOOC platforms to provide content that is more relevant and meaningful, predictively model learner interactions, personalize learning experiences, and intelligently search and retrieve educational content. For example, by classifying course content based on semantic characteristics, it can be well-organized; attention-based recommendation systems add a layer of personalization and suggest pathways for learners; and predictive analytics models can identify learners that are at risk for non-completion so that they receive timely intervention.

This thesis is located at the crossroads of educational technology and artificial intelligence. Its goal is to develop intelligent solutions to facilitate the functionality and adaptability of MOOC platforms. For this purpose, this research makes four different but interrelated contributions: (1) automatic classification of courses through deep learning and semantic classification, (2) personalized recommendation of courses based on attention-based modeling of learner preferences, (3) prediction of learner's temporal behavior via sequence-aware architectures, and (4) context-aware semantic retrieval of courses using sophisticated embedding techniques. The four contributions are presented in this thesis under a sound methodological framework which encompasses data preprocessing, feature engineering, algorithm design, and empirical investigations.

Every modeled process outlined in this thesis exceeds a limitation of current MOOC systems while complementing each other within the same architecture. Accurate classification of courses, for example, will maintain consistency of content structure, and personalized recommendation engines will associate the relevant learning materials with the profile of the learner. At the same time, behaviour predictors will facilitate mechanisms of proactive support while semantic retrieval will enhance the navigation and discovery of learning resources. Combined, the processes describe a multi-tasking approach which aligns technical complexity with educational relevance that will build the basis for intelligent MOOC ecosystems. The structure of this thesis reflects the layered progression of research and organization of the outcomes in practice:

- **Chapter 1** provides an introductory overview of the evolution of e-learning systems from the early days of digital technologies for education to the more recent MOOC phenomena, and the ongoing introduction of AI into the digital learning environment and concludes with a summary of the overall research for the thesis.
- **Chapter 2** presents a state-of-the-art literature review of the main areas of research relevant to this work: course classification, personalized recommendations, learner modeling, and semantic searching and critically identifies the existing gaps, emerging

trends, and methodological challenges in the literature, which guided the design of the models proposed in this work.

- **Chapter 3** presents the most significant contribution of this thesis. It outlines the methodology and presents four new models: a course classification CNN, an HMHA-C-BERT model that combines a BERT embedding and a hierarchical attention mechanism for personalized recommendation, and a TSA-GRU model for learner behaviour analytics using temporal sparse attention and a Multi-Head Bi-Encoder framework for a semantic retrieval system based on deep contextual similarity. The models are evaluated in-depth with robust experiments and benchmarked against robust baseline performance.

  In closing, these chapters represent a joint vision of intelligent, scalable, and adaptive MOOC platforms that dynamically respond to the preferences and needs of learners through integrated pedagogical and AI models. My method not only addresses current technology issues but by incorporating sound pedagogical approaches, this thesis has potential to positively contribute to the ongoing transformation of digital education. My proposed methods will provide a sustainable alternative for engaging online learning that is more equitable and effective.

To summarize, as an increasing number of students want accessible forms of high-quality education, on a massive scale, MOOC platforms will need to develop responses to increasingly complex learner profiles. Intelligent systems, and with a specific focus on intelligent systems that leverage AI and deep learning algorithms, are a path to that development. The research and innovations described in this thesis takes us one step closer to the intelligent, learner-centred educational technology that will empower learners with a variety of experience and backgrounds across the globe.

Chapter 1: Evolution and Personalization of E-Learning Systems with MOOCs

2.1. Introduction

In the past few decades, education has seen a monumental transformation stemming from the digital revolution. Traditional learning has been slowly diminished as most of the education was offered in a classroom setting. This paradigm shift has ushered in the arrival of e-learning, which is an umbrella term used to describe any form of electronically mediated learning. E-learning has played a drastic role in democratization of knowledge regardless of geographical, economic, and social context.

E-learning has evolved in direct relation to the development of the infrastructure of the internet, multimedia possibilities, and mobile computing options. There is a historical basis evolving from older-generation computer-based training (CBT) systems. Most learning environments have transitioned to cloud-based systems, where about future learning flexibility and scalability, the possibilities are limitless. The most transformative aspect of this learning world has come from the initiation of Massive Open Online Courses (MOOCs), which can help thousands of learners' complete coursework [1]. MOOCs act as exhibit representations of large-scale inclusive education resources. These platforms are changing global education by removing entry and converging learners into self-paced and learner-centric experiences. However, as the number of MOOCs has expanded, their challenges have followed suit. Concerns arise about the high dropout rates, low levels of interactivity and one-size-fits-all delivery. The wide diversity of learner's different backgrounds, motivations, and learning styles demands some level of more personalized and adaptive elements to engage learners and achieve meaningful outcomes. As a result, we have begun to see a movement towards intelligent, data-driven learning ecosystems that can rapidly adapt and respond to learners' needs, as opposed to just providing static content. Now, AI is playing an increasingly important role in creating the next generation of e-learning systems. From intelligent tutoring systems, personalized recommendation [2] systems, learning analytics, and adaptive content, AI technology is supporting the next evolutionary phase of education systems that are scalable, personalized, and supportive of learners' needs specifically. While these developments have all seemed to improve learner experience, they have also provided instructors and platform designers with additional functionality to track progress, predict and anticipate obstacles, and act proactively.

This chapter provides the basic understanding necessary for examining these transformations. The first chapter reviews the history of online learning, detailing its development since initial experiments to the emergence of modern-day intelligent platforms. The chapter then lays out the major components and models of pedagogy involved in e-learning systems before exploring the ways in which AI is playing a central role in personalization of delivery and scalability of experience. The chapter provides a rich discussion about MOOCs including typologies, architecture, pedagogical approaches, and challenges. The chapter ends by acknowledging the increasing relevance of intelligent personalization and how it connects to important research gaps, when re-conceiving a more adaptive and learner-centered MOOC context.

2.2. History of Online Learning

The transition to online learning would not have been possible without the emergence of digital systems and technologies and communication systems. From its early experimental foundations in the 1960s, through the subsequent understanding of "bundled" online learning in the 1970s to widespread adoption of web-based technologies in the 21st century, developments have reshaped the educational landscape.

2.2.1. The Beginnings: 1960s–1980s

Before the advent of the internet, we already had a version of remote learning in postal mail correspondence courses which can be considered the earliest form of distance education. But one of the first instances of computer-assisted learning was PLATO (Programmed Logic for Automated Teaching Operations) developed at the University of Illinois in the 1960s. PLATO was a truly revolutionary technology, allowing terminals to connect to lessons that users could interact with, many years before we would formally establish the World Wide Web [3].

In the 1970s and 1980s, educational institutions were experimenting with the use of mainframe computers and early networks for use in content delivery. Video-tapes, television broadcasts, and teleconferencing were also gaining popularity as distance education tools, however, they were largely one-way forms of instruction [4].

2.2.2. The Rise of the Internet and Web-Based Learning: 1990s

The World Wide Web was born in the 1990s and was a turning point in the context of online education history. The use of web technologies permitted more interaction and usability within the learning space and accessibility to education was easier than with previous technologies. Furthermore, it was also a formative time for online education and the development of the first Learning Management Systems (LMS), which gave instructors the ability to upload course materials, track student learning, and communicate with them[5]. Institutions developed fully online courses and programs that targeted primarily adult learners and professionals, who were seeking flexible education options. Email, online forums, and multimedia content became staples of the learning experience, as well as a more sophisticated e-learning system's underlying infrastructure.

2.2.3. The Emergence of MOOCs and Intelligent Systems: 2000s–2010s

The decade of the 2000's saw an explosive increase in the popularity of online education, supported by a growth in availability of broadband internet, mobile technology, and multimedia tools. Universities and private companies began to offer online degrees and certificates and other training programs on a large scale. A watershed moment was 2012 when MOOCs (Massive Open Online Courses) arrived on the scene with platforms like Coursera, EdX and Udacity [6]. MOOCs were intended to democratize quality, affordable education by connecting millions of learners with high quality education from well-resourced universities around the world via free, online courses.

At the same time, research in Educational Data Mining (EDM) and Learning Analytics began receiving traction as a means to collect and understand data in order to improve learner behavior, engagement [7] , and outcomes [8]. There was an innovation of AI-driven tools such as intelligent tutoring systems and personalizing learning algorithms that complemented common online learning practices[9].

2.2.4. The Post-Pandemic Era and the Future of Online Learning: 2020s Onward

The COVID-19 pandemic in 2020 hastened the implementation of online education around the world. Educational institutions at every level had to make a quick transformation to online/digital learning; overnight they discovered both the capabilities and shortcomings of current e-learning structures[10]. This time period has prompted new investments in the infrastructure, ongoing professional learning for teachers, and technology-enhanced educational technology utilizing artificial intelligence.

Today, online learning is evolving towards a hybrid model of both synchronous and asynchronous methods; online learning is integrating strong elements of artificial intelligence to personalize learning; and the profession is focusing on the learner engagement, learner adaptability, and learner accessibility. The field continues to emerge towards intelligent, inclusive, and on-demand data-informed educational models.

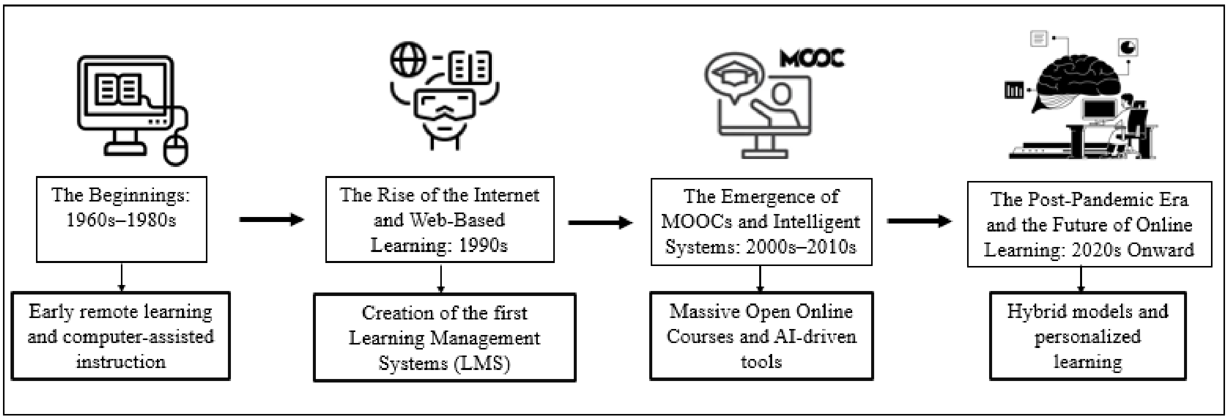


Figure 1: History of Online Learning

The infographic documents in Figure 1 showed the evolution of online education from its roots in the 1960s (PLATO systems, mail-based learning) and web-based learning in the 1990s through the development of MOOCs and AI systems (2000s-2010s) as well as, the shift caused by COVID-19 into today's AI-enabled hybrid models.

2.3. Key Components of E-Learning Systems

E-learning systems are complex systems that have different components that are technological, pedagogical, and subject-specific that are cohesive in nature and are designed to support online learning. The design of e-learning systems is a determining factor in its efficiency and effectiveness, and the extent of support for learning, learner engagement, and the assessment and personalization of learning in e-learning systems as dimensions of 'learning' in e-learning systems and needs to be evaluated to the level that is designed to support effective learning.

2.3.1. Learning Management Systems (LMS)

Learning Management Systems (LMS) are the primary building blocks in almost all e-learning environments. The LMS is the foundation for managing, delivering, and tracking educational content. LMS are platforms that provide a variety of functions, which include course creation, content delivery, assignment submission, assignment grading, and learner analytics. Popular LMS platforms like Moodle, Blackboard and Canvas provide tools to educators to support synchronous and asynchronous learning. They are integrated into resources such as multimedia, forums, and quizzes and other external tools, allowing for a full learning ecosystem [11]. LMS platforms also act as the framework for the SCORM (Sharable Content Object Reference Model) and xAPI interoperability standards, allowing content to be reused and portability between systems.

2.3.2. Collaborative Learning Tools

Collaboration is vitally important in today's e-learning systems, and communications and collaboration tools for peer and instructor interactions foster social presence, peer interactions, and the co-construction of knowledge [1]. Collaboration tools include discussion forums, chatrooms, wikis, shared whiteboards, and video conferencing tools like Zoom, Google Meet, and Microsoft Teams. Collaboration tools facilitate Community of Inquiry (CoI) environments that foster cognitive, social, and teaching presence [12]. Whether integrated into an LMS or used as a standalone app, collaborative tools allow real-time interactions, peer feedback, collaborative assignments, and virtual office hours - all important elements of learner motivation and engagement, especially in asynchronous courses [13].

2.3.3. Adaptive Digital Content

Adaptive digital content is educational content that adapts based on the learner's profile, preferences, performance, and goals. Adaptive content is personalized based on individual learning experiences leveraging Artificial Intelligence (AI) and Learning Analytics, which applies a set of data analytics on the learner's behaviors and interactions to develop personalization pathways that are right for them. Adaptive content can take shape as personalized quizzes, modular video lessons, reading material adjusted to skill levels, and intelligent tutoring systems. The use of rule-based systems, machine learning models, and learners' modeling frameworks in designing adaptive e-learning environments appears to be growing[14]. Adaptive content increases engagement and provides pacing but more importantly, has found to increase learning by matching learning resources to a unique learner.

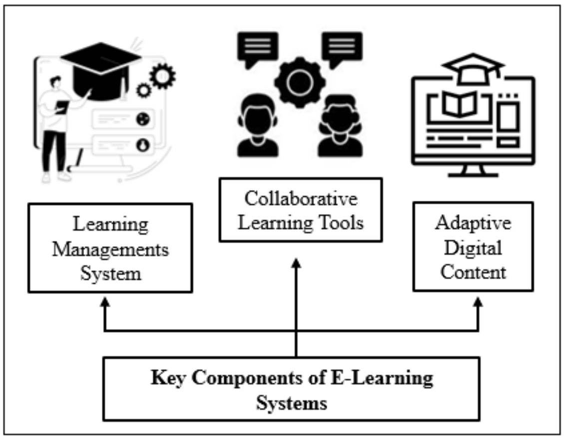


Figure 2: Key Components of E-Learning Systems

The infographic Figure 2 showcases three important pieces of the puzzle of modern e-learning environments: Learning Management Systems (LMS), Collaborative Learning Tools and Adaptive Digital Content. Together, these components provide scalable, interactive, and personalized learning experiences.

2.4. Pedagogical Models in E-Learning

Pedagogical models in online learning are the theory and practice for designing, delivering, and assessing learning in digital learning environments. Pedagogical models explain how learners interact with content, instructors, and peers and they influence the way in which learning is achieved. Three of the most recognized models - Transmissive, Constructivist, and Socio-Constructivist - provide the foundation for many e-learning systems in today's different educational institutions.

2.4.1. Transmissive Model

The Transmissive Model, known as the behaviorist model, involves the straight transmission of knowledge from the teacher to the learner. In this model, the teacher is the primary actor who transmits knowledge in a linear and structured format while the learner is only responsible for receiving and reproducing this knowledge. This model relies upon a transmitter (the teacher) and a passive recipient (the learner).

In an e-learning context, the transmissive model is often demonstrated through video lectures, the use of PowerPoint slides, automated quizzes completed after watching the lecture, and a straight replication of what occurred in the classroom (asynchronous e-learning essentially replicates the experience) the basis for this model of e-learning is still found in the traditional classroom activity. The transmissive model is standardizable and scalable, making it applicable to massively scalable online courses (such as MOOCs), particularly for the foundational knowledge[15] . Nonetheless, the transmissive model has critics as it might lead to passive learning and a lack of application of higher order thinking skills related to analysis, synthesis and creativity [16].

2.4.2. Constructivist Model

The Constructivist Model claims that learners actively build their own meanings and knowledge of the world by experiences and reflections. In the Constructivist Model, learners are placed at the center of the learning process rather than teachers, learning, especially experiential learning, is most needed, with some space originating for more conventional learner-centered activities and engagements, such as problem-solving and critical thinking.

In the e-learning domain, constructivist approaches can be directly reflected and encountered through active and collaborative environments, exploration environments, project-based learning modules, and a range of interactive simulations. Learners are encouraged to explore the content, process their findings, hypothesize, and test their ideas with useful scaffolding tools in development [17]. Learning needs to be real, and contemporary learning environments like a virtual lab, gamified modules, and adaptive learning are likely to embody constructivist principles and requirements to develop the deep understanding of information distinctly required for intrinsic motivation.

2.4.3. Socio-Constructivist Model

The Socio-Constructivist Model draws on the ideas of Vygotsky (1978). The Socio-Constructivist Design Model asserts that learning is social and that knowledge is constructed socially through dialogue, negotiation, and collaboration. E-Learning environments that activity co-construct knowledge to embody the Socio-constructivist principles facilitated by collaborative tools such as discussion forums, peer-review systems, group projects and social media. These tools provide a mechanism for learners to participate in communities of practice, co-construct knowledge, and foster key 21st century skills, such as communication, collaboration, and digital citizenship [18].

Further, socio-constructivism has strong connections to connectivism learning theories in networked learning environments, where knowledge is fluid and collectively constructed -

regardless if learning in networked spaces[19].

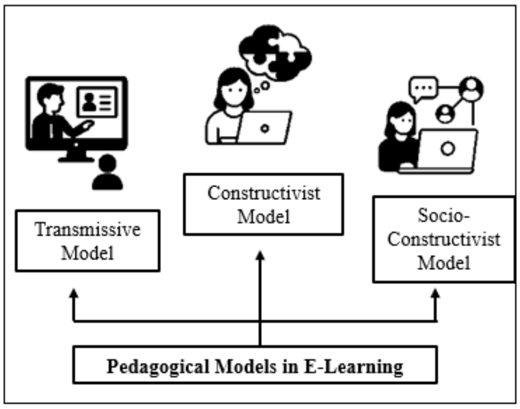


Figure 3: Pedagogical Models in E-Learning.

The Infographic Figure 3 represents three primary approaches in online education: the Transmissive Model which focuses on delivering knowledge directly from a source, the Constructivist Model which emphasizes exploratory, learner-centered knowledge construction, and the Socio-Constructivist Model which emphasizes collaborative knowledge construction both socially and cognitively. Together these models inform the design and execution of effective digital learning spaces.

2.5. The Role of Artificial Intelligence in E-Learning

As digital transformation continues to evolve the education sector, Artificial Intelligence (AI) has become a vital source of innovation and driver of change for e-learning systems. AI does more than simply augment existing educational platforms with adaptive, data-driven, and personalized digital instruction; it recreates and revolutionizes the digital teaching and learning experience. AI applications have the potential to increase learner engagement, improve instructional delivery and to enable development of meaningful learning experiences for learners at scale (especially in MOOCs and intelligent learning environments) [19]. Latest developments in artificial intelligence, such as Reinforcement learning, have showcased how smart systems can adjust themselves as per changing environments, showcasing possibilities of further use of such systems for applications in education [22], [23] .

2.5.1. Learning Personalization

Personalization in e-learning is the ability of a system to tailor learning content, learning pathways, and instructional methods to the needs, preferences, and behaviors of the individual learner. Artificial Intelligence helps with this because it is based on sophisticated algorithms capable of analyzing loads of learner data including performance measures, activity records, time on task, and even cognitive or affective states.

AI-enabled adaptive learning systems model student profiles dynamically and in real-time using techniques such as Bayesian Knowledge Tracing (BKT), reinforcement learning , and deep learning. They adaptively direct the learner's trajectory while ensuring the content is both contextually relevant and pedagogically beneficial. If a learner struggles in a concept, the AI could redirect him or her to remedial content and/or provide alternative representations of instrumental information such as explanations in text, video, or interactively based on the learner's approach to learning and learning style [24].

Knewton, Squirrel AI, Smart Sparrow and others have shown personalized learning pathways and positive learner outcomes in areas of learner satisfaction, retention and engagement, achievement and completion.

2.5.2. Intelligent Tutoring Systems (ITS)

Intelligent Tutoring Systems (ITS) are sophisticated learning technologies that use artificial intelligence to offer individualized feedback, which is normally provided by another person, namely a human tutor [25]. Other than Intelligent-E-Learning systems, which typically provide all learners with the same learning (regardless of prior knowledge), ITS systems dynamically personalize instruction according to learners' cognitive states, monitoring learners' conceptual state and progress likewise; its data helps to provide targeted instruction, support, and feedback without waiting for the instructor.

The main components of intelligent tutoring systems are designed as modular interdependencies with four components:

- The Domain Model: is the body of information/knowledge to be learned. This includes the important concepts, skills, rules, and problem-solving procedures regarding the instructional content. Within the domain model decodes student input when generating appropriate feedback; for example, if the domain model is for programming, it would contain all relevant syntax, logic structures, and debugging.
- The Student Model: provides a real-time and personalized profile for the learner. It monitors the learner's cognitive state, the knowledge they have acquired, the pace at which they learn, and potential misinterpretations. To support real-time instructional decisions, the student

model models the student based on their actions, e.g. their response patterns, how much time they spend on a task, and their error patterns. This model continues to learn from the student’s actions, such that the ITS at any moment can discriminate between a novice, who struggles to understand foundational concepts, and an advanced learner who is able to deal with higher-order challenges.

- The Tutoring Model: encodes pedagogical strategies and rules of decision-making that contribute to how the instruction is carried out. For example, the tutoring model manages the sequencing of materials, the generation of hints, which feedback modalities, and potentially when to intervene or challenge the learner. So, depending on the learner's state the tutoring model may present either scaffolded examples, ask Socratic questions, or prompt metacognitive reflection to solidify comprehension.
- The Interface Model: is responsible for managing the interaction between the system and the learner. This includes the graphical user interface (GUI), i.e., visual displays, interactive elements, multimedia components, response mechanisms. A well-designed interface model can help provide a learning environment that is accessible, intuitive, and engaging, with respect to learner motivation and usability.

Each of these components is brought to life by AI through real-time decision-making, pattern recognition, and predictive analytics. An intelligent tutoring system (ITS) can examine a learner's behavior, and based on predicted future performance, adapt instructions for that learner automatically. Suppose, for example, that the ITS recognizes that the learner keeps making the same mistake when solving an algebra equation. In that case, the ITS could initiate the tutoring model designed to either provide contextual hints or offer a worked example [26].

Recent works in the AI domain demonstrated that hybrid optimization approaches [27] could also be useful for adaptive decision making in dynamic environments [28], [29].

2.5.3. Recommendation Engines

Recommendation systems that rely on Artificial Intelligence have become commonplace features on e-learning platforms, most notably MOOCs, as learners can feel overwhelmed with the sheer volume of educational content available to them. AI-based recommendation systems utilize algorithms (i.e., collaborative filtering, content-based filtering, hybrids) to suggest learning material, activities, and courses based on the user’s learning goals, user performance, and user preferences.

More sophisticated recommendation engines also draw on techniques involving Natural

Language Processing (NLP), semantic analysis, and knowledge graphs to match learners with the most appropriate contextual resource. Additionally, recommendation engines can use affective computing techniques to detect learner emotions in order to help generate recommendations [30].

Given that learners often slip away from acquiring knowledge and skills in informal online environments like Coursera (all paid modules) or edX, recommendation systems would be useful in creating engagement by suggesting surrounding modules, peer discussions, or additional content, which could eventually create uptake. Focused engagement activities for a learner are likely to help reduce slipping and can develop a deeper learning experience that learners tend to seek.

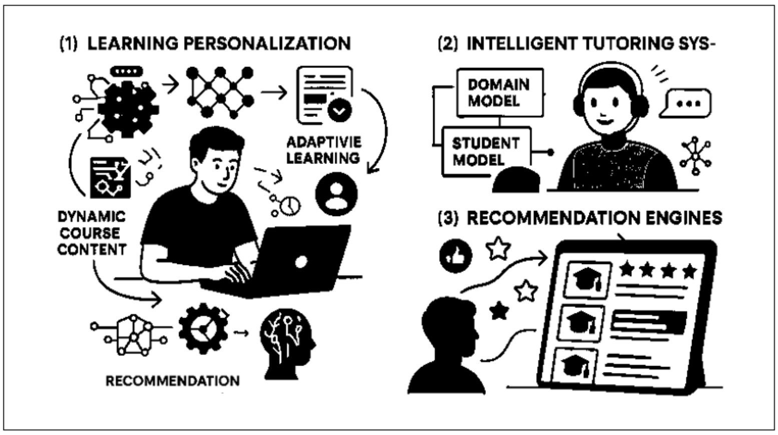


Figure 4: The Role of Artificial Intelligence in E-Learning.

The infographic maps in Figure 4 out three major applications of AI in current digital learning contexts: Learning Personalization through adaptive content and dynamic learner modelling[31]; Intelligent Tutoring Systems which give students unique feedback based on domain, student, and tutoring models; and Recommendation Engines that provide engagement and learning outcomes through algorithms, natural language processing, and contextual suggestions to course developers and students.

2.6. E-Learning Platforms and Ecosystems

E-learning platforms are the technological foundation of virtual education. Simply put, e-learning platforms are the interface for learners, instructors, and content. E-learning platforms represent an environment for capability to manage the organization of educational resources and provide ways for communicating, assessing, personalizing, and using analytics. Learning technology is moving at a rapid pace, so there are now different styles and architectures of these platforms and typologies specific to different pedagogy and learner contexts. This section describes the classification of e-learning platforms, their technical architecture, and the application of AI in them, along with actual examples.

2.6.1. Classification of Platforms: LMS vs. MOOC vs. Custom Platforms

E-learning platforms can be grouped into three main categories, each with somewhat different aims and capabilities:

- Learning Management Systems (LMS)**:** LMS's such as Moodle and Blackboard are designed for use in structured learning contexts (e.g., academic settings and corporate training programs). LMS platforms provide a suite of tools for educators to deliver course content, assess learners, track participation and report on compliance. These systems are structured to provide control for instructors and compliance for institutions.
- Massive Open Online Courses (MOOCs): MOOC platforms are represented by organizations like Coursera, edX, and FutureLearn. MOOCs feature open access, massive participation, and mostly independent learning. The design of a MOOC is guided by principles of scalability and independence, organized according to features like moderated discussion forums, artistic video lectures, and automated quizzes.
- Custom E-Learning Platforms: Organizations and startups can leverage the excitement that people have about obtaining e-learning customization by simply building their own systems that cater to audiences, industries, or educational models. These systems will often have interesting niche feature implementations including gamification, AR/VR modules and specialty analytics. E-learning systems will regularly attempt the use of modularized microservices and front-end frameworks with a flexible basis in order to facilitate rapid iterative design or incorporate learner input.

To address the changing landscape of e-learning infrastructure, numerous recent studies have subsequently developed offering detailed frameworks and taxonomies which demonstrate how e-learning platforms can be classified according to their fundamental technological structure, pedagogical positioning, scalability, and features for personalization. These classifications are essential for educators, system designers, and researchers if they wish to select or develop e-learning platforms that align with specified learning objectives and profiles for learners. The following studies represent meaningful contributions to the field, providing overall insights into how different types of e-learning contexts can be organized and hopefully tailored, and ultimately utilized for a variety of learning contexts. [32] This study presents a complete classification of MOOCS in terms of massiveness and openness. The new classification has four categories: small and less open, small and more open, large and less open, and large and more open. This taxonomy enables MOOC purveyors, educators, and researchers to understand and design courses that best serve instructional needs of many learners.[33] The article provides a systematic review of personalized learning software systems, providing a potential taxonomy based on 3 key aspects: environment, learner model, and content. The work is designed to build on each of these three aspects to design learning experience systems that consider the learner preferences and needs.

2.6.2. Platform Architecture and Technology Stack

The architecture of today's e-learning platforms usually has a layered structure, where components are modular. Each e-learning platform is comprised of a front-end interface (presentation layer), back-end application logic, and connection points or integrations between the various layers. Here are some of the common layers for plenty of architecture in e-learning.:

- The Presentation Layer: web and mobile based user interfaces were developed using frameworks like React, Angular, or Flutter.
- The Application Logic: this layer was developed using backend technologies like Node.js, Django, or Laravel. Application logic will consider any user authentication, the mechanism to deliver content to users, the user-facing logic from quizzes, and the tools for communication.
- The Database Layer: database stores either relational (e.g., MySQL, PostgreSQL) or NoSQL (e.g., MongoDB) data types for user information, course content, and interactions.
- Media streaming: cloud-based services (e.g., AWS CloudFront, Vimeo, YouTube APIs) often operate to manage video content and the potential access term for scalability.
- Analytics Engine: analytical modules and external services (e.g., Google Analytics, xAPI/LRS system) inform for learning analytics and reporting needs.

Advanced future platforms are designed utilizing microservices architecture to allow for scalability and maintainability, also aiding and/or incorporating thorough API gateways for third-party services e.g., Ai-based recommendation services, payments, or plagiarism checkers.

2.6.3. Integration with AI and Analytics

The use of artificial intelligence and data analytics has transformed the possibilities offered by e-learning platforms. Some examples of areas of integration include:

- **Tailored recommendations:** AI models such as collaborative filtering make recommendations on courses, content, or learning pathways based on the learner profile and registration and learning interaction history.
- **Automated grading and feedback:** Natural language processing (NLP) and machine learning models can analyse open-ended responses and provide feedback to the learner in a timely way.
- **Predictive analytics:** Learner interaction data models of prior engagement behaviours through historical trace data, predicting learner dropout likelihood, engagement behaviours, and action relevance of content.
- **Adaptive assessments:** (smart tests) assessors dynamically and automatically change assessment question difficulty or change the option of content delivery based on a learner's assessment performance.

Additionally, AI-enhanced platforms often utilize intelligent chatbots [34], sentiment analysis, and automated scheduling systems, improving the support provided and personalization. Furthermore, recent developments in artificial intelligence have proved the usefulness of collaborative intelligent systems in making decisions and navigating autonomously, thereby demonstrating the versatility of AI techniques to be applicable in various areas [35], [36].

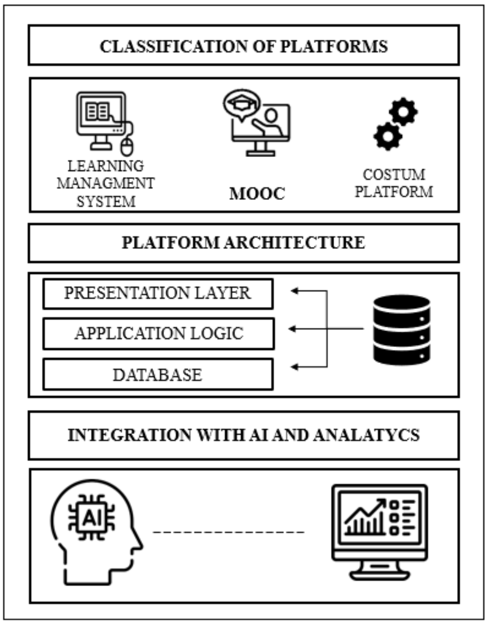


Figure 5: E-Learning Platforms and Ecosystems.

The infographic in Figure 5 explains the technical infrastructure of virtual education by tussling with Learning Management Systems (LMS), Massive Open Online Courses (MOOCs), and Custom Platforms. The infographic illustrates three architectural layers: presentation, logic, and data. These aspects are critical for e-learning but often overlooked, particularly how AI and analytics inhabit those layers to support a personalized learning environment that is at once scalable and data driven.

Recent studies have documented the move from traditional learning management systems to e-learning platforms with the help of AI primarily with the integration of large language models that provide personalization in learning, adaptive feedback, and learner support. Examples of AI integration are using modular inclusion in LMS environments, such as MAIC, and tools like SocratiQ, which replicate human-like tutoring. The research also discussed platform classification, educator readiness, and the critical inclusion of social-emotional learning (SEL). Collectively, these discussed advancements indicate a move towards more intelligent, adaptive, and learner-centric digital learning systems.[37] describes a next generation Learning Management System (LMS) that uses large language models (LLMs) to dynamically adapt content presentation, feedback, and assessment methods to match learner use patterns, therefore providing a specific example of AI-enhanced personalization in educational systems.[38] SocratiQ is an AI-based interactive tutor that imitates Socratic inquiry via generative models, and illustrates how intelligent agents can facilitate learners' personalized trajectories and increase learner participation, especially in self-paced MOOCs.[39] The paper proposes the idea of MAIC (Massive AI-empowered Courses) where real time large language models are used to provide dynamic feedback, content recommendation, and learner support, ushering in a new era of AI-MOOC convergence.[40] A4L provides a modular framework for embedding a variety of AI tools, such as natural language processing and recommendation engines, within established LMS systems. A blueprint for constructing future e-learning platforms with intelligent, embedded layers [41]. [42] reconsiders the classification of MOOCs with technological, pedagogical, and learner-centered factors, providing a more nuanced framework for examining the ways platforms vary and where they should be used.[43] examines teachers' readiness for integration of Artificial Intelligence (AI). The study also identifies training gaps and infrastructure needs identified as necessary for success in e-learning platforms, especially in rural context and in secondary education environments. [44] While this study addressed physical environments, the EDU-AI model shows how principles of intentional design can be applied to digital learning environments to enhance learner experience by allowing AI to customize learning experiences.[45] Although this study does not specifically relate to education, it explores a customizable content filtering system that has the potential to be implemented in educational platforms to establish credibility for users and better curate content.[46] examines the design of interactive blended courses to develop students' social and emotional learning (SEL) skills and engagement. Based on Merrill's first principles of instruction, the study found that active learning methods produced significant improvements in students' SEL skills, and their level of engagement in higher education contexts.[47] explore the incorporation of social and emotional learning into digital ecosystems. The focus will be on new areas of research on how digital technologies can assist in developing SEL skills and that different approaches (holistically) to e-learning settings will be considered.[48] examine the oscillations of the transition to (and fade away from) online learning, providing the reasons for the popular adoption and then its partial decline. The study describes difficulties of adaptability to online learning and suggests approaches for improving the sustainability and impact of online learning platform.[49] present E-TRIALS, a program that supports experiment experimentation and student evaluation in computer-based learning environments. The paper highlights methods for data-driven decision making aimed at refining instructional designs and student achievement.[50] use techniques in machine learning to study student engagement in Moodle--a widely used learning management system. The study identifies patterns in student engagement, which can provide some understanding of student behaviour and help inform teaching practices and increase learner engagement.

2.6.4. Case Studies: Moodle, EdX, Coursera

To describe the diversity and effects of contemporary e-learning ecosystems, we present case studies of three significant platforms:

- **Moodle:** An open-source LMS widely used in academia, which is modular and can be highly customized. Moodle supports SCORM/xAPI content, can facilitate competency-based education and learning analytics. Moodle is integrated with, or has the capacity to integrate with, a growing number of AI plugins such as MyLearningAnalytics and IntelliBoard.
- **edX:** Originally created by MIT and Harvard, edX is a MOOC platform that supports professional and university courses. It is based on the Open edX platform, which is modular, creating the potential of using a wide variety of AI-empowered features, including personalized dashboards and automated assessments. EdX accepts Learning

Tools Interoperability (LTI) applications that can be integrated into a blended learning environment.

- **Coursera:** One of the largest MOOC providers, with a wealth of content that is produced by partners, which includes universities and companies. Coursera uses refined algorithms that employ very advanced AI methodologies to recommend courses, pacing of learners, and automatically grades learners' assignments. Its platform includes interactive dashboards, peer review assignments, and workforce alignment tools that can facilitate workforce professional development.

Developments in e-learning platforms like Moodle, edX, and Coursera represent recent evolutions toward AI-enhanced tools for learner and engagement and learning performance. Moodle's open-source design allows for permutations of the platform, with a repository of AI-based plug-ins, including IntelliBoard, which can provide analytics that are predictive and offer dashboards that facilitate real-time performance tracking. On the other hand, edX, using a modular and open-source platform called Open edX, has provided AI tools such as personalized recommendations for courses, automated grading tools, and the use of real-time analytics to guide learner pathways. Finally, Coursera extends the platform toward the use of complex AI algorithms for recommendations for course choices and pacing, as well as automated grading, according to a recent introduction of AI-assisted tools for peer grading of assignments with a focus on efficiency and quality feedback.

Moodle has developed into a flexible open-source Learning management system (LMS) which can support a wide range of educational needs through a modular architecture and is establishing standards for integrating with AI-powered tools.[51]IntelliBoard integrates seamlessly with Moodle and can provide the more advanced learning analytics feature, predictive learning-based insights, and customizable dashboards, to support views of student performance and engagement. [52] Moodle has an array of built-in analytics and third-party integrations that provide educators with insights to improve personalization within the online learning experience. [53] Moodle has several plugins to support xAPI and change standard Moodle activities into xAPI format, which will enable you to capture more detailed data regarding the interactions of learners to produce more advanced analytics. [54] Moodle's plugin ecosystem has tools such as IntelliBoard and Learning Locker that offer real-time monitoring of activities, customizable reports, dashboards, and graphics that support data-informed decision-making to help enhance education. Open edX, the platform behind edX, provides an environment that is modular and flexible while being designed with scalability in mind and AI integration.[55] Open edX is the open-source software that supports edX courses and accepts various AI-powered tools and enables scalable and personalized learning experiences based on modular course structures and integrations to provide various customization options. [56] Open edX supports multiple features including LTI integrations, content libraries and supports modular courses that allows for the integration of AI-powered tools and personalized learning experiences.[57] Proctor track offers custom integration with Open edX via LTI enables a secure online proctoring and assessment environments on the Open edX platform. [58] The new Quince release of Open edX has personalized learning paths which allows learners to progress through their courses at their own pace and further personalize the learning experience. Coursera has one of the biggest MOOC platforms and used artificial intelligence when dealing with a scalable peer assessment process in a personalized learning environment.[59]Coursera is leveraging AI-based grading for peer reviews as a way to be more efficient and consistent, while giving learners immediate and valuable feedback.[60] Coursera leverages AI to personalize recommendations of courses for learners based on their personal professional development goals, supporting learners in skills development and career growth.[61] This course taught how to use Generative AI to perform data-based performance reviews, enhance the quality of feedback, and better connect individual professional development goals with organizational goals and objectives.[62] This course examined how AI can improve processes, decision-making, and employee productivity overall with workforce management.

2.7. MOOCs Specific Characteristics and Challenges

2.7.1. Definition and Typology of MOOCs

MOOCS (Massive Open Online Courses) are a disruptive innovation in the digital education landscape, with a flexible online format, that has made access to learning more scalable and accessible across the world, across disciplines and organizations. MOOCs, which are extremely massive (unlimited access/enrollment), open (free/low cost; no formal requirements), and online (web-based), challenge the educational paradigms of traditional institutions and make high-quality content freely accessible to people around the world. The term MOOC was created in 2008 in reference to the course Connectivism and Connective Knowledge (CCK08) taught by George Siemens and Stephen Downes. CCK08 was anchored in the learning theory connectivism, which emphasizes the interplay of social and technological networks in learning. The first MOOC course created the path for the different types of MOOCs that we see today[63].

Over the years, MOOCs have transformed into various types which express a variety of educational ideologies, learner appointments, and levels of personalization:

2.7.1.1. xMOOCs (Extended MOOCs)

These xMOOCs follow a well-defined structure with an instructivist approach aligned with the pedagogy of a traditional classroom. Video lectures, quizzes, assignments and automated assessments provide learning experience with limited interaction. Examples of xMOOCs include Coursera, edX, and FutureLearn. While they have the advantages of scale and efficiency, the demographic of students has criticized a lack of interactivity and passive content delivery [64].

2.7.1.2. cMOOCs (Connectivist MOOCs)

Based on connectivist theory, cMOOCs value autonomy, social learning, and constructive peer learning in co-constructing knowledge. cMOOCs are focused on learner engagement, not a linear dissemination model from expert to student. cMOOCs promote learning that happens in forums or other group contexts, and by using wikis in a collaborative way, blogs or user-generated content shared in a decentralized way [65].

2.7.1.3. Hybrid MOOCs

These classes combine the instructivist components of xMOOCs with the networked interactivity of cMOOCs. It should be noted that the instructors will use the blend of content and process also integrate the course in shared blended learning environments with video content, peer assessments, discussions, and occasional synchronous webinars and local meet-up components.

2.7.1.4. SPOCs (Small Private Online Courses)

SPOCs adapt MOOC platforms by limiting the course size to a small, closed cohort such as enrolled university students or corporate trainees. The course generally provides individualized feedback, direct access to instructors, and more control over assessments [66].

2.7.1.5. AI-Enhanced & Personalized MOOCs

Adaptive learning with artificial intelligence and learning analytics has created a new generation of MOOCs that offer adaptive content sequencing, dynamic feedback, and personalized learning trajectories. These models can adjust in real-time based upon learners' engagement patterns, performance measures, and preferences.[67]

Type	Pedagogy	Scale	Personalization	Examples
xMOOC	Instructivist	Massive	Low	Coursera, edX
cMOOC	Connectivist	Massive	Medium – Community	CCK08, P2PU
Hybrid MOOC	Mixed	Variable	Medium	Blended university models
SPOC	Instructor-led	Small	High	HarvardX SPOCs
AI-Enhanced MOOC	Adaptive	Variable	High	FutureLearn pilot systems

Table 1: MOOC Typologies and Key Features

The typological overview in Table 1 maps a conceptual framework to make sense of MOOCs, generally, but also to make sense of the struggles they face, which are articulated later in this chapter. The different migrations of MOOC's reflect changing technologies and learning theories, as well as continued pedagogical imperatives.

2.7.2. Architecture of MOOC Platforms

The architecture of the Massive Open Online Course (MOOC) uses technology and a pedagogy subsystem to develop complex technological systems for the development of large-scale, flexible and scalable digital learning outcomes. Each MOOC platform incorporates several different technologies and solutions that work together to facilitate and manage learners and their learning in terms of course content, learner engagement, assessment, data collection, and learning analytics, all whilst ensuring the MOOC is usable and accessible for the global audience of learners. In a high-level overview, a typical architecture of a MOOC platform can be categorized into five layers.

2.7.2.1. Content Management Layer

This layer provides the infrastructure for the development, organization and delivery of course content. This layer will provide the authoring tools that allow instructors to upload and organize video lectures, quizzes, readings, and discussion prompts. Modular blocks of content are utilized by edX and Coursera to allow for reusing and adapting course designs [64].

2.7.2.2. Learner Interface and Interaction Layer

This is the interface in which learners' access and engage with the platform. Key features include supporting video streaming, interactive assessments, discussion forums, note-taking, and using gamification. Responsive web technologies will support access from any device and mobile compatibility[68].

2.7.2.3. Communication and Collaboration Layer

This is the front-end interface where learners interact with the platform. It supports video streaming and interactive assessments and has tools for discussion forums, note-taking, and gamification. Responsive web technologies to allow access from multiple devices while mobile support is built-in [69].

2.7.2.4. Assessment and Feedback Layer

The Assessment and Feedback Layer comprise automated grading systems alongside peer-reviewed assignments. One influential research account from ACM stated that the application of statistical models to alleviate grader bias greatly strengthened peer-grading accuracy on open-ended assignments in MOOCs[70]. More recent developments have integrated Large Language Models (LLMs) to automate feedback and standardize it - achieving a closer agreement to expert grading than peer review did.

2.7.2.5. Data and Analytics Layer

The back-end architecture captures large amounts of user data, including clickstream logs, engagement scores, and performance logs. These data support educational data mining (EDM) and learning analytics (LA) and allow instructors and researchers to identify student progress, predict dropout rates, and adapt their teaching [71].

2.7.2.6. Infrastructure and Integration Layer

The Infrastructure and Integration Layer is responsible for backend components, security [72] and scalability, and interoperability with external systems via API and LMS integration. Though technical architecture studies of specific MOOCs are not common in the open literature, the Infrastructure and Integration Layer is generally comprised of microservices, cloud technologies, authentication systems, and compliance technologies. In terms of MOOCs, it is critical that we maintain security [73] through providing secure access to learning platforms. To address this, [74] have proposed a hybrid multi-factor authentication framework that incorporates biometrics and implements behavioural analysis, which will enhance the verification process for users by preventing non-authorized people from being able to access the learning environments of the Internet. In addition, [75] proposed a hybrid secure routing approach for mobile ad-hoc networks (MANETs), highlighting the importance of robust security mechanisms that can be adapted to large-scale e-learning platforms [76]. Additionally, most research in regards to cloud-based deployment of MOOCs addresses the need for a decoupled architecture and global scalability as respective design principles.[77]

Below we illustrate a visualization of the layered architecture in Figure 6 that encapsulates how this stack of interconnected components supports scalable and interactive online learning environments:

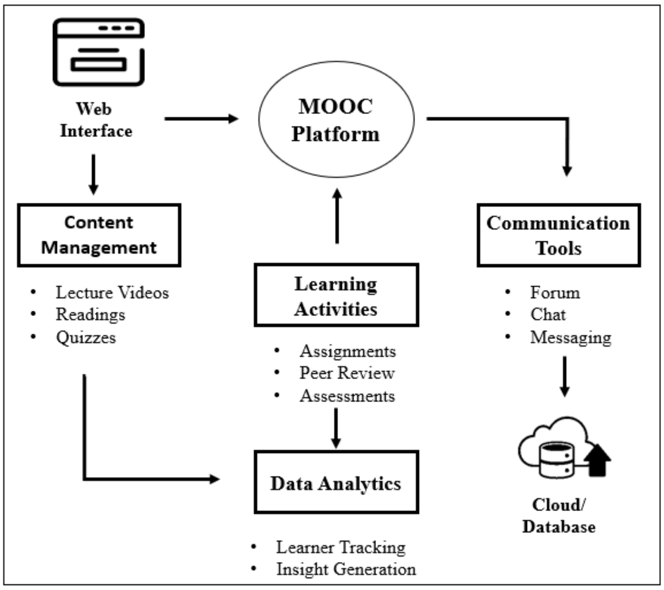


Figure 6: Architecture of MOOC Platforms

Recognizing the architecture of MOOC platforms in Figure 6 not only reveals the technical scaffolding of large-scale online education as it is now developing but the pedagogical implications as well. As MOOC platforms develop it becomes more apparent that they are beginning to include artificial intelligence, personalization engines, and systems that may integrate into an institution’s existing system creating a future that will soon have learning ecosystems that are smarter, more adaptive and more focused on learners.

2.7.3. Pedagogical Models and Instructional Design in MOOCs

While there are instructional design frameworks for course development, for example ADDIE, Dick & Carey, and Gerlach & Ely, MOOCS may use a range of educational paradigms, such as behaviorism, cognitivism, constructivism, connectivism, and heutagogy.

2.7.3.1. Behaviorist and Cognitivist Roots

The majority of xMOOCs are based on behaviors and cognitivist principles. In xMOOCs, learners are provided structured content where subsequent quizzes or assignments are included to reinforce the learning, or to test for mastery. ADDIE the instructional design artifacts of Analysis, Design, Development, Implementation, Evaluation - served as a standard and remains a model description of developing these courses, as it carefully aligns objectives, content, and assessment with the needs of the learner[78], [79].

2.7.3.2. Constructivist and Social Learning Approaches

Constructivist MOOC designs focus on learner activity and collaboration. A constructivist MOOC incorporates a mix of case studies, peer assessments, project-based learning, and social media tools to promote active knowledge construction, in line with Vygotsky's social learning theory[80].

2.7.3.3. Connectivism and cMOOCs

Connectivist MOOCs (cMOOCs) were first conceptualized by [63] as a way of learning that involved navigating distributed networks of knowledge. Designing learning experiences around learner autonomy, diversity, openness, and connectivity using multiple platforms such as blogs, social media, or RSS feeds was center stage in stories about cMOOCs, as demonstrated by their courses CCK08 and PLENK2010. However, since then, research has reported that most current MOOCs are poorly linked to authentic connectivist practices [81].

2.7.3.4. Heutagogical and Learner-Centered Models

Heutagogy centers around the importance of self-direction and metacognition in learners and describes autonomy as ownership over personal learning trajectories rather than just mere choice of path. This is very relevant in the case of MOOCs where one can assume a significant variability of levels of motivation, prior knowledge, and digital literacy [82].

2.7.3.5. Instructional Design Frameworks

In addition to ADDIE, Dick & Carey and Gerlach & Ely, offer systematic models that provide a way to connect instructional goals to assessments and approaches. Such models help inform the effective and consistent construction of a MOOC out of pedagogical intentions [83].

2.7.3.6. Design Patterns and Scaffolding

Patterned approaches, such as inquiry-based SNaP and reusable interaction templates, assist MOOC designers in developing scalable and socially rich learning environments. Theoretical efforts around cooperation patterns and cognitive engagement architectures [84] established the need for intentional scaffolding that supports higher-order thinking in MOOCs.

What we see in practice with most MOOCs is the blending of many pedagogies structured behaviorist/cognitivist modules, social constructivist activity, engagement through connected networked activities, and learner autonomy. There are instructional design frameworks that operationalize the integration of these pedagogies in course flows and assessments. Although there are plenty of opportunities for integrating the pedagogical richness of social constructivism, connectivism, and heutagogy, the implementation of connectivist and Heutagogical principles in MOOCs is still limited. This suggests that MOOC designers and facilitators need to incorporate social interaction, added scaffolding at scale, as well as autonomous ownership of the learning experience using a more deliberative approach.

2.7.4. Learner Profiles and Diversity in MOOCs

MOOCs display remarkable learner variability, including differences in motivation, demographics, engagement behaviors, and self-regulatory skills. Understanding this variability is crucial to developing a strategy for adaptive interventions.

2.7.4.1. Motivational and Self-Regulation Differences

Learners enroll in MOOCs for various purposes of learning such as career enhancement, academic development, curiosity or certification. Those taking MOOCs for credentials were generally found to be taking the course for more of a learning purpose, with higher performance orientation and self-regulatory capacity as well as early engagement in the course materials [84], [85]. On the contrary, learners with limited self-regulatory capacity seemed to have difficulty when challenges were faced without any additional support [87].

2.7.4.2. Engagement Trajectory Patterns

Cluster analysis indicates obvious patterns for engagement associated with learner success. Final-assessment-takers in MOOCs had structured engagement; their sessions were more organized with respect to session lengths, they consistently took quizzes, and they actively engaged in forum discussions [85]. These were certainly not evident in the weaker students. Overall, these patterns are indicative that cognitive engagement of this nature is necessary for purposeful design of learning activities.

2.7.4.3. Learner Profile Taxonomies

Patterns from systematic profiling included categories called Browsers, Self-Assessors, Serious Readers, Active Independents, and Active Socials, which indicate different behavioral focuses. Dragging the analysis forward [88], [89] multi-course analysis demonstrates the predictive capacity of engagement trajectories for learner achievement [85].

2.7.4.4. Demographic and Cultural Variation

MOOC participation crosses different geographies and demographics like age, region, and race create differences in behaviours. For example, certification averages vary significantly on a country-to-country basis, and inequalities with access to logical, cultural, and digital participation may offer varying engagements [87]. In some places, female learners contribute fewer participants in STEM MOOCs but have generally similar completion rates to males.

2.7.4.5. Profiles and Recommender Design

Profiling learner subpopulations helps shape adaptive learning pathways. To illustrate, clusters from FutureLearn MOOCs have been used to inform support and recommendations[90]. MOOC learners are a highly varied group in terms of motivation, engagement styles, and demographics. Profiling these differences is important for developing personalized learning paths, adaptive interventions, and useful impact-driven designs for our online learning platforms. In turn, understanding and accommodating learner subgroups like browsers, self-assessors, and active social participants will help make MOOC design more inclusive, effective, and learner-centered while decreasing dropout, and improving completion and satisfaction.

2.7.5. Learning Analytics and Log Data Mining

The rapid development of digital education platforms, specifically MOOCs (Massive Open Online Courses), has created massive opportunities for understanding learner behavior through Learning Analytics (LA) and Educational Data Mining (EDM). LA and EDM are complementary fields that use data-driven methodologies to explore the vast and extremely complex data sets associated with learners' digital interactions in digital learning environments. Learning Analytics involves the measurement, collection, analysis, and reporting of data about learners and their contexts for the purpose of understanding and optimizing learning, and the environments in which it occurs [91]. While Learning Analytics focuses on measurement and reporting as part of the learning experience, Educational Data Mining emphasizes the development and application of computational methods [92] for the purpose of discovering patterns in educational data from large-scale educational settings [93]. The MOOC ecosystem involves large, heterogeneous populations that are globally distributed, making the development of engagement, content customization, and understanding of at-risk learners even more critical as well as the engagement of pedagogy. Through a systematic analysis of log files, clickstream data, assessment results, and interaction data, LA and EDM provide meaningful advice and indicators for educators and system designers to use based on analytics. This logic can be mapped to inform adaptive learning systems, real-time feedback systems, early warning systems, and improved designs to address the serious issue of dropout rates in MOOCs [94], [95].

2.7.5.1. Types of Learner Data in MOOCs

MOOC platforms produce many different learner data that can be grouped from the type, source, content, and intended use. The types of data described generally will be the baseline inputs for both learning analytics and EDM techniques as demonstrate Figure7. The main types are:

- **Clickstream and Interaction Data:** these are the low-level logs that account for every type of learner interaction; page views, button clicks, video play/pause/ seeking events, time-stamped behaviors, etc. These data help visualize and then track learner navigation. displays and then timelines of engagement with learning materials. These types of data can lead to broader learning patterns that may identify where learner's dropout or identify learners engaged in successful learning approaches [96].
- **Assessment and Performance Data:** included in this are grades on quizzes, exams, peer-graded activities, auto-graded activities etc. These are summative indicators of learning and cognitive development. Such data will over time also help develop longitudinal analyses of learning trajectories and assessments for predictive modeling of student success or failure [97].
- **Forum and Social Interaction Data:** learners in a course interact with one another using discussion boards, peer reviews, and collaborative group assignments. Analyzing this data can support social presence, collaborative engagement, and sentiment analysis [98]. Social network analysis can then be used to identify "central" participants and isolated learners.
- **Video Interaction and Media Engagement Logs:** these logs present granular detail about learner interactions with multimedia content. Indicators such as rewatches,

changes in playback speed, and video drop-off rates can assist instructional designers and those responsible for producing video content, in optimizing learner engagement [99].

- **Demographic and Enrollment Data:** data, including learner's location, gender, prior learning, motivation, and course completion history, benefits learner segmentation and course content personalization purposes. The demographic profile may play a role in learner persistence, participation, and learners' preferences for various modes of learning [100].

2.7.5.2. Educational Data Mining (EDM) Techniques

Educational Data Mining involves extracting meaningful patterns from educational data through a variety of machine learning, statistical, and data science methods. In the MOOC context, EDM techniques can be used for purposes such as predictive modeling, learner profiling, engagement analysis, and content recommendation.

- **Classification Algorithms :** classification models (e.g., decision trees, logistic regression, support vector machines) are widely used to predict learner outcomes such as dropout risk, completion probability, and performance levels based on early activity data[101].

- **Clustering and Segmentation:** unsupervised learning approaches such as K-means, DBSCAN, hierarchical clustering, help identify latent groups of learners (e.g., lurkers, active participants, or perfectionists) to design tailored interventions and learning pathways [94].

- **Sequential Pattern Mining:** this method reveals patterns of frequent sequence of actions which can indicate reasonable learning sequences or could show problematic navigation behaviors which proceed dropout. Sequential models like (but not limited to) Hidden Markov Model (HMM) or Recurrent Neural Network (RNN) can be considered [102].

- **Natural Language Processing (NLP):** NLP-based analyses can be conducted on forum posts, essay responses, and feedback data to gain insights into the learner's sentiment, understanding of topics, and cognitive engagement. There are many techniques including topic modeling (LDA), sentiment analysis, and discourse analysis [98].

- **Social Network Analysis (SNA):** Social Network Analysis (SNA) explores the relationships of learners based on their interactions in forums or group activities. The metrics of degree centrality, betweenness, and closeness help identify influential learners and inform the design distinto or collaborative kangawen collaborative learning context [103].

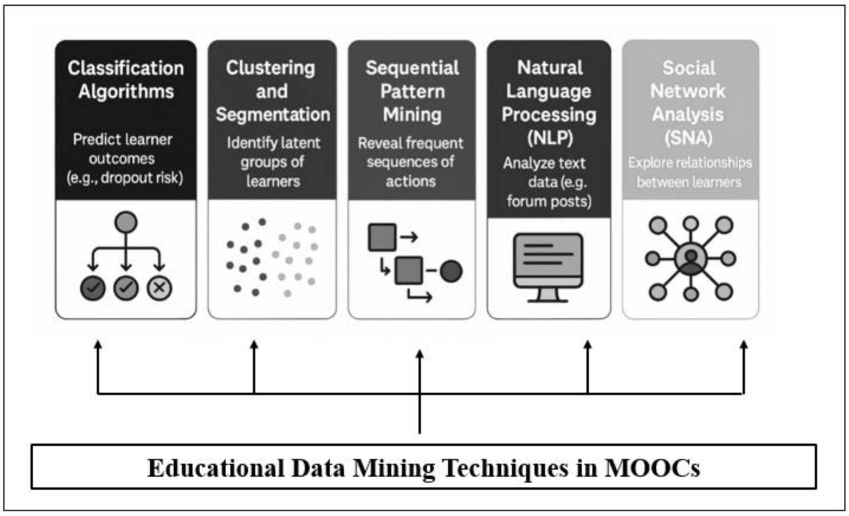


Figure 7: Educational Data Mining (EDM) Techniques in MOOCs.

2.7.6. Key Challenges in MOOCs

Massive Open Online Courses (MOOCs) represent a transformative force in digital education. However, the scope of their global reach and their ability to expand are often offset by enduring challenges that impede the educational effectiveness and learner success of the MOOCs course experience, particularly in the technological, pedagogical, psychological, and infrastructural dimensions (Figure 8).

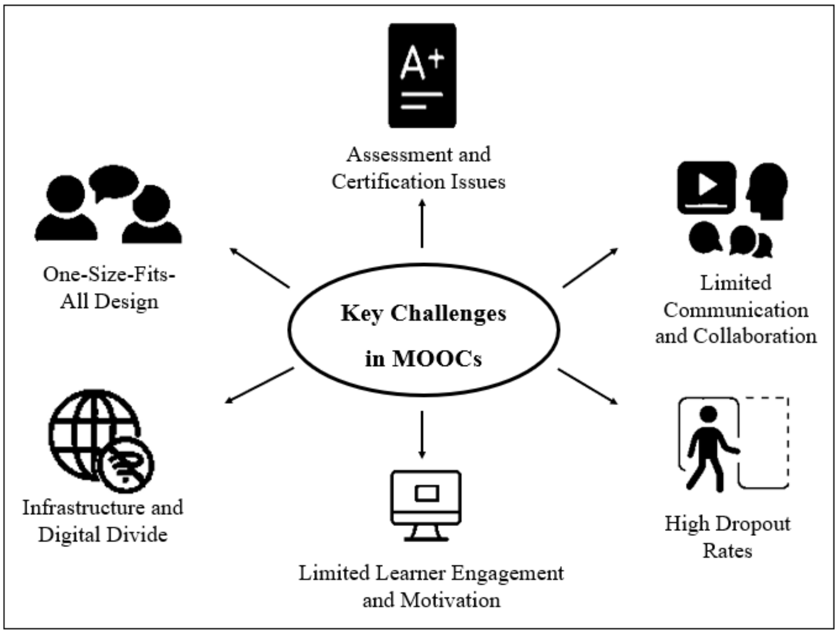


Figure 8: key challenge in MOOCs.

2.7.6.1. High Dropout and Low Completion Rates

One of the oldest and most frequent critiques of MOOCs has been that student's dropout at a very high rate (Figure 9). While millions enroll in MOOCs each year around the world, research demonstrates that fewer than 10% of participants are likely to successfully finish their courses. All of the reasons we cited above lack of face-to-face accountability, no personalized support, unrealistic expectations vs. course demands, all explain, in part, why drop outs happen [104].

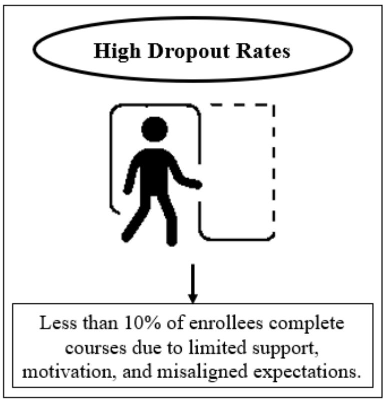


Figure 9*:* High Dropout and Low Completion Rates in MOOCs.

2.7.6.2. Limited Learner Engagement and Motivation

MOOCs are primarily structured with video lectures, auto-graded quizzes, and passive learning structure. These components often do not or cannot promote either engagement or higher-order thinking (Figure 10). The lack of gamification, instructor presence, and interactive features contributes to disengagement and motivation issues for adult learners, who are often faced with competing demands[105].

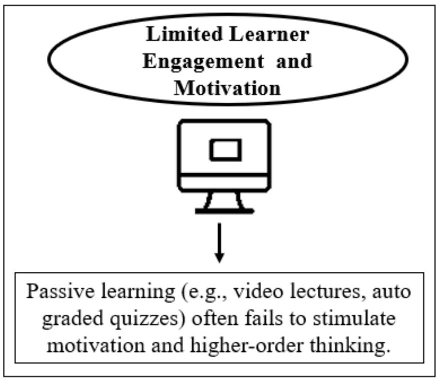


Figure 10: Limited Learner Engagement

2.7.6.3. Limited Communication and Collaboration

Constructivist learning hinges around dialogue and social negotiation, but many MOOC platforms only provide asynchronous discussion forums with limited instructor input and feedback (Figure 11). This limits collaborative knowledge building, which is necessary for peer learning, critical thinking and retention in the long-term [106].

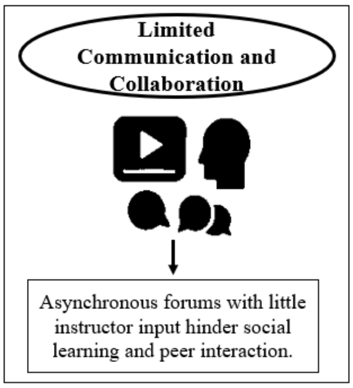


Figure 11: Communication and Collaboration Barriers in MOOCs

2.7.6.4. Assessment and Certification Issues

MOOCs generally require automated or peer-assessment models (Figure 12) to ensure scalability. Although most automated and peer-assessment models deal with their own logistical issues, they frequently reduce the validity or depth of the assessment. Specifically, peer grading lacks reliability, and automated tests tend to be too simplistic for assessing higher-order skills like critical thinking, problem-solving, and creativity[107].

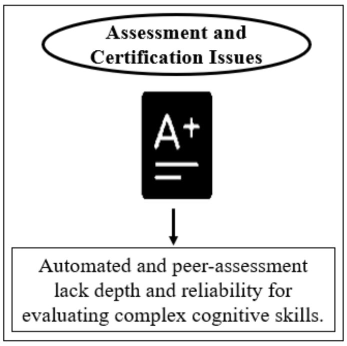


Figure 12: Assessment and Certification Limitations in MOOCs

2.7.6.5. One-Size-Fits-All Design

Typically, MOOCs only provide one model for instruction, regardless of the individual learner's background, learning style, and competencies as they relate to any balance of worldwide learners as shown Figure 13. As a result, learning experiences do not leverage adaptive learning paths, intelligent recommendations, or customization based on learner analytics to personalize the experience[108].

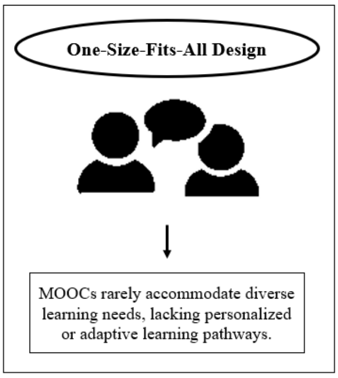


Figure 13: One-Size-Fits-All Design

2.7.6.6. Infrastructure and Digital Divide

While MOOCs aim for a global reach, infrastructure (Figure 14) is an enormous hurdle. Learners from rural or low-income backgrounds often cannot access high-speed internet, have access to computing devices, or lack basic digital literacy, and lack the ability to participate in online education completely [109].

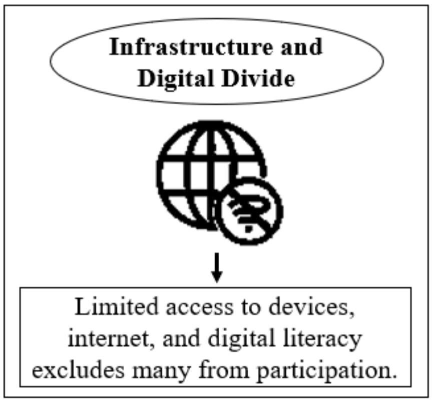


Figure 14: Infrastructure and Digital Divide

2.7.6.7. Language and Cultural Barriers

The ongoing predominance of English MOOC offerings continues to pose significant obstacles (Figure 15**)** for learners who do not speak English, not just in terms of language, but also in terms of culture, as many course materials that MOOCs present reflect Western norms that may be completely out of step with learners from different contexts. Language and cultural dissonance have the potential to escalate the non-relevance, inaccessibility, and disengagement of learners. [110] Articulated the demand for culturally sensitive interventions in MOOC design when surveying 152 MOOC instructors from 25 countries, including recommendations such as multilingual subtitles, simplified language, and support for peer translation; however, most of those efforts were still in their infancy, and unlikely to gain momentum quickly .Likewise, [111] suggested a culturally adaptive design framework that considered localized content and examples of teaching from various contexts more culturally responsive to, and more supportive of, multicultural participation. The Effectiveness of Multilanguage Platforms (EMMA) project [112] opened up new avenues for many MOOC designers, and provided a range of possibilities for multilingual and cross-cultural engagement. In comparison [113],lamented that many MOOCs may still not have design features which engaged the cultural awareness, and entrenched characteristics of global learners who are increasingly diverse. A recent article in [114] reiterates these issues, by stating that while most language MOOCs are trying to incorporate some aspect of interculturalism a few manage to do so with even a culturally comprehensive understanding of academic cultural integration - and the results also indicate the apparently universal absence of broadly designed culturally-inclusive MOOCs.

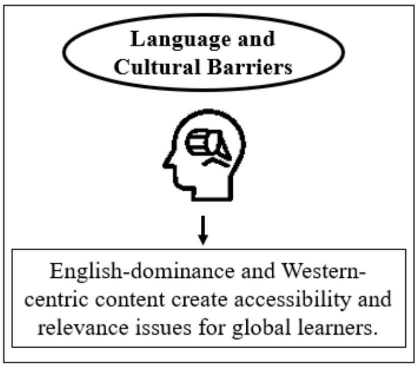


Figure 15: Language and Cultural Barriers in MOOCs

2.7.7. The Need for Intelligent Personalization

MOOCs have substantially broadened access to college education for millions of learners by delivering instructional content flexibly and in a variety of ways. However, the breadth of the delivery model and the diversity of learners present significant issues related to teaching and engagement. Certainly, one of these issues involves the absence of personalisation in large-scale MOOCs and the diversity of the learner community. Learners have multiple cultural, educational, linguistic, and motivational backgrounds, which require some degree of independence, differentiated instruction and support. Without opportunities for personalized attention, learners are likely to become lost in the myriads of resources to meet their needs. Lack of content matches learners' criteria can result in disengagement, and the risk of low retention rates or dropping out of the MOOC experience.

Intelligent personalization in MOOCs is the process by which learners lead their own learning experiences (in whatever capacity they want) by employing artificial intelligence (AI), machine learning (ML), and analytics to adapt, personalize, and tailor learning experiences to a learner's individual characteristics, actions, and preferences. It tries to allow the learner to experience some of the personalization of individualized one-on-one tutoring by modifying and adapting content (presentation), progression and learning pathways, feedback responses, and assessments. From a pedagogical perspective, personalization will develop learner agency, motivation, and cognitive engagement; all key factors for meaningful learning [91]. Furthermore, it overcomes the limitations of static one-size-fits-all educational design by responding to the evolving needs of learners over time as demonstrates Figure 16. One can identify multiple approaches to intelligent personalization in a MOOC system:

- Content Personalization: Recommending relevant videos, readings, or exercises to learners based on their goals, prior knowledge, and what they have already done.
- Pacing Adaptation: Adapting/altering learning pathways based on learners' pacing, difficulty with certain concepts, and time available.
- Assessment Personalization: Producing adaptive quizzes and assignments in response to the proficiencies and misconceptions of a learner.
- Support and Feedback Personalization: Providing in-the-moment tailored feedback mechanisms or support (e.g., chatbot tutors or peer recommendations).
- Interface and Interaction Adaptation: Modifying visuals, navigation pathways or modes of interaction based on learner preferences or accessibility.

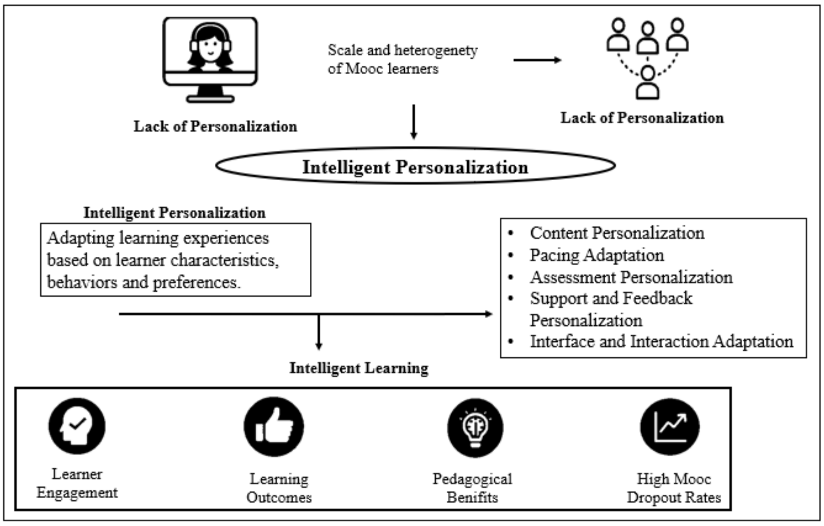


Figure 16: The Need for Intelligent Personalization in MOOCs

Recent developments in deep learning [115], natural language processing and reinforcement learning have allowed personalization engines to more effectively model learner preferences and predict future behavior. For example, recommendation systems [116] based on collaborative filtering or hybrid methods can now offer more precise content suggestions, and affective computing tools can analyze learner sentiment and engagement based on forum posts or clickstream data to reflect when interventions should be altered. The high dropout and low completion rates reported across many MOOC platforms highlight an increasing need for personalization. Research by [94] find that analyzing learner subpopulations can support intervention delivery tailored to them, while [117], finds a relationship to improved persistence when applying a sentiment analysis to forum postings in real time. Moreover, the rise of lifelong learning paradigms, where learners access online platforms at different times to acquire new skills or to reskill, makes personalization an essential element to retaining learners and delivering meaningful content. Smart personalization takes MOOCs beyond levels that are only passive and generalized, to levels of adaptive and learner centric. It may be plausible that this method serving as a bridge could provide a solution to the problem or disaffection gap between massive and meaningful by providing learners not only with content, but also experience that considers the learners goals, context and the way they learn. The need for personalization is further emphasized by the high dropout rates and low completion rates reported on many MOOC platforms. Studies show that personalized learning paths can significantly boost learner persistence and satisfaction. Furthermore, with the growth of lifelong learning paradigms, where learners return periodically to online platforms to upskill or reskill, personalization becomes a cornerstone of effective learner retention and meaningful content delivery.

2.7.8. AI-Based Enhancement of MOOCs

Artificial Intelligence (AI) has transformed its way into the key to improving MOOCs. AI has facilitated adaptive learning, intelligent tutoring, predictive analytics, automatic assessments, learner engagement, and metacognitive support.

2.7.8.1. Adaptive Learning and Personalized Paths

AI-enabled adaptive learning systems provide course content based on a learner's performance, engagement, and behavior. These systems, which use machine learning, neural networks, and decision tree models, are shown to improve performance and engagement indicators. For example, [118] MDPI's 2024 review found that adaptive systems improved test scores as well as the user retention metric.[119], also reported over 90 percent accuracy with employing clickstream data to find learning styles to deliver custom content. [120] also provided a systematic summary of the adaptive learning platform implementations and the typical challenges experienced when implementing them in real-world settings.

2.7.8.2. Intelligent Tutoring Systems and Chatbots

AI-driven Intelligent Tutoring Systems (ITS) and chatbots provide adaptive, personalized tutoring with the use of MOOC conversational platforms. [118] presented the SMIT based ITS system, which combined adaptive testing with continuous feedback, and participant learning responses were positively impacted. [121] provides the best overview of advances in ITS, including real-time adaptation and ethics. The EduChat system [122] uses large language models to support Socratic questioning, evaluate essays, embed emotional support, and yield pronounced benefits in learner trust and retention.

2.7.8.3. Predictive Analytics and Early Warning Systems

Predictive analytics uses learner behavior data, like clickstream, time spent engaged and quiz performance, to provide dropout risk, and ultimately invokes not only timely interventions. Previous investigations have identified the outcome-focussed dashboards built by [123] which offer behavioral indicators of learners to predict the dropout status and completion of the course. [124] comprehensive review reported the predictor models' usefulness in early-alert frameworks.[125] developed M2LADS, which incorporates biometric and behavioral information in a combined multimodal, real-time model of monitoring to provide early proactive support for students deemed at risk.

2.7.8.4. Scalable Assessment Automation

AI has the potential to greatly increase the scalability of MOOCs through its ability to automate the development of content, feedback, and grading.[126] used graph-based deep learning models for grading prediction and found that their model was more precise and grading took less time. EduChat by [122]. used LLM's for generating quizzes/assessments Autonomously and were able to provide personalized quiz inputs. Studies indicate that generative AI fosters metacognition and reduces instructor grading burden through helping students reflect on their cognition[127].

2.7.8.5. Learner Engagement and Support

Utilizing AI chatbots and interactive quizzes increases engagement with the capacity to introduce real-time learning recommendations, motivational messages, and prompts. AI self-regulated learning tools enhance engagement and retention as evidenced by[127]. [128] revealed adaptive quizzing increased student motivation and performance by allowing users to answer questions of mixed degrees of difficulty. Importantly, EduChat was able to provide emotionally intelligent teaching by engaging student feelings which increases student engagement and trust in their learning process[122].

2.7.8.6. Enhancing Self-Regulated Learning

AI fosters self-regulated learning through personalized pacing indications, progress feedback, and goal setting suggestions. [127] believe that AI can provide scaffolding in the development of metacognitive skills through the incorporation of analytics and chatbots. [129] demonstrate that intelligent feedback systems enhance learners' self-monitoring and self-discipline in MOOCs. [130] indicated that AI/ML intention-based platforms enhance learner agency and adaptive content changes.

According to recent research, Artificial Intelligence (AI) is transforming MOOCs' additional quality components like personalized learning paths, intelligent tutoring, proactive intervention, and automated assessment, engagement, and self-regulated learning, to name just a few. These components of technology will help MOOCs take on some of the primary challenges facing MOOCs dropout rates, engagement, and providing generic experiences - into rich multi-dimensional learner-centered environments.

2.7.9. Current Trends and Future Directions in MOOCs

The landscape of MOOCs is changing rapidly, influenced by technology, credentialing changes, and shifting learner needs. Here are the current trends shaping the future of MOOCs, with each topic reflecting new priorities in online teaching:

2.7.9.1. Artificial Intelligence, Personalization & Adaptive Learning

AI-enhanced personalization offers a new dimension for MOOCs, as it permits adaptive learning pathways of instruction which can be adjusted in real-time. As Moocs4Gamification highlighted recently, adaptive learning platforms have enhanced engagement, performance, and retention as they personalize content and assessments in real time through machine learning techniques. For example,[131] [132]investigated the educational principles and implementation issues of AI-enhanced adaptive systems. [119] investigates learning styles of learners through analysis of their clickstream data and reports high accuracy in identifying educational styles through clickstream indicators, permitting content focused on learner styles. [133] analyzes AI personalization and offers insight into the future of personalization and the impact of AI on the higher education sector.

2.7.9.2. Microlearning & Micro-Credentials

The rise of micro-credentials show that the world of learning was becoming more modular and skills-based.[134] explored the inclusion of accredited micro-credentials into traditional types of MOOCs, and explained how those micro-credentials can create stackable learning experiences. [135] looked at countries in how micro-credentials can drive adoption and provide policy directions. [136] which presented evidence of micro-credentials and how the policies intent in the unbundling of qualifications were emerging.

2.7.9.3. Blended & Immersive Experiences

Increasingly blended and immersive MOOC models have used virtual reality (VR) to provide more interactivity and experiential learning. [134] outlined the possible opportunities and technical issues with immersive VR in online learning environments. [135] evaluates a pilot MOOC where VR was used to facilitate collaborative learning. In addition to this [136] evaluating hybrid courses with VR and mixed reality examples using VR to facilitate fintech education, demonstrating better acquisition of skills.

2.7.9.4. Globalization and Localized Content

To expand access, many MOOCs are offered in community languages and culturally fitting ways. While there is not a great quantity of specific quantitative studies, policy analysis and platform reports do indicate a comprehensive trend in the growing field of multilingual MOOCs [137]. Platforms like Coursera and edX are expanding their vernacular offerings to reach previously unreached learners around the world.

2.7.9.5. Learning Analytics & Data-Driven Design

Learning analytics is being used for an increase in pedagogical improvement and course redesign. [138] details how data-driven insights are influencing course design decisions for MOOCs. Meanwhile, [139] examined the use of analytics in distance education, and noted its use for surfacing learning bottlenecks and improving learner success rates in terms of outcomes. Finally, [140] provided evidence of closed-loop, learning design, while implementing decisions tracked by analytics.

2.7.9.6. Corporate Partnerships & Upskilling Pathways

MOOCs are increasingly collaborating with corporations to provide workforce upskilling programs [141]. Coursera's enterprise offerings are an example of this increase as they leverage AI-curated learning paths aligned with one's career objectives [142]. [143] reflects strong institutional interest in corporate micro-credential programs rooted in university-industry partnerships. [144] conducted in-depth interviews with HR professionals in Europe and highlighted the benefits and barriers of adopting MOOCs for employee training.

2.7.9.7. Interactive Features & Gamification

The incorporation of gamified elements in MOOCs increases interaction among the students.

[145] assessed gamification in MOOCs with the Octalysis framework and saw notable improvements in engagement levels. [146] indicated that scoring, badges, and simulation of real-life tasks improve levels of motivation in MOOCs and lower the attrition levels. [147] discussed motivation and engagement with points and leaderboards as additional layers of tracking and social interaction.

2.7.9.8. Sustainability and Ethical AI

Increasing apprehension related to sustainability, equity, and ethical use of artificial intelligence in MOOCs is now once again coming to the forefront [148]. signals a developing academic and institutional engagement with responsible AI and equitable digital learning practices. Similarly, MOOC reflects increasing public discourse on the ethical implications of artificial intelligence in education [149]. Apart from the need for fairness and transparency, it is imperative to ensure safe and secure AI powered learning platforms to establish a reliable trust-centric environment. Hybrid models of AI based intrusion detection systems has shown great promise regarding the protection of intelligent systems from cyber-threats while ensuring satisfaction and reliability [150].

In combination, these advances suggest that MOOCs are about to undergo a shift from massification into personalized, rich, accessible and equitable learning experiences. AI-powered adaptivity, modular credentialing, access with a global reach, analytics to inform design, corporate upskilling, interaction, gamification, and commitment to ethics are all coming together to reshape online learning into the future.

2.8. Conclusion

Our examination of e-learning systems and MOOCs in this chapter demonstrates a sophisticated and complex development of digital education. There is now greater sophistication in the technologies, pedagogies and strategies for learner engagement. The role of Artificial Intelligence in this transformation is fundamental, and it alters how content is delivered, personalized and assessed. AI can provide adaptive paths in learning, intelligent tutoring systems, predictive analytics, and engagement-enhancing systems that begin to offer solutions to limitations that have plagued large scale online learning approaches for years - most notably low retention, personalization and interactivity. The chapter also highlighted the layered architectures in MOOC platforms, and the differences in learning profiles of learners, as well as the different pedagogical models that support learner autonomy, motivation, and engagement. These are important insights that point to the necessity for scalable, flexible, and learner-centered systems that appropriately respond to various educational demands. The addition of learning analytics and data mining techniques has opened new doors for level, feedback-driven instruction, real-time data-driven instruction, and design for evidence-based practice, moving MOOCs along the pathway to transformative educational systems.

Overall, this chapter's development illustrates a paradigm shift in terms of education, primarily in its nature and use, in the digital era. Education is not limited to learning or e-learning nor are MOOCs external to education, what remains to be seen is the level of innovation, ethical considerations of AI, and an understanding of heterogeneous learners and learner behaviours. With a historic development trajectory, major aspects, pedagogical origins, and technical enablers being presented as the environment and evolution, this chapter has set the baseline for enacting the retitled- empirical-materialized characteristics of MOOCs, with an explicit review of the current platforms, technologies, and research themes that make intelligent, scalable, and personalized online learning a reality.

Chapter 2: State of the art of MOOCs

3.1. Introduction

The rapid progression of Massive Open Online Courses (MOOCs) has stimulated digital education, providing scalable and accessible learning for many audiences worldwide. The initial excitement about MOOCs has now matured into a more critical and insightful view of their values and limitations, as well as their potential for intelligent improvement. In this chapter, I am going to review the current state-of-the-art from MOOCs, by organizing discussion as it relates to pillars that support intelligent learning environments, from both technological and pedagogical perspectives. The foundation of contemporary MOOCs is the relationship between artificial intelligence, learning analytics, personalization techniques, and semantic technologies to support more adaptive, engaging, and evidence-based learning. While many MOOCs remain repositories of learning content, MOOC platforms are moving toward becoming intelligent and learner-centric ecosystems capable of supporting individual pathways in education through technologies. To systematically map the various research approaches, this chapter will outline these approaches in terms of four primary axes that highlight both methodological advances and application depth: course structure classification, personalized recommendation systems, learner behavioural implications, and semantic course retrieval. Each axis highlights a wealth of research from traditional machine learning approaches through to novel deep-learning and hybrid models, highlighting how the advances in research have addressed the major challenges facing MOOCs such as learner drop-out, low completion rates, lack of personalization and efficient search and discovery. The first axis focuses on advances made in course structure classification, with an emphasis on future uses of hybrid knowledge-based systems and transformer architectures for educational content classification, classifying content into a fine-grained organization and delivering it in a relevance aware manner. The second axis encompasses personalized recommendation in MOOCs, outlining the broad range of approaches, from collaborative filtering to reinforcement learning, that seek to improve learner user engagement and learning outcomes through personalized content delivery. The third axis considers the modeling of learner behaviors important for real time interventions and dropout predictions, although, motivation modeling and feedback mechanisms increasingly leverage multi-modal AI and explainable analytics. Finally, the fourth axis considers enhancements to semantic course retrieval systems, where we witness a powerful advancement from keyword-based search systems to more context-aware and ontology-based retrieval systems, both of which better support the learner's intention to access content. By synthesizing the past and current influential works across these axes, this chapter reveals emerging trends, issues yet to be resolved, and future directions for researching MOOCs. This chapter also sets the stage for more inquiry into the use of intelligent technologies in learning at scale and provides the foundation for the following chapters focusing on new system design and intelligent enhancement of MOOCs.

3.2. Related Works

To enable a coherent comprehension of the research landscape about the MOOCs and intelligent learning environments understanding, we organized related works around the main axes of contribution of our study. Specifically, we organized the literature into the course classification, personalized course recommendations, learner behavior analysis, and semantic course retrieval. This organization reflects our methodological and application research areas, allowing for a systematic comparison and highlighting our contributions to close existing gaps in the current research [151].

3.2.1. Course Classification

The task of course classification in MOOCs requires automatically categorizing a course or course-related content (e.g., titles, descriptions, discussion posts, or feedback) into predefined categories that may include subject domains, cognitive levels, learner intent, or urgency. Classification adds value for many downstream applications, for example recommendation systems [152] , course search, adaptive learning systems, and learner support systems. Over the years, many different methods to complete this task have been developed including traditional machine learning, neural networks, transformer architectures, and hybrid or optimization-enhanced models.

3.2.1.1. Machine Learning-Based Approaches

Earlier investigations of MOOC course classification mainly relied on traditional machine learning methods, still heavily relying on hand-crafted features, and on structured course metadata (e.g., titles, descriptions, or tags made by instructors). Common classifiers used included SVMs, Decision Trees, k-NN, and Naïve Bayes, and models frequently relied on bag-of-words, TF-IDF, or other domain specific features. For example, in [153] authors describe and recommend a semantic rule-based model that relied on OWL ontologies, and was documented in a SPARQL query system, to support the classification of MOOC courses. Their

semantic model was designed to ensure coherence between prerequisite courses and learning objectives by representing "semantic knowledge" during course classification. Structuring education content as an ontology allowed for a more systematic and logically coherent representation of a course based on the educational objectives and the relationships between topics. This was potentially useful for aligning instructional design and learner progression.
In a similar vein, the paper in reference [154] used an ontology-driven model that implemented collaborative filtering and learning analytics to support a competency-assessment and learner-centered education approach. The model classified MOOC resources in real-time through user interactions and input provided by learners about their learning path and preferences, thereby informing the system about the adaptations that need to be made. The model improved upon just classification and offered useful recommendations for learners associated with the system. This study coupled ontological modeling with user-behavior analytics and demonstrated some of the possibilities of more traditional machine-learning approaches with more semantic and learner-aware facets to the models. The authors in [155] created personalized e-learning situations, as well as MOOC recommendations, within a smart education ecosystem, while also incorporating contextual data received from the Internet of Things (IoT) and real-time feedback from learners. Their machine-learning-based framework was designed to accommodate environmental factors, device behavior, and learner feedback on interaction to offer richer input features for classification as well as recommendations. The system was able to classify and propose courses based on the learner's context, need, and preferences. The findings from the study noted some potential benefits of pairing traditional algorithms with rich contextual signals and provided a promise that current intelligent sensing capabilities improve MOOCs classification tasks.

Most recently, in [156] the authors compared a series of classical machine learning algorithms (including Random Forest, Logistic Regression, and SVM) across many different feature types (linguistic, statistical and semantic) to classify the MOOCs in Bloom's Taxonomy levels. Results showed that even traditional classifiers can provide competent results when tied to rich and diverse features. In [157], a TF-IDF and gradient boosting classifier pipeline was proposed to classify MOOCs by domain area, based on large-scale metadata from open learning platforms. The classifier produced interpretable and high-performance classifications, with relatively low computational costs, making it suitable for a real-time educational platform. These traditional methods have provided interpretability and less computational expenses than deep learning methods but often come with scalability concerns and concerns about the reliance on domain-specific feature engineering; thus, the reason to explore deep and hybrid learning methods as described in the next sections.

3.2.1.2. Neural Network-Based Models

Researchers responded to the challenge of feature engineering by employing deep learning models that could learn latent representations of raw text. Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN), which include LSTMs and BiLSTMs, became what most researchers and practitioners opted to utilize. In [158], a hybrid CNN-BiLSTM was proposed for the classification of MOOC forum posts into relevant course-related code, such as pedagogical for some concerns, instructor for instructor-related concerns, technical for tech-related concerns among others. The CNNs classified local word-level features while the BiLSTM was utilized to classify the sequential semantics [159]. There was shown to be an improvement in classification performance with noisy and unstructured forum data; however, the models still did not have a global context. Deep learning models established the foundation for later larger models, such as transformer-based approaches.

Expanding on it further, a CNN–LSTM transfer-learning framework called ConvL was proposed in [160] to automatically identify confusion, urgency, and sentiment polarity in forum posts across different courses. To this end, ConvL employed a CNN to represent the local features in a post and an LSTM to model the temporal semantic relations in posts and adopted cross-domain fine-tuning to improve generalizability to new courses. Another study [161] reported a dual-channel deep attention-based architecture called DAPDM that combined BERT embeddings with a CNN + BiLSTM + attention mechanism to mine sentiment and topics from course reviews. This model improved the semantic representation and provided strong performance in sentiment and topic extraction tasks in large-scale MOOC review datasets

A model [162] used ensemble CNN–LSTM architectures to provide early warning of dropout in MOOCs, where CNNs were used to extract features from learners, and the LSTMs modeled the temporal progress of the learners with respect to activity as a time-series. Although ensemble and single-architecture approaches were assessed, the ensemble models outperformed single architecture methods and provided at least some early warning signals in the learner retention analytics. These neural network models demonstrate that by combining CNN and RNN layers and potentially including attention mechanisms [163], [164] or transfer learning from course to course could have significant advances in classification of content and behavioral characteristics in MOOCs and supersede some context and data imbalance challenges.

3.2.1.3. Transformer-Based Architectures

The emergence of the transformer-based model, and especially BERT ("Bidirectional Encoder Representations from Transformers") and its variants, represented a paradigm shift in natural language processing for MOOC tasks. Whereas traditional word embeddings represented a single word value for each word in a category, BERT presents multiple values based on other words found on either side of the embedding. BERT takes this a step further by using deep learning based on a bidirectional context, so it is inherently better for the subtlety, hierarchy, and ambiguity often found in MOOC language, for instance. As described in [165], developed a fine-tuned BERT model for the cognitive and pedagogical classification of MOOC learning objects. The classification determined Bloom's Taxonomy levels for the labeled content which is based on how cognitively complex and pedagogically motivated the content is. The results showed how robust BERT is for generalizing across many MOOC platforms and course types, and that it required very few labeled data to perform well, demonstrating potential as a flexible model for a large-scale workforce in the educational sector. In a similar vein, in [166], the authors introduced a BERT-BiLSTM-Attention model to analyze the learner reviews from the online space. The study had the objective of classifying these learner reviews based on sentiments and intended topic, to build better understanding of student perceptions and experiences. Through aggregation of BERT contextualized embeddings, BiLSTMs sequential experience, and attention mechanisms' human interpretability, BERT-BiLSTM-Attention was able to classify nuances of emotional and thematic ideas in learner reviews more accurately than non-hybrid systems.

In [167], the authors integrated BERT embeddings into an LSTM-based system that classified learner experiences in MOOCs in relation to learning outcomes of the course. The goal of the system was to connect the learning outcomes intended through the educational design, with reports from learners, to assess the effectiveness of instructional design and instructional practices in a repeated course. A positive effect of such pedagogy was to compare subjective inputs provided by learners against the objective outcomes specified through curricular goals. The paper in [168] addressed the challenge of real-time learner support through a four-stage pipeline based on weighted BERT features fused with multi-channel CNNs for urgency detection of forum posts. The classification of urgency was a key factor in how instructors or facilitators could reply to critical posts more quickly. In generalized terms, both demonstrated a hybrid model that outperformed traditional methods of distinguishing between urgent and non-urgent requests (then BERT greatly increases the feature space, while CNNs help with spatial learning). Lastly, in [169], the authors applied BERTopic, a novel topic modeling framework that utilizes BERT embeddings and explores clustering methodologies to detect urgent discussion topics situated in MOOC forums. Compared to conventional topic modeling techniques such as LDA and NMF, BERTopic clustered topics in a more coherent and defined manner, which improved the relevance of identified topics of urgent discourse. This increases responsiveness of instructor interventions for discussions involving time-sensitivity. Overall, these studies demonstrate the innovative capacity of applying BERT-based architectures for improving classification, feedback analysis, urgency detection [170], and thematic analysis in MOOCs. The foundations have been established for the design and development of more responsive, intelligent e-learning environments.

3.2.1.4. Hybrid and Optimization-Enhanced Models

Recent developments primarily concentrate on integrating various modeling paradigms for improved classification performance and interpretability. Commonly explored models integrate transformer encoders with attention layers, optimization methods (e.g., reinforcement learning), or external semantic structures (e.g., knowledge graphs), and use various approaches to enable hybrid models. MOOCCubeX, described in [171], is an example of this type of approach. It is a large-scale knowledge centered repository built through weak supervision and BERT based concept extraction. MOOCCubeX builds a fine-grained concept graph (with more than 600k concepts mapped to course resources and learning objectives) and allows co training over labeled data and unlabeled data. This infrastructure enables adaptive, personalized classification of MOOC content to pedagogical goals. Empirical results show that hybrid models developed this way consistently outperform non-hybrid approaches through their structured interpretability, generalize well across domains and across the structure.

Another recent framework combines GCNs with reinforcement learning for the purposes of knowledge aware classification and decision making in MOOC platforms. In [172], the authors created a heterogeneous knowledge graph that contained entities such as courses, concepts, and users, which they combined with Graph Convolution Networks to generate meaningful embeddings. They then input these embeddings into a reinforcement learning policy that decided on the appropriate classification, which aimed to personally align learners with the appropriate course content. Because the decision was based on prior graph determinacy, the graph-based reasoning paths could provide explanations for recommended or tagged resources that referenced explicit paths in the related knowledge graph. The third hybrid approach involved knowledge graph enhanced BERT styles in conjunction with reinforcement learning for classification tasks. In [173], they proposed a knowledge-guided ensemble model that amalgamated structured knowledge from cousins (such as Concept Net and Wikipedia) with a BERT-based model. For those decisions, reinforcement learning was employed to optimize the various weights associated with the different knowledge inputs for both classification accuracy and semantic coherence. This approach also attempted to manage the information noise and reliability requirements that are problematic in interpretive domains such as MOOCs.

Table 2 offers a more systematic perspective on recent developments in MOOC course classification and semantic tagging. It compares a selection of state-of-the-art methods, each having unique limitations on scalability, interpretability, semantic coverage, and adaptability. The table gives a summary of the main contributions of the methods presented, what challenges they overcome, and their performance compared across dimensions. The comparison provides potential for highlighting trends in the field, and the evolution of transformer-based and ontological models.

Table 2: Comparative Table of Recent Methods for MOOC Course Classification

Method	**Year**	**Resolved Problem**	**Gap Resolved**	**Accuracy**	**Scalability**	**Interpretability**	**Resource**	**Adaptability**
FT-BERT [165]	2022	No scalable MOOC classification	Transformer-based automatic classification	High	Medium	Medium	Medium	High
BERT-BiLSTM-Attn [166]	2025	Manual MOOC course labeling	Handles nuanced cases, boosts classification	High	Very High	Medium	Medium	Very High
NN-LSTM+BERT [167]	2025	Limited semantic coverage & content update	Deep semantic coverage for adaptive scaling	High	High	Low–Medium	Medium	Very Good
BERT-Attn [168]	2025	Inefficient human review analysis	Automates tag review via enhanced NLP	High	Medium	High	High	Very Good
BERTopic [169]	2024	Difficulty in urgent post detection	Improves topic coherence and urgency detection	High	High	Good	Medium	Very Good

BERT-CNN [158]	2023	Low engagement, no urgency detection	Balances urgency detection in imbalanced data	High	Medium	Medium	Medium	Good
Ontological Model [153]	2023	Course prefer inconsistency	Verifies course structure & detects gaps	High	Moderate	High	Moderat e	Moderate
Ontology + Reasoning [154]	2024	Static curricula, no adaptation	Enables personalized paths via reasoning	High	High	Medium	Medium	High
Concept Extraction [171]	2021	Short text concept extraction	Multi-stage weakly supervised extraction	Moderate	High	High	Moderat e	Moderate
AI-based SEO [155]	2023	Poor on-page optimization	Enhances indexing through intelligent methods	Medium	High	High	Medium	Low

3.2.2. Personalized Recommendation in MOOCs

As MOOC platforms continue to grow, personalized recommendation systems are becoming even more critical in plausibly improving learner engagement, retention, and academic performance. There are many approaches to creating personalized systems, diverse from traditional collaborative filtering to neural networks, reinforcement learning, and hybrid systems, that address the personalization problems.

3.2.2.1. Collaborative Filtering and Matrix Factorization

Collaborative Filtering (CF) remains one of the most widely used and successful techniques in recommendation-based systems, especially in MOOCs when it comes to developing educational content for learners based on their interests and behaviors. CF uses previously recorded user interactions or ratings, such as when learners rated courses and how much they interacted with the courses, to identify similarity between users, or items (e.g., courses), and use it to predict preferences, without requiring explicit semantic information about the content. The authors in [174] provided a user-based collaborative filtering already to elicit learner preferences based on the learner interactions such as course ratings, and learner's behavioral data. The authors provided evidence that systems can adaptively improve recommendations over time and evolve personalization, with CF acting as a model of learner interest across categorical boundaries in MOOC content.

Nonetheless, it appears that the issue of the sparsity of interaction matrices remains a common problem with CF recommendation models, especially in educational contexts at scale. In response to this problem, [175] proposed a hybrid recommendation model, known as "Co-SVD" which integrates collaborative filtering and Elastic Matrix Completion methods. Their hybrid approach aimed to mitigate the effects of sparse data by creating a method that can complete the missing entries of users in an user-item matrix. Not only did this improve the quality of recommendations when there was a lack of historical data (i.e., sparse) cases, it also provided improved scalability and robustness in cold-start recommendations when there is limited user or course data is missing.

In addition to presentation of collaborative methods, [176] presented a machine-learning based recommender system which included learner motivation as a key attribute in the recommender engine. Their system used Random Forest classifiers to model the level of learner engagement in courses, following appropriate recommendations based on learner engagement levels. By including learner motivation as an attribute, their recommender system adds a behavioral and psychological characteristic to learner recommendations which traditional CF methods fail to capitalize on. [177] provided a more comprehensive analysis of the potential contribution of collaborative filtering within MOOCs by conducting a survey comparing collaborative filtering, content-based filtering and hybrid filtering. Their results described characteristics of CF methods, particularly in the identification of learner preferences and identifiable trends through peer-collaboration. At the same time, limitations were also discussed, namely cold-start issues, sparsity of data, and inability of CF to represent content or written elements, which call into question the quality of the recommendations if left unaddressed and without hybridization or alternative learning opportunities. The applications of collaborative filtering with modified, advanced matrix factorization methods, predictive machine learning frameworks and hybrid recommender systems are a promising approach to the personalization of MOOC. Primary, not only does this approach address known issues with data, but they also contribute to improving the quality of learning engagement and learner-centered support in educational contexts.

3.2.2.2. Neural Collaborative Filtering and Deep Recommenders

The emergence of Massive Open Online Courses (MOOCs) has opened many doors to learning, but it has also raised several issues around content overabundance, learner engagement, and tailored navigation. For these reasons, neural collaborative filtering (NCF) and deep recommender systems represent powerful solutions that structure the interplay of complex learner-course interactions using the deep neural architecture. In this scenario, [178] proposed a deep neural recommendation framework using K-means clustering to differentiate synchronous and asynchronous learners. This hybrid clustering-recommender pipeline accounted for the temporal engagement patterns learners established and contextual learning behaviors, which led to more tailored and useful course recommendations around learners' scheduling preferences. The authors noted that embedding learner behavior into latent feature spaces and clustering similar learner vectors allowed systems to directly adapt to different pacing and content requirements. Inspired by the drawbacks of static rankings, [179] proposed a MOOC recommendation service called MHRR (Meta Hierarchical Reinforced Ranking). MHRR combines reinforcement learning and a hierarchical model to iteratively update ranking policies and improve user learning paths and context-dependent content preferences. The use of a meta-learning layer improves sample efficiency and generalization across different learner groups. The reinforcement learning element of MHRR allows it to learn the 'best' sets of recommendations through feedback simulation from hypothetical learners, appropriate for ongoing dynamic learning environments.

In terms of sequence modeling, [180] presented a hybrid model, integrating CNN and RNN architectures, to extract learner traces from cloud-based educational data. This hybrid model provides the ability to model both spatial features of learners (through CNNs) and temporal dependencies (temporal behavioral patterns through RNNs). Combining these approaches serves a dual purpose: to model both the content that students are exposed to, and the context of how or when they interacted with that content - possibly yielding intervals of activity, revisiting topics, and some retention of momentum. In a similar manner, [181] used a hybrid CNN-LSTM approach to dropout prediction for STEM-oriented MOOCs. They used a time-aware architecture which conceptualized learning sessions as a sequence of behavior snapshots over time. The model identified sequential dependencies and erratic activity patterns (e.g., number of interactions with video drop off over time or quizzes skipped) that could trigger early intervention programs. The model ability to learn from past experiences was improved by LSTM-based temporal memory and convolutional extraction of features, providing the means to model the longitudinal experience of learning engagement.

In another study, [182] introduced neural-based Named Entity Recognition (NER) into the recommender system's search engine [183] on a MOOC platform to increase semantic understanding. The researchers concluded that when a MOOC entity was identified and contextualized as a named entity (i.e., course topics, skills, organizations) it would improve both index and search recommendations. Using NER allowed them to tag courses with granularity which aided in the enablement of concept-based personalized search and real-time adaptive filtering. For generative and retrieval-based improvements, [184] introduced GeAR (Generation Augmented Retrieval), a framework that combines generative retrieval with embedding-based ranking. This model will allow for greater semantic coherence in course recommendations by generating informative queries and candidate representations for a candidate learner's evolving interests. GeAR provides a strong framework for exploiting sparse or noisy profiles of learners, which is a typical situation in a MOOC environment with a very diverse population of learners.

Lastly, [185] proposed a multi-architecture recommender that includes BERT, TextCNN, BiLSTM, and attention for dropout prediction and content recommendations. In this case, the model develops dropout prediction scores and recommendations for reinforcement materials based on learners' histories of performance and their sequential learning paths. By encoding all textual data with their learning sequences and context, this model provides a very personalized learning experience based on predictions of learners' behaviour. Deep neural recommenders that leverage CNNs, LSTMs, attention, NER, reinforcement learning, and semantic models will provide valuable personalization, temporal modelling, and user profiling functions in MOOC settings. Such models not only improve prediction accuracy but also develop adaptive learning systems that respond to the learners' changing needs, intentions, and behaviours.

3.2.2.3. Context-Aware and Knowledge Graph-Based Recommenders

The use of semantic technologies and contextualization in recommendation systems has led to better personalization experience in MOOCs. These modern models include learner profiles, learning modalities, and contextual or meta data about the learner and their environment, dynamically yielding content recommendations. For example, the hybrid intelligent framework described in [186] combined a fuzzy FSLSM (Felder-Silverman Learning Style Model) with semantic web technologies, such as OWL ontologies to formally describe learner preference, course metadata, and pedagogical relationships, and SWRL (Semantic Web Rule Language) to encode domain knowledge and learner behavior. By allowing for personalized inferences based on queries via RDF-based knowledge graphs using SPARQL, this system was able to recommend courses better suited for an individual's cognitive style and instructional modalities. Subsequently, [187] introduced a semantic course recommendation model based on ontological mappings of learning trajectories through prerequisite structures, topical dependencies, and learning outcomes represented semantically. The model was able to identify learner prior knowledge, then recommend learning pathways through reasoned semantic queries, greatly enhancing the pedagogical coherence of the recommendations. Likewise, [188] explored how semantic models could help in visualizing learning journeys. It provided an adaptive feedback mechanism that visualized the learner's journey in real-time against the semantic representation of the course structure that informed inferences of the learners' gaps in understanding and generated the content to address the identified gaps in real-time. Feedback mechanisms were also improved with gamification strategies as found in [189]. The study presented a feedback loop in real-time through gaming elements like badges, levels and challenges along with a semantic representation of learner's activities. This feedback mechanism was effective at increasing engagement and could present real-time personalized content based on the learner's semantic profile by adapting the difficulty and sequencing of the course content based on how the learner's activity is semantically mapped to their learning activity.

The impact of learner personality types on recommendations was also incentive in [190]. This work operates MBTI personality types with semantic enrichments using the WordNet database. In this case, mapped the user personality type, they provided contextualized recommendations about content, that represented the emotional and cognitive conditions of the learner learners. To acknowledge diversity in learner backgrounds, [191] and [192] presented systems which were multilingual and aware of location. These systems employed multilingual ontologies and geospatial metadata, demonstrating endogenous accessibility to localized, linguistically appropriate content. Thus, learners receive recommendations that are potentially aligned with their topic of interest, as well as their cultural and language contexts and reduced cognitive entry hurdles. Lastly, [193] identified visual personalization in its advanced work on institutional onboarding [194]. This incorporated a generative adversarial network (GAN) to adapt course thumbnails and the utilize learner-inferred preferences to influence UI designs. Alongside, [195] presented a framework based on content analysis of video and engagement prediction. The video implementation included facial expression detection and visual interaction detection [196], which would be used to adapt course sequencing and pacing. Together, these contributions represent a key advancement towards intelligent, learner-centred

MOOC platforms. When combined with ontological reasoning, psychological models, and real-time learner context, they describe a step towards more humane, accurate, and valuable learning recommendations.

3.2.2.4. Hybrid Recommendation Systems

In the MOOC ecosystem, hybrid recommendation systems harness the potential of complementary hybrid AI models to counteract the disadvantages of separate components and are emerging as a powerful solution in this space. Hybrid recommendation systems improve flexibility, personalization and performance through a combination of collaborative filtering, content-based filtering, reinforcement learning, and explainability frameworks. In [197], offered a hybrid explainable recommendation model that incorporated K-Nearest Neighbors (KNN) with Matrix Factorization (MF) and a Lime (Local Interpretable Model-Agnostic Explanations) model. They illustrate the growing need for recommendations to provide transparent decision-making models instead of a black box of recommendations. In their model, along with performing the recommendation process, explanation is focused on decision making so that the learner understood the course recommendation. In educational online spaces, the explanation would help establish trust in recommendation systems. The case studies of [198], extend the idea of hybridization of recommendation models by including multimodal data i.e. course text, lecture video, and user interaction logs. They describe a hybrid ensemble of deep learning architectures to harvest commonalities between multiple channels of heterogenous data streams. They demonstrate that multiple content streams ultimately allow for deeper personalized course recommendations; responding to learners' not just on explicit preferences but usual learning patterns inferred from clickstream behaviour and video viewing engagement. [199] focused specifically on real-time behavioral analytics where they implemented machine learning techniques in their case, ridge regression and recursive feature elimination to construct a hybrid system able to analyze learners' in -course behaviors in real-time. The model allows just-in-time interventions by predicting when learners are languishing/stuck/disengaging and providing recommendations for courses or resources in real-time. The addition of reinforcement-learning (RL) has the potential to bring a new level of adaptivity to hybrid systems. [200] provide an approach for improving MOOC search and recommendation algorithms' ability to discover suitable courses by detecting key elements in course descriptions and user queries using Named Entity Recognition (NER). This enables the recommendation algorithm to better grasp what users are seeking for and the content of each course, yielding more accurate, relevant, and personalized course recommendations than the traditional keyword-based technique. In [201], the authors modeled MOOC learning trajectories as Markov Decision Processes (MDPs) and applied Q-learning to make real-time recommendations for learners engaged with the MOOC using their previous interaction data with the MOOC content. In adaptive systems like the one the Q-learning framework demonstrates, recommendations continuously adapt to demonstrate changes in the learners' behavior such as engagement, course completion, and knowledge mastery. What was described in [202] moves the idea of adaptive systems one step further by utilizing offline RL methods to develop recommendation policies from data about learners' previous behaviors without the requirement of trial-and-error testing in an online context. Offline RL is also potentially useful for online systems that support large numbers of learners who learn in different ways and may generate sensitive data. Using a wider variety of historically sensitive data can increase safeguards and scalability for learner-recommendation systems. Inclusivity and learner-specific support were considered in [203], wherein the Adaptive Learning Recommender System (ALRS) was developed to support learners with dyslexia. This hybrid model leveraged evidence-based pedagogical practices in supporting a variety of personalized content and interventions that enacted cognitive and perceptual differences. While hybrid personalization does support equity of learning experiences, this also conformed to certain constructs of universal design for learning (UDL). Finally, the impact of AI-driven hybrid personalization strategies on learning outcomes was empirically assessed in [204]. The study looked at the effects of AI-supported (dynamic learning pathways and not adaptive assessments) hybrid models. The results demonstrated strong improvements in three significant areas of learner engagement, satisfaction, and academic performance, highlighting the useful nature of hybrid modelling in post-secondary education. Indeed, hybrid recommendation systems in MOOCS represent the intersection of machine learning, educational (behaviour) analytics, education and pedagogical theory, and human-computer interaction. Furthermore, the varying degrees from explainability, multimodal learning analytics, real time and timely, analytics, and inclusive designs make hybrid recommenders a cornerstone for attaining adaptive, personalized, and learner-centric online education on scale.

3.2.2.5. Complementary Techniques and Visibility Optimization

In the rapidly changing environment of online education, the quality and efficacy of MOOC recommendation systems is increasingly determined by its capability to merge intelligent personalized recommendations with enhanced discoverability of content. As discussed above, there are important aspects regarding search engine optimization (SEO), answer engine optimization (AEO), and engaging in the semantic web, that are central to improving the visibility of course content alongside enhanced AI driven recommendations. For example, the metadata from the course content and the user interaction behaviors, discussed in Chapter 3, within the MCRS system presented in [205], could help refine course recommendations using page optimization related to semantic relevance to provide stronger support for engagement, satisfaction, and user retention. Likewise, [206] proposed a hybrid recommender system that combined ontology-based domain knowledge modeling with memory-based collaborative filtering processes to semantically refine course suitability providing not only learners' preferences, but with additional hierarchy and taxonomy related to content - to help in better alignment with personalized learning pathways. [207] also uses behavioral data, particularly behavioral patterns and engagement models, to support the concept of learning behaviors in relation to improving recommendation quality. In this study, temporal learning patterns and navigation behavior were used to modify recommendations based on the most current interactions. This reinforces the point that user experience data is critical to exposure and personalization practices. Similarly, [208] introduced an attentional autoencoder technique that models the vectors of both course content relevance and user interest. The autoencoder is optimized on context-aware embeddings and the latent attention weights are modeled to maximize both global course relationships and personalized semantic proximity. This illustrates how visibility, or ranking, in a recommendation system can be optimized with neural attention practices.

Simultaneously, learner dropout or disengagement can be a critical feedback signal for adaptive recommendation. To that end, [209] proposed a Fractional-Iterative BiLSTM network designed to predict dropout in limited data, and while early intervention is possible in this context, this model also helps clarify recommendations as unintentional course abandonment means that recommendations made in the future should be adjusted. The use of prediction in this way acts as a reinforcement agent in at least two ways: Firstly, prediction generates feedback about personalisation and recommendations and secondly it adapts suggestions to reflect courses that can reasonably be expected to result in long term engagement. Together, this has been shown that visibility optimisation techniques - like ontological reasoning, collaborative filtering, attention modelling, and dropout leave efforts as they work synergistically with wider content marketing and adaptive learning, to help provide learner-centric, pedagogically sound, and discoverable recommendations.

Table 3 below provides a comparative analysis of some of the most recent personalized recommendation methods, which attempt to solve learner engagement issues, cold-start problems, dynamic preferences, and content relevance. The principal factors identified, personalization accuracy, adaptability, scalability, and interpretability, are summarized through weights and scales to help understand the advantages and trade-offs of each approach.

Table 3: Comparative Table of Recent Methods for Personalized Recommendation in MOOCs

Method	Year	Resolved Problem	Gap Resolved	Accuracy	Scalability	Interpretability	Resource	Adaptability
RL [201]	2023	Personalization of learning paths	Adapts to learner responses	High	High	Medium	Moderate	High
DNN [178]	2023	Cold start & overload	Clustering + sequence modeling	Medium	High	Medium	High	Very Good
AEO [205]	2018	Low organic visibility	Use structured data	High	High	Medium	Low	Good
SEO AI [206]	2018	Search Algorithm mismatch	Optimizes content via AI	High	High	Medium	Medium	Very Good
On-page SEO [207]	2020	Bad indexing	Better structural markup	High	High	Medium	Low	Good
Explainable RS [197]	2025	Black-box RS	High transparency + trust	High	High	Very High	Medium	Very Good
SWRL [186]	2022	MOOC cold start	Ontology + FSLSM	Medium	Medium	High	Medium	Good
Hybrid RS [187]	2020	Low search relevance	Semantic + adaptive filtering	High	Medium	Medium	Medium	Good
CF + Knowledge Graph [177]	2023	High dropout rates	Contextual + deep models	Medium	Medium	High	Low	Moderate
AI-Learning [204]	2025	Static teaching	Adaptive content & feedback	Low	Low	Low	High	Moderate
Co-SVD [175]	2024	Cold start	Sparse RS without	High	Medium	Medium	Medium	Good

			prior data					
MHRR [179]	2023	Niche course visibility	Better user profiling	High	Medium	Medium	High	Very Good
ALRS [203]	2023	High dropout rates	Feedback-driven hybrid RS	High	Medium	Medium	Medium	Very Good
Behavioral DL [198]	2024	Behavioral analysis	Deep Learning on learner logs	Medium	High	Medium	High	Very Good
Eye-Tracking RS [210]	2020	Passive engagement tracking	Attention-based course RS	Medium	Medium	High	Medium	Good
Motivation Classifier [176]	2022	Motivation-based RS	Predicts learner intent	High	Medium	Medium	Medium	Good
Ridge-RFE (GBM) [199]	2024	Imbalanced feedback	Real-time predictive control	Very High	Medium	Medium	Very High	Good
CNN + RNN [180]	2025	Static learning paths	Deep sequential learning	High	Medium	Medium	High	Good
CNN + LSTM Hybrid [181]	2024	Dropout detection	Sequential + temporal analysis	High	High	Medium	High	Very Good
MRSE [192]	2021	MOOC overload	ML for scalable filtering	Medium	High	Medium	Medium	Good
NER Hybrid [182]	2025	Unstructured MOOCs	Semantic entity linking	High	High	Medium	High	Very Good
GeAR [184]	2025	Semantic mismatch	Enhanced retrieval semantics	High	High	Medium	Medium	Good
Image Detection [193]	2025	Deepfake concern	Media verification	High	Very High	Medium	Medium	Very Good
Video Style Study [195]	2024	Engagement uncertainty	Style-driven cognitive match	Medium	High	Very High	Medium	Good

Frac-BiLSTM [209]	2023	Dropout prediction	Fractional calculus enhancement	High	Medium	Medium	Medium	Good
Onto-Knowledge Graphs [188]	2023	Static sequencing	Semantic trajectories	High	High	Medium	Medium	Very Good
Onto-KG [211]	2023	No learner-context linkage	Learner-course relation graphs	High	High	High	Moderate	High
Semantic Integration [189]	2025	Inconsistent data sources	API + semantic merger	High	High	High	Moderate	High
BERT for MBTI [190]	2025	MBTI from text	Captures nuanced personality	High	High	Medium	Moderate	Good
RL Sequencing [202]	2022	Personalized content flow	Offline-trained RL	High	High	Moderate	Moderate	High

3.2.3. Learner Behavior Analysis and Modeling

Learner behavior analysis is key to building adaptive and intelligent MOOCs as it contributes to different models such as dropout risk assessment models, learner engagement, emotional detection analysis, sequential behavior analysis, and timely and adaptive feedback and scaffolding that can bolster learner support and course design practices. This section summarizes the contributions based on four sub-themes derived from literature and recent projects.

3.2.3.1. Dropout prediction

Unlike many learning platforms, MOOCs have the distinct advantage of offering flexible learning to a diverse range of people in many different locations around the world. However, one ongoing challenge for MOOCs is the high dropout rate for enrolled learners. For that reason, understanding patterns in dropout and learner dropout predictors have become an important topic of study, in learner behavior analysis. Several new studies have described AI approaches, particularly deep learning and hybrid approaches for engaging models and more accurately predicting dropout with increased interpretability. A few studies focused on the use of deep sequential models for recognizing patterns and predicting dropout. For example, [212] developed an RNN-GRU based architecture for proposed for supporting dropout prediction, identified at-risk students by analyzing sequential log data. The model was effective, by using both information from historical interaction and context or learning data to dynamically predict disengagement. Similarity, [213] describes the FWTS-CNN, a feature-weighted time-series model consisting of convolutional layers, that identifies both short- and long-term behavioral cues from learner interactions specifically started modeling dropout behavior. The FWTS-CNN improved upon existing models, primarily by recognizing fused feature importance leading to dropout behavior. LSTM architecture has similarly shown to be effective for dropout modeling behavioral sequences.

[214] relied on LSTM networks to capture longitudinal student engagement and support early identification of risk in self-paced environments; the authors demonstrated the benefit of capturing non-linear dependencies across time that they could use for pre-emptive intervention strategies. Also building upon this work, [215] combined Multi-Layer Perceptron's (MLPs) with explainability, namely SHAP (SHapley Additive exPlanations), to identify dropout risks while also providing interpretable explanations by modelling the behavioral features that contributed to the prediction. The study may have provided a way to close the gap between performance and interpretability, a common issue in the educational applications. [216] also indicated the importance of personalization, specifically inclusive design, by completing a systematic literature review and hybrid predictive architecture. Their predictive architecture combined deep learning with traditional modeling to provide better support for students with disabilities and the importance of equitable and accessible predictive systems. Uncertainty management in modeling behavioral data on a large scale is an ongoing challenge, especially with the noise that occurs in MOOC datasets.

[217] framed their work using neutrosophic logic in a deep learning format, at a time when uncertainty, indeterminacy, and inconsistency were standardized features of educational data. Their model, therefore, accommodated uncertainty and improved prediction's robustness and realism. Their model allowed them to better serve ambiguous learner signals and offered a more novel way to address the fuzziness of behavior boundaries within dropout contexts. Other studies extended predictive modeling beyond single dimensional variants by accounting for multidimensional use data. [218] described a deep learning model that predicted dropout events using multidimensional temporal sparce data. Their model enabled dropout prediction using data to take into consideration correlations between dimensions corresponding to student engagement across multiple dimensions. The holistic engagement standpoint provided a much

more comprehensive way to model student behaviors and derived richer behavioural signals for predicting dropout. [162], similarly, used an ensemble framework incorporating both CNN's and LSTMs. LSTM's recognized patterns within established sequences and provide sequential memory while CNN's provided learners with additional local aspects of the pattern. In sum, their approach captured both micro and macro trend-level interactions with explicit use of time. Their model produced both superior predictive accuracy (lower false negative rate) and generality in predicting said interaction across MOOC datasets.

Additionally, [219] used neural network models to study dropout using behavioral data from a Latin American MOOC provider. Their results supported the cross-cultural validation of predictive models, demonstrating the importance of localized educational contexts for dropout prediction. Their work also demonstrated how neural networks are adaptable to different demographic and behavioral profiles, enhancing the generalizability of their models. Collectively, these works display a clear transition away from more static, rule-based, and statistical methods of dropout prediction, and towards dynamic and deep learning driven models. Additionally, there is also increasing interest in hybridization and ethical concerns. As MOOCs continue to evolve and diversify, the capability to model and mitigate dropout risk in a personalized, interpretable, and inclusive manner will continue to be central to improving learner retention and educational quality.

3.2.3.2. Engagement and Motivation Modeling

Modeling engagement and motivation in learners is becoming increasingly important in the modern e-learning paradigm, especially in MOOCs, where learners act alone and there is little contact with instructors. Engagement is a triadic concept, comprising behavioral, emotional, and cognitive. Modeling engagement must be done as accurately as possible, otherwise it will be difficult to promote their continued engagement and associated learning. New developments in affective computing, multimodal analytics, AIoT (Artificial Intelligence of Things) [220], and deep learning will make it easier - and scalable - to model the 3-dimensional constructs in real-time. This aligns with the findings from [221], which has situated user satisfaction as an important mediating variable into their assessment of engagement and user loyalty, at least, in terms of public sector ontologies in e-learning platforms; their approximation maps the process of perceived usability and satisfaction in service of long-term learner engagement. The authors studied engagement [222] in terms of its ethical and structural dimensions, as well as the historical merit and transformational power of digital education public learning spaces provide, bridging the gap between disconnected antecedents and engagement modeling. Each of these studies remind us to consider the wider socio-technical elements that may impact motivation beyond our studies that have isolated behavioral traces.

There is a need for inclusive engagement strategies in accounts focusing on underserved and special needs populations. For example, [223] performed an empirical study using an adaptive e-learning system on different types of dyslexic students. The results indicated that the degree to which an interface could be personalized and the ability to adjust content pacing, can contribute to better engagement and overall learning for students with reading disorders. In parallel, [224] provided a wider framework for AIoT based learning environments to adaptively increase motivation and performance, focusing primarily on the Arabic-speaking learners and other historically under-served populations. The adaptive e-learning environments studied included environmental sensors, usage analytics, and real-time personalization to establish more adaptive and equitable learning environments and learning systems. In recent years, the use of computer vision and deep learning for the recognition of emotional and attentional cues has received more attention [225].

[226] utilized Vision Transformers with few-shot learning to detect emotional engagement in webcam data, but what stands out is they required limited labeled data to train robust performance on detecting nuanced affective states. Furthermore, [227] implemented deep convolutional neural networks to identify attention shifts and engagement lapses in learners, using real-time video streams, indicating that passive monitoring systems can be applied in e-learning contexts. Indications of these methods are a move from static self-reports, or log-based heuristics, to real-time, multimodal engagement tracking. Engagement modeling has moved beyond mere affective detection to more comprehensive modeling frameworks. [228] used an integrative model to evaluate behavioral, cognitive, and emotional engagement through K-means clustering of asynchronous online learning data. Their work highlighted key differences in engagement profiles and referenced the need to consider the specifics of the context when designing interventions. The necessity to consider how learner feedback can be interpreted using sentiment and emotion analysis is explored in addition by [229] with a hierarchical attention mechanism applying both CNN and LSTM to comment texts on courses delivered online. Their model recognized both sentiment polarity and measure of semantic elaboration of learners' feedback providing a richer depiction of motivational dynamics from otherwise unstructured data sources.

In addition, there are a number of studies that have specifically researched the role of contextual and socio-demographic variables in engagement prediction. [230] created a predictive model that included contextual variables in the features of the model, for example: geographical access, device types, and socio-economic background to cluster learners, and engagement levels. In their emotion-aware personalization model the authors demonstrated that context-based clustering could be used to adapt differentiated learning approaches. With a similar aim, [231] put forward a hybrid deep learning model for the detection [232] and tracking of learner engagement over time. The authors showed an innovative approach to fuse behavioral logs, emotional markers and content interaction data via a multi-stream deep architecture. They offered a comprehensive description of the state-of-the-art predictive capability of the engagement classification adherence to the adaptive content delivery framework.

Taken together, these studies highlight a clear shift to multimodal, flexible, and ethically engaged forms of engagement and motivation modeling. From webcam affect detection [233] to personalization powered by AIoT, the field is quickly advancing to account for the individual and real time factors of learner engagement. Importantly, the integration of sentiment analysis, clustering methods, and neural attention has also allowed researchers to move past surface level behaviors and explore the deeper psychological and situational aspects of motivation. Beyond this, equity and inclusivity-oriented approaches especially for students with disabilities and learners from minoritized areas have the potential to realize greater equity in personalized e-learning experiences. Moreover, these changes have the potential to create more equitable e-learning experiences, geared to addressing these issues. When MOOCs are available globally, future engagement models may involve cross-modal data stream combined with explainable AI and lens models for transparency, fairness, and action-ability. This development will the opportunity for learners to predict and contextualize disengagement so that inline interventions can provide real-time engagement support to maintain motivation and cognitive engagement with the learning task.

3.2.3.3. Temporal and Sequential Modeling

Both temporal and sequential modeling are foundational in examining how students engage with educational content over time. Temporal modeling is one of the few ways to capture the dynamic nature of learner behaviors, opening potential avenues for early support, personalization on-the-go, and decreasing barriers in timely feedback mechanisms. Researchers are increasingly trying to model dynamic, time-aware activity streams since more timestamped data has become available through educational data mining, MOOC, and e-learning systems. It has been made clear in several recent studies that deep learning methods, especially Recurrent Neural Networks (RNNs), Gated Recurrent Units (GRUs), and Long Short-Term Memory (LSTM) networks see strong paradigms of temporal behavior capture and allow researchers to facilitate stronger predictions of future behaviors, given previous ones. One promising area for future research is the modeling of sequential data in deep sequential models to predict student performance and detection of disengagement. For example, in [226], they proposed a Transformer-based architecture with few-shot learning to detect student engagement in dynamic learning environments. This approach showed promising performance with very few labeled data, which was a big selling point for the model as there usually is very little labeled behavior data in the real-world.

Likewise, [214] modeled at-risk learners in self-paced environments using a model-based on LSTMs. Their approach identified long-term dependencies in a sequence of behavior and alerted instructors to dropout risk early, which enhanced the ability to provide pedagogical interventions as timely as possible. By synthesizing these elements, [162] introduced use of ensemble models combining CNNs and LSTMs to model dropout behavior within MOOC contexts. The CNNs extracted local and mid-level feature patterns from the engaging signals and used the LSTM to map local patterns to a temporal evolution over time. The model was able to simultaneously model low-frequency long-term learning behavior while still being sensitive to rapid short-term change. These approaches showed not only the capability of deep learning for educational data, but also the potential to model the complete set of learner variability by using hybrid architecture. Another major advance in the field of learning analytics was made with the development of localized platforms, such as the Tadakhul System described in [234]. The indigenous MOOC platform used a BiLSTM-CNN architecture to re-personalized learners' flows of content based on the feedback from each learner. The adaptive and learner feedback loops coupled with the ability to dynamically alter the content sequence based on individualized learner progression were a strong demonstration of the potential of temporal modelling for real-time personalization, while also allowing for cultural contextualization.

Besides prediction, temporal modeling has been used for behavioral clustering and consequent success prediction. [235] used a clustering algorithm along with CNNs to identify learners and their temporal engagement behavior groupings and predict success in academic courses. Their educational data mining framework demonstrated how unsupervised learning was used with deep models to provide insight into hidden groups of learners and performance risk. This type of work also has important implications for institutions wishing to monitor student learning across larger cohorts more systematically and on a scale. Nonetheless, some recent works have emphasized utilizing temporal analytics as more than just sequence modeling. For example, [236] suggested a synergy-oriented framework that utilized process mining with explainable artificial intelligence (AI) and ontological modeling. This framework used log data to produce learning paths and then used human-readable rules to interpret behavior change. Thus, the framework provided transparency in understanding why students made learning decisions. Similarly, [237] compared two types of natural language processing (NLP) and process analytics to investigate students' problem-solving behavior in terms of Mooc problem-solving behavior. Here, they analyzed student forum discussions and students' open-ended responses using NLP and process rules to understand the cognitive strategies across time. In effect, they identified the students' cognitive strategies in temporal form, demonstrating how NLP and process analytics could be used together to provide a better representation of student behavior.

Temporal modeling has also been used to create emotion and sentiment-aware analyses of educational behavior. In previous research, [229] used a Hierarchical Attention Mechanism from CNN-LSTM models to analyze the portrayals of sentiment in learner feedback that occurred longitudinally over time. Their model identified how emotional changes occurred over the course, and how trajectories of learner sentiment and affect over the time course correlated with dropouts, frustration, or satisfaction levels. By using attention mechanisms, the authors could identify the text-based moments that represented a significant motivational change and thus integrated affective elements into the temporal behavioral model. Finally, hybrid models, such as the one proposed by [231], illustrate how temporal modeling may be applied across a multiple data stream. The authors reported the integration of behavioral logs, engagement signals, and social and affective inputs to detect and characterize learner engagement levels over time with CNN-LSTM hybrid model. This allowed for engagement levels to be recorded longitudinally, as well as engaging with learners in real-time while they interacted with e-learning platforms. Multiple modalities of learners input allow for high-fidelity engagement modeling. The field of temporal and sequential modeling in learner behavior analysis finds itself at the crossroads of a rich intersection of predictive potential, interpretability, and multimodal data processing. Seeing matters graduate from the traditional sequence models such as GRUs and LSTMs to more contemporary and sophisticated Transformer-based and hybrid deep learning models certainly demonstrate the heightened emphasis on both complexity and diversity in online learning behaviors. Furthermore, with the additional focus of explainability tools, context data, and affective analysis then contemporary models have the seamless ability to provide rich and readable representation of the learner journey. These advancements do not simply exacerbate the modeling of learning, but also lay the foundations for reignited, inclusive, and pedagogically aligned education systems.

3.2.3.4. Intelligent Tutoring and Feedback Systems

The integration of Artificial Intelligence (AI) into tutoring and feedback systems marks a powerful shift in online education by allowing systems that adapt to learner needs, behavior, and affect in real-time. These Intelligent Tutoring Systems (ITS) are now at the heart of personalized learning and behavioral modeling, providing rich data about student performance so that instructional strategies can be updated and adapted in real-time. Especially in MOOCs and large-scale online learning where instructors interact with fewer than 25% of their enrolled students, AI-based feedback can be used as virtual mentors to detect disengagement, diagnose epistemological misconceptions, and offer individualized learning support. One important area of recent studies is that of adaptive tutoring systems for targeted or underserved populations. [223] has described empirical testing of an adaptive e-learning platform that was specifically designed for dyslexic learners. Their research indicated that personalization was critical about how information was presented, how the navigational interface was configured, and the cadence of presentation. Their results further showed that as the adaptation of the platform had given learners more control over these factors, comprehension and retention were significantly improved for learners with specified needs. This work has implications for intelligent tutoring systems if we consider that they need to be inclusive, accommodating not only learning styles but also addressing neurodiversity and accessibility needs.

From a more algorithmic and predictive perspective, [216] conducted a systematic literature review about deep learning-based engagement and performance prediction in MOOCs. Of the various conclusions, a prominent conclusion highlighted the potential of proactive feedback systems that combine models of learner behaviors and data analytics in real-time. These systems are no longer based strictly on reactive tutoring, they are in a position to potentially predict declines in performance and provide interventions in a timely fashion as either automatic tutoring support or recommendations for re-mediated context. [217] extended this idea to deep neutrosophic learning which is the ability to reason with uncertainty. Which positions their framework to deal with contradictory or ambiguous behavioral data in the learner space, along with probabilistic outcomes, which is relevant for supporting interventions for when inconsistent and contradictory behaviors are present in real-world datasets. More algorithmic approaches have received attention, however the encapsulation of principles of instructional design in intelligent tutoring environments have derived attention.

[238] developed an evaluation framework for MOOCs based on Biggs' constructive alignment theory. Their framework is macro-oriented and considers how well the learning goals, instructional strategies and assessment strategies are aligned within the platform. Although it isn't an AI model in a traditional sense, their framework is quite useful in the design of pedagogically nuanced, intelligent feedback systems, which enables AI pathways and suggestions to be aligned with learning outcomes at the course level. In terms of behavioral understanding and modeling of content, [237] used Natural Language Processing (NLP), in order to analyze students approached to solving problems in a MOOC. They found that the forums and open-ended responses written by the learners contained rich and valuable behavioral signals that could be used to form the basis of intelligent feedback. They built the conceptual representation of problem-solving by using clustering and topic modeling, which allowed them to predict learners' approaches to problem-solving. The prediction of learners' approaches to problem-solving would allow intelligence to deliver hints and feedback that were appropriate based on the inferred cognitive style. Importantly, this technique allowed the system to "understand" not only outcomes and performance but, the process in which the outcome was achieved.

A broader systemic perspective can be found in [239], who considered the role of institutional readiness in AI inbox tutoring. They argued that AI can extend learning by providing intelligent feedback, but the integration of AI must be supported by a pedagogical fit, faculty development commitment, and digital infrastructure. Their writing emphasized a holistic view of intelligent feedback systems that consider not just technology components, but the organizational and instructional ecosystem within which these systems operate. Absent that pedagogical fit, even the most advanced AI tools may be underutilized or pedagogically incompatible. All these writings suggest that intelligent tutoring and feedback systems are now capable of far more than canned responses or generic advice. They are getting increasingly sophisticated, predictive, and pedagogically aware, synthesizing data from behavioral analytics, affective computing, learning science, and instructional design. They embrace uncertainty handling, personalization of learners, feedback topic awareness, and even institutional fit to create a more network of adaptive learning support. Intelligent tutoring systems have evolved as a rich intersection of deep learning and behavioral modeling with educational theory. These systems do not simply "track" student behavior but interpret and react to student behavior and predict it in hopes of tailoring content and feedback in the moment. As discussed in the presented papers, advances range from focused applications designed to assist dyslexic learners, to uncertain behavior modeling, to process-aware instruction, to AI-based analysis of learner-generated content. Future research will likely extend intelligent tutoring systems further into multi-mode environments, utilizing data related to voice, video, and gestures, while promoting fairness, understanding, and human-in-the-loop instructional design.

The analysis of learner behavior in MOOCs has progressed from simple engagement-derived measurements to a more comprehensive approach using multimodal, explainable, and personalized systems that enhance retention supports real-time adaptation and provides equitable access to quality online educational experiences, as summarized in the Table 4 below.

Table 4: Comparative Table of Recent Methods for Personalized Learner Behavior

Method	**Year**	**Resolved Problem**	**Gap Resolved**	**Accuracy**	**Scalability**	**Interpretability**	**Resource**	**Adaptability**
E-learning Service Quality [221]	2023	Low user satisfaction	Enhances service and personalization	High	Moderate	Medium	Low to Moderate	Moderate
AI-Enhanced Education [222]	2025	Limited personalization	Enables adaptive and scalable learning	High	High	Medium	Moderate	High
AIoT Learning Systems [224]	2025	Generic instruction	Delivery tailored content	High	High	Medium	Medium	High
Adaptive E-Learning [223]	2023	Poor support for dyslexics	Personalizes materials	High	Medium	Medium	Medium	Good
ViT + LSTM [226]	2025	No engagement labels	Combines DL and cloud for prediction	High	Medium	Moderate	Medium	Good
Sentiment Analysis [240]	2022	Ignored learner feedback	Uses feedback to guide content	Medium	Medium	High	Medium	Good

RNN-GRU [212]	2020	Missed at-risk students	Captures behavior over time	Moderate	High	Moderate	Moderate	High
FWTS-CNN [213]	2020	MOOC dropout issues	Models time-based engagement	Moderate to High	High	Moderate	Moderate	Good
Explainable DL [215]	2023	Black-box predictions	Adds interpretability	High	High	High	Moderate	High
Clustering on LMS Data [241]	2024	Hidden engagement signals	Reveal patterns early	High	High	Moderate	Low to Moderate	Good
DL + ML Integration [216]	2025	Incomplete dropout models	Fuses methods for better accuracy	Medium	High	Medium	Medium	Good
Neutrosophic DL [217]	2025	Noisy behavior data	Handles uncertainty in data	High	High	Moderate	Moderate	Good
Survey in Gaza [242]	2023	Lack of local insights	Captures learner perceptions	High	Moderate	High	Low to Moderate	Moderate
CNN [227]	2020	Poor online engagement	Adds emotion detection	High	High	Medium	Medium	Good
CNN [243]	2021	High resource usage	Builds light, efficient model	Medium	High	Medium	Low	Good
CNN-KMeans [235]	2022	Poor clustering accuracy	Optimizes K-means with metrics	–	High	Medium	Medium	Good
RNN-CNN [218]	2022	Noisy MOOC behavior	Reduces noise, automates features	–	Medium	Medium	Medium	Good
CNN-LSTM [162]	2024	Early dropout detection	High short-term accuracy	High	Medium	Medium	Medium	Very Good
Tadakhul System [234]	2024	Cold start issues	Predicts behavior for personalization	High	Medium	Medium	Medium	Very Good

Machine Learning [228]	2023	Narrow engagement view	Adds emotional/cognitive metrics	–	High	High	Low-Medium	Medium
Process Mining + XAI [236]	2025	Non-transparent insights	Adds explainability	–	High	Very High	Medium	Very Good
CNN-LSTM [229]	2023	Weak sentiment context	Improves context with attention	High	Medium	Medium	Medium	Good
NLP Framework [237]	2023	Off-platform behavior ignored	Analyzes free text for insight	Medium	Medium	High	Medium	Good
Predictive AI [230]	2024	Poor learner grouping	Segments students for support	High	Medium	High	Medium	Very Good
Teacher Attitudes [239]	2025	Low AI adoption	Identifies awareness gaps	High	Low	Medium	Low	Good
Hybrid Engagement [231]	2023	Complex engagement signals	Merges emotion, clustering, decisions	High	Medium	Medium	Medium	Very Good
Deep Neural Network [244]	2022	Sparse feature prediction	Uses rich feature set	High	High	Medium	High	Very Good

3.2.4. Semantic Course Retrieval and Search

The practice of semantic retrieval in MOOCs has changed from traditional keyword-based methods into sophisticated and contextualized search systems. For ease of understanding this section categorizes and collates studies into four sub-areas making sure that reference only occurs once.

3.2.4.1. Traditional Information Retrieval

In its early implementation, the MOOC platforms employed classical information retrieval methods such as TF-IDF and BM25 to retrieve course content. While keyword-based information retrieval systems can serve a purpose in low-resource contexts, they have a limited capability to assess context or semantics. One study [245] highlighted SEO principles in the agriculture domain using various strategies of keyword optimization and opened up possibilities for cross-domain relevance that suggest fit for educational content. However, these models can struggle with vocabulary mismatch and do not capture meanings, decreasing their utility for contextual or conceptual retrieval." "The research community has also begun to address the limitations of traditional models. One example is the BMX model that uses entropic similarity and semantic context to improve upon BM25 by combining vector-based components BMX demonstrates improved performance benchmarks in long-context retrieval [246]. In another example, combining embedding-based similarity metrics with BM25 has brought substantial improvement in IR with biomedical [247], [248] literature indicating that integrating lexical and semantic approaches generates higher performance than relying on either one alone [249]. describe how retrieval is moving from a feature space towards neural models and the need to model query–document intent and meaning especially when using multi-stage pipelines where BM25 is leveraged to retrieve candidates followed by semantic based re-ranking [250].

3.2.4.2. Semantic Embedding and Bi-Encoder Models

The emergence of semantic search represents an important evolution in modern information retrieval, specifically in the field of online education. Traditional keyword-based search approaches are limited in their ability to capture the context of learner queries, especially about MOOCs (Massive Open Online Courses), which can be lengthy, heterogeneous, and semantically rich. In this context, dense retrieval models that utilize semantic embeddings are adjusting to more advanced semantically aware systems that understand the meaning of the user queries and their course content. Two promising innovations we will discuss are bi-encoders and dual-encoders. These systems have suggested improvements in retrieval performance in precision, scalability and generalization. For example, a recent publication provided performance increases with a dense retrieval model for academic question answering, where a bi-encoder model with hard negative sampling significantly outperformed traditional retrieval-based approaches in terms of Mean Average Precision (MAP) and Normalized Discounted Cumulative Gain (NDCG). This work contrasted retrieval that augmented the model with LLM-generated synthetic titles, along with similar features that combined embeddings to produce additional contrastive embeddings, in addition to other domain-specific loss functions as provided over the LoTTE and BEIR datasets, indicating the model became more generalizable in the open-domain academic retrieval task space [251].

Research in the field of Search Engine Optimization (SEO) is now beginning to intersect with AI-based retrieval. As in papers [252] and [253], the importance of applying AI-infused SEO practices - namely utilizing NLP-based keyword targeting, semantic indexing, and content recommendation algorithms - to increase the visibility and ranking of MOOC content is emphasized. These practices not only support the visibility of educational resources through conventional search engines but also ensure that resources surfaced are consistent with the learner's intent and context. Semantic embeddings used in conjunction with personalized recommendation systems advance the use of intelligent course retrieval. Semantic search engines now avoid strict string matching and turn natural language questions into acts of semantic retrieval based on commonalities in themes, prerequisites, learner goals, or pedagogical framing. This is important because of the variety in learner profiles, and the increasing speed at which the digital learning ecosystem is expanding.

3.2.4.3. Ontology and Graph-Based Retrieval

Ontologies and graph-based models are central to the representation of knowledge in a specific domain in the task of semantic course search and retrieval along with providing intelligent query semantics. They support intelligent query resolution that extends beyond traditional keyword-based search by formalizing the relationships between various entities (i.e., learners, instructors, courses, and knowledge concepts) into ontologies or graphs to allow for more advanced semantic reasoning and context aware retrieval. Heterogeneous Information Networks (HINs) are a well-known approach for representing multiple types of entities and relationships in a single unified graph structure and can include HINs, other models using a formalized representation of knowledge in a domain, and combination of a HIN and ontologies, etc. For example, [254] discussed ACKRec a HIN-based recommendation framework which represents the interrelations of learners, instructors and courses in MOOCs using a HIN. Specifically, their model uses attentional graph convolutional networks that use semantic meta-paths to encode multi-relational representation of learning to deliver recommendations to learners in a HIN. Using a HIN allows for a more granular modelling of learner preferences by considering not just content, but also the social and instructional context of learning.

Concurrently, researchers are developing ontology-based semantic retrieval systems that offer accurate, rich data sets. [255] presented a semantic academic search engine that used the Web Ontology Language (OWL), Protégé for developing ontology, and Apache Jena for inference and reasoning. These technologies allowed the system to interpret the user query based on a formal representation of domain knowledge and facilitate features such as synonym matching, traversing a concept hierarchy, and reasoning over relationships that are not explicitly stated. For example, in response to the phrase "machine learning foundations," it provided a list of relevant courses, even though the titles or the descriptions used ideas related to supervised learning or foundations of artificial intelligence. These examples are particularly pertinent as we see the development of Answer Engine Optimization (AEO). [256] indicated, we are moving towards a near future of educational discovery where AI-powered search engines, such as Google's Search Generative Experience (SGE) and ChatGPT, will become opportunities for discoverers of educational resources. So rather than displaying links in response to keyword-based queries, these search engines are geared towards determining user intent, generating a synthesized answer, and frequently answering the query in the search interface, a zero-click search environment. Moreover, ontology and graph-based retrieval systems provide a strong basis for advances in semantic retrieval to facilitate course search in MOOCs. Recognizing the complexities of educational domains and the emerging lines of development in conversational AI and zero-click search, ontology-based systems improve the relevance of the content retrieved and allow MOOC platforms to evolve into intelligent learning ecosystems. Future work could focus on ontology alignment across multilingual platforms, dynamic ontological evolution based on learner interactions, and hybrid graph-transformer models that support scalable semantic reasoning in large-scale educational contexts.

3.2.4.4. Context-Aware Search Systems

The progression of semantic course retrieval has shifted from simple keyword-based matching to more comprehensive, context-based retrieval systems that account for factors like user intent, societal and ethical implications. These new types of retrieval systems aim to offer improved and more appropriate relevant content to course users while retaining fairness, accountability, and inclusiveness in academia. One challenge before developers of retrieval systems that utilize AI or ML is manipulative behavior and misuse, with implications for adversarial attacks [257]. For example, [258] illustrates how adversarial search engine optimization (ASEO) leverages and exploits ranking systems based on large language models (LLM). Adversarial techniques allow the objectives of objectivity and trustworthiness in content delivery to be distorted and maintained and have severe implications for the integrity of academia and learner trust. Their study specifically outlines strategies adversaries pursue for creating inputs that shift semantic relevance signals, justifying the need for defenses against ranking adversarial attacks.

Apart from technical vulnerabilities, political and cultural biases may also influence elements of content retrieval and delivery in context-aware systems. [259] undertook a cross-lingual study using generative search engines, such as Microsoft Copilot, to see if political narratives differed depending on "language" or regional settings. They found that politically sensitive information produced by chatbots powered by Artificial Intelligence was different in five languages. The results suggested that based on the regionally adaptive responses to the user's requests, algorithmic gatekeeping could perpetuate or alter user perceptions, adding further ethical considerations to educational search tools that align with transparent, accountable AI. Furthermore, from a socio-technical perspective, inclusivity is yet another area where context-aware systems have shown promise. As a means to mitigate the digital divide in low-resource linguistic communities, [260] created a speech-based retrieval system for the Amharic language using CMU Sphinx ASR the automatic speech recognition system [261], [262], [263]. This system demonstrates the ability to facilitate non-textual, voice-based interactions in underrepresented communities, offering accessible educational retrieval experiences for marginalized learners. The available technologies reflect the need for multimodal interfaces, especially in global MOOC contexts.

On a more societal level, the impacts of search engine bias driven by AI-powered retrieval go beyond functional efficacy to the degree to which public narratives are shaped and constituted. [264] explored the effect of bias in search engine representations of persons with intellectual or developmental disabilities on public perceptions. The studies show that where search engines depict unfair or stereotypical portrayals, views on persons with disabilities were directly influenced, emphasizing the role of fairness, representation, and ethical AI design in authentic search models, especially in a learning context where the learners' views can be shaped by the exposure such biases present. Taken together, these studies demonstrate that context-aware search systems are not merely technical functions but rather socio-technical systems with implications that may extend across personalization, accessibility, cultural equity, and public discourse. Future directions must not only fit semantic relevance as meaning above all, but also respect principles of transparency, inclusivity and accountability into the design of intelligent course retrieval systems.

The domain of semantic course retrieval in massive open online courses (MOOCs) is a fast moving and multi-dimensional area. Current trajectories highlight a movement from older retrieval methodologies to newer, more advanced semantic search techniques, including embeddings, bi-encoder construction, graph-based ontologies and context management methods, that are increasingly focused on supporting personalized learning experiences and more equitable learning environments, as highlighted in Table 5.

Table 5: Comparative Table of Recent Methods for Semantic Course Retrieval and Search

Method	Year	Resolved Problem	Gap Resolved	Accuracy	Scalability	Interpretability	Resource	Adaptability
ACKRec [254]	2020	Poor modeling of user-content connections	Model connections on HIN meta-paths	High	Medium	Medium	High	Good
Semantic Search [255]	2025	Poor academic search precision	Uses ontologies for more learning from the query	High	High	Medium	Medium	Very Good
AEO [256]	2023	Low content to search engine visibility	Improves SEO through structure and/or conversation data	High	High	Medium	Low	Good
SEO 6O Model [265]	2024	No defined criteria for the SEO for courses	Classify important features to optimize SEO	High	High	Very High	Medium	Good
LLM-Augmented Retrieval [266]	2024	Weak query-document registration	Improves retrieval with LLM context	Medium	High	Medium	Medium	Good
Bi-encoder Model [251]	2024	Poor document retrieval efficiency	Use dense vectors in matching information	High	High	Medium	Medium	Good
On-page Optimisation[245]	2023	Incorrect website indexing	Increase SEO structure and visibility	High	High	Medium	Low	Good
AI-Enhanced SEO Practices [252]	2024	Expired history of SEO practices	Improves keyword/content optimization	High	High	Medium	Medium	Good

AI-driven SEO [253]	2024	Projected changes in search algorithm	Uses AI to adapt and improve engagements	High	High	Medium	Medium	Very Good

3.3. Conclusion

The detailed analysis of recent evolutions in MOOCs discloses a deep and dynamic research ecosystem influenced by increasingly intelligent, personalized, and data-rich mechanisms. We examined the four broad methodological pillars that have transformed widescale online learning from learning analytics to analytics for learning: course classification, personalized course recommendations, learner behaviours and semantic course retrieval. Within course classification, deep learning and hybrid modalities, especially transformer architectures, have increased semantic understanding and thematic organization of educational content. Recommendations have developed from undergraduate students utilizing basic collaborative filtering to become sophisticated attention-based and reinforcement learning systems for delivering personalized learning pathways that expand to accommodate learner preferences, learning progress and goal attainment. The learner behaviour domain has been enriched by temporal modelling, input layer modalities, and the advent of explainable artificial intelligence to describe engagement, performance, or attrition. Semantic course retrieval has also notably developed semantic course retrieval using knowledge graphs, context embeddings, and deep retrieval systems authenticating meaningful interactions between learners and content repositories. Stepping back from these innovations, it can be seen that MOOCs are moving into intelligent ecosystems, where adaptivity, scalability, and personalization are not just desirable features, but are fundamental design principles. With the interplay of artificial intelligence, learning analytics, and pedagogical design, we are witnessing the rise of a new era of educational systems that are more learner-centered, address diverse needs, and deliver meaningful learning experiences at scale. Although significant advances have been made, many challenges are inevitable, including interoperability, ethical use of learner data, real-time adaptivity, and alignment with instructional goals. Addressing these challenges requires a common methodological framework that can subscribe to whatever existing approaches work or contribute new ones and models.

Within this setting, the following chapter will outline the methodology approach and bring to light the multi-task contributions of this research. It will expand on the architectural principles, learning strategies, and technical development of intelligent MOOC systems that will move intelligent MOOC systems closer to more integrated, scalable, and learner-oriented approaches.

Chapter 3: Methodological Framework and Multi-Task Contributions

4.1. Introduction

The latest movements in artificial intelligence (AI) and deep learning (DL) in the digital education arena have opened possibilities for creating intelligent tutoring and learning systems. In the case of Massive Open Online Courses (MOOCs), there is a growing realization that also requires intelligent, scalable, and adaptive means to cope with the complexity and heterogeneity of learner behavior, course contents, and managing individualization in pedagogical communication. This chapter presents the overall methodological framework for this study, and it provides contextual information to present the technical contributions into the MOOC environment and the challenges it poses will specifically include course classification, course prediction (personalized course recommendation), modelling for learner behaviour, and semantic course retrieval [267].

By providing four innovative models in terms of different educational tasks, the chapter provides a holistic perspective that couples the state-of-the-art neural architecture, optimal strategies, and attention. The proposed methodologies in this research aim to provide improvements in the accuracy, scalability, interpretability, and individualization of intelligent systems using MOOC-based data.

We begin by describing the general research methodology that lays the foundation for all subsequent contributions including dataset creation, feature extraction methods, and model evaluation methodologies. We then provide a detailed description of each contribution:

- The first contribution introduces a Convolutional Neural Network optimized with Lion Optimization Algorithm (CNN-LOA) for robust and scalable course classification. This model answers the pressing issue of how to automatically classify the enormous library of MOOC content by learning to classify course titles and descriptions into discriminative features while also using LOA to enhance hyperparameter search and the stability of convergence by utilizing physics-based modeling.
- The second contribution deals with personalized recommendations using hybrid deep architecture known as HMHA-C-BERT. This model fused BERT embeddings with

BiLSTM and multi-head attention to model the semantic embeddings of courses and user preferences in a deep fashion. A contextual compression method was also used in architecture to enhance computational effectiveness with minimal impact on outcomes.

- The third contribution is a temporal attention based recurrent model (TSA-GRU) for prediction of learner behavior. It fused Gated Recurrent Units (GRU) with a temporal sequential attention mechanism to model the changing evolution of students learning over a duration. This model is intended to provide an early prediction of dropout or the reduction of performance enabling timely intervention based on the prediction.
- The fourth contribution defines a semantic retrieval framework that leverages a Multi-Head Bi-Encoder framework. Here, it learns representations for both user queries and course descriptions, through two parallel but independent neural processes, then combines their representations using a multi-head attention mechanism. The result is a practical context-aware, efficient, accurate course retrieval mechanism for MOOC-like repositories.

The models are all evaluated to generalizable findings using appropriate training-validation splits, cross validation, and statistical significance tests for no less than three test sets or cohort splits. Performance is assessed using precision, recall, F1, accuracy, and correctness, selected as appropriate for each task.

The objective of this chapter is to demonstrate the entire methodological process used in this research starting from the data preprocessing approaches and feature engineering approaches as the basis of all models presented going forward. It introduces the CNN-LOA model for course classification, the HMHA-C-BERT model for personalized course recommendations, the TSA-GRU model that includes temporal modeling and builds to predict learners' behavior, and the Multi-Head Bi-Encoder model for semantic course retrieval. This chapter also builds on the experimental design and evaluation approach to build methodological consistency and empirical rigor across all systems proposed. Together, the approaches outlined represent a common methodological framework to be applied to the development of intelligent MOOC platforms. Through the integration of deep learning, hybrid optimization methods or attention-based approaches, the chapter attempts to create a baseline for the development of educational technologies that are adaptable to the learner, interpretable, and learner centered.

4.2. Overview of Methodological Approach

This thesis attempts to address the growing complexity and diversity of learners' needs in digital learning contexts and subsequently, puts forth an AI-supported methodological framework that consists of four original contributions to enhance the effectiveness of MOOCs (Massive Open Online Course). However, rather than being based on a single pre-existing research framework, this thesis offers the design, implementation, and evaluation of four intelligent systems as each was created for a well-defined problem within the MOOC ecosystem. These problems include:

- course classification.
- personalized course recommendations.
- Learner behavior analysis.
- semantic course retrieval.

The methodologically overall approach is empirical, systems-based, and data-dependent, which combines machine learning (ML) and deep learning (DL) approaches with controlled experimental design to study how intelligent models can affect the personalization, space, and instructional quality of large distance education environments. Each of the four contributions was informed by recent advancements in artificial intelligence and their limitations based on the analysis of research literature within distance education systems (Chapters two and three). The research methodology which is a structured and iterative process will follow these key stages:

4.2.1. Problem Identification and Contribution Framing

All contributions start with a specific gap or limitation of existing MOOC-related platforms. These gaps are based on empirical evidence and a critical analysis of previous work in online learning, learning analytics, and educational data mining. The aim in all cases is to offer a model that represents a measurable advantage over conventional methods.

4.2.2. Model Design and Algorithm Selection

For each problem identified, a suitable intelligent architecture is either selected or created. Some examples include using Convolutional Neural Networks (CNNs) for classification problems, and transformer architectures with hierarchical attention mechanisms for recommendation systems. Another example includes GRU-based hybrid models for predicting learner behavior, and multi-head Bi-Encoder frameworks for context-aware semantic search.

Optimisation approaches are used wherever possible e.g., using the Lion algorithm for hyperparameter optimisation to improve model performance.

4.2.3. Data Collection and Preprocessing

The experimental research relies upon real world datasets collected from available MOOC platforms, or communal institutional repositories, with these datasets composed of structured metadata (e.g., course categories, learning objectives), unstructured content (e.g., course descriptions, learner feedback and reviews), and behavioral log data (e.g., clickstream, quiz submissions, forum activity). Preprocessing such as data cleaning, feature extraction, data normalizing, and embedding generation occurred to optimize the quality and usability of the input data.

4.2.4. Model Training and Evaluation

Every system is trained in a high-performance computing environment and evaluated through standardized metrics appropriate for the given task. For example:

- The classification models are evaluated based on training and testing accuracy and training and testing loss to evaluate both learning and generalization performance.
- Recommendation systems are evaluated using Train Loss, Train Accuracy, Test Loss, and Test Accuracy.
- Behavioral prediction models are evaluated on training/testing loss and accuracy, with 5-fold cross-validation and statistical significance tests included to establish robustness and generalizability.
- The semantic retrieval system will be evaluated using metrics for classification such as Train/Test Accuracy, Loss, Precision, Recall, and F1-score.

In general, we use baseline models and regular techniques for comparison points to measure the success of the proposed systems.

4.2.5. Results Interpretation and System Integration

Each contribution's results are evaluated in relation to contributions' innovation on model performance, scalability, and pedagogical implications. Some surprising discussions include interpretability of the results, the generalizability of the models, and potential integration into existing MOOCs. Wherever appropriate, we have included visualizations, ablation studies and error analyses in addition to the final written results to give more understanding.

This multi-phase methodology aims to demonstrate the technical feasibility of cutting-edge artificial intelligence models in online education, but also to help share lessons learned about the design of intelligent, learner-centered, and scalable digital learning systems. The selected contributions are representative of important domains of innovation at the intersection of artificial intelligence, and e-learning, and together they provide a forward-looking perspective on intelligent distance learning.

4.3. MOOC Course Classification using CNN and Lion Algorithm

Given the rapid growth of digital education, the designation of MOOCs (Massive Open Online Courses) is becoming a daunting task, particularly due to the various course offerings multiplying each day. Even traditional hand-curated or rule-based systems are not sufficient to keep up with the velocity and unstructured contexts of MOOCs. Therefore, this contribution provides a novel intelligent classification solution that utilizes Convolutional Neural Networks (CNN), along with the Lion Optimization Algorithm (LOA), a bio-inspired metaheuristic that mimics social interactions of certain types of lions and the cooperative hunting patterns groomed by lions. The goal of the CNN-LOA proposed in this contribution is to improve classification accuracy, scalability, and generalization performance of MOOC course classification. This is done with an automatic optimization of key CNN hyperparameters. This section explains the motivation behind the method, describes the architecture and optimization method, shows the dataset with the preprocessing methods, and evaluates the performance of the method with results of extensive experiments and comparisons to baseline classifiers [268].

4.3.1. Problem Statement and Objectives

Massive Open Online Courses (MOOCs) have greatly increased access to education globally through free, online courses covering a wide range of disciplines. The expansion of content is now becoming particularly problematic where course classification and personalized recommendation systems are concerned, due to the volume and volume of content with a diversity of inherent features. Manual and rule-based classification is neither effective nor scalable and is not equipped to deal with the ever-changing and unstructured data of MOOCs.

Developments in machine learning (ML) and especially deep learning (DL) allow for improved automatic classification of courses through Convolutional Neural Networks (CNNs), which automatically extract pertinent features from text-based input. However, other associated issues still exist. For instance, static, manually tuned hyperparameters, reliance on data known as a priori to be representative of a course (lack of generalization of unseen data) and inflexibility in handling reorganization of course material. One of the more significant issues is hyperparameter tuning, which is also difficult, can be very time-consuming, and will significantly impact performance while requiring trial and error.

To address these challenges, our contribution proposes a hybrid deep learning and optimization framework - CNN-LOA, which integrates Convolutional Neural Networks with the Lion Optimization Algorithm, which is a bio-inspired metaheuristic algorithm based on the social hierarchy and cooperative hunting strategies pursued by lion social groups in their natural habitat. The proposed method automated the tuning of important hyperparameters of CNN to not only improve classification accuracy, but also generalization, and to improve robustness and scalability of MOOC classification systems.

This subsection seeks to answer the following research question:

In what way does the optimization of the CNN parameters using bio-inspired algorithms modify the accuracy and efficiency of course classification on MOOC platforms?

Aims of the Contribution:

- ✓ To create a scalable and intelligent model for classifying MOOCs based on course descriptors using metadata and description.
- ✓ To create a set of hyperparameters approaching optimality for CNNs with the Lion Optimization Algorithm to improve learning speed and accuracy of classification.
- ✓ To enhance the semantic meanings of the course content with rich feature representation and data processing.
- ✓ To evaluate the accuracy and efficiency performance of the developed CNN-LOA models in comparison to conventional classifying methods.

4.3.2. Dataset Description and Preprocessing

This section describes the preprocessing and classification workflows that we applied to the course dataset where we classified thematic areas for each course using text processing and classifiers. Specifically, text content was converted to numerical format and K-Means clustering was used to identify groups of topic-based courses. Once the data was classified it was used to train the CNN model using the Lion Optimization Algorithm (LOA). The workflows are broken down below in a sequential manner:

4.3.2.1. Importing Libraries and Dataset

The analysis started by importing libraries, including pandas for data manipulation, and scikit-learn for text vectorization and clustering. The dataset was obtained from Kaggle and includes fields such as CourseTitle, Description, Availability, and Cost. We loaded the dataset into a Pandas DataFrame to facilitate the structured analysis and pre-processing of data.

4.3.2.2. Text Preparation

CourseTitle and Description were concatenated in a new column called Text to provide a greater level of description for each course. This column format provided greater context for subsequent text analysis and future model learning.

4.3.2.3. Cleaning Text Data

We created a custom clean_text function to improve the text data quality. The cleaning procedure included:

- Removing special characters, punctuation, and numerical digits using regex.
- Tokenization of text utilizing nltk. Word tokenizes.
- Lower casing the tokens for consistency.
- Removing common English stopwords (e.g., "the", "and") using NLTK's stopwords list.
- After cleaning our content, we stored it in a new column, and the text data was ready for vectorization.

4.3.2.4. Text Vectorization

Cleaned text data was converted to numerical data using the (term frequency-inverse document frequency) vectorizer that highlights unique words which are important in an individual course but infrequent in the complete data set which entails interesting semantic patterns suitable for clustering.

4.3.2.5. Clustering with KMeans

The TF-IDF vectorized data were clustered using KMeans clustering which produced seven clusters. The representative terms were analyzed to group the clusters into the top 10 representative terms and placed in broad thematic categories such as Computer Science, Health, Engineering.

4.3.2.6. Area Classification

The classification used a rules-based classification function to assign a course to a thematic area using the keywords from the clustering clusters. If the course matched the terms in the thematic cluster, it was labelled that area classification otherwise it was labelled as "Other".

4.3.2.7. Dataset Update and Visualization

The original dataset was updated with an extra Area column and saved to a new CSV file. The new dataset was subject to new visualization including bar plots and word clouds to show the structural balance of the dataset, showing the frequency and distribution of courses by thematic.

4.3.2.8. CNN Input Structure Preps for Training

After classification there are five columns in the dataset: CourseTitle, Description, Availability, Cost, and new Area. For CNN training, the course description was concatenated and then became the model's input. The input was fed into an embedding layer that turned the text into a numerical vector so the model could learn from the text data. Finally, the courses were labelled to one of the eight finale categories: Computer Science, Health, Mathematics, Humanities, Engineering, Food Science, Education, and Other.

4.3.2.9. Dataset Splitting

As it was important for the model to train and evaluate input datasets, the dataset was split into three separate datasets (datasetX_train=75%, datasetX_val=5%, datasetX_test=25%). A single seed value of 42 for reproducibility ensured the data was re-split the same way for the training and evaluations.

4.3.2.10. Experimental design

The CNN model was run in different configurations to better understand its learning behaviour and classification accuracy. The experiments were also run with a batch size of 32, a validation split of 5%, and over ^{2}& for each configuration, training and test accuracy, and training and test loss were recorded for assessment of the model performance. Later LOA (Lion Optimization Algorithm) was used to adjust hyperparameters to improve classification performance. In Figure 17 we will describe the dataset preprocessing.

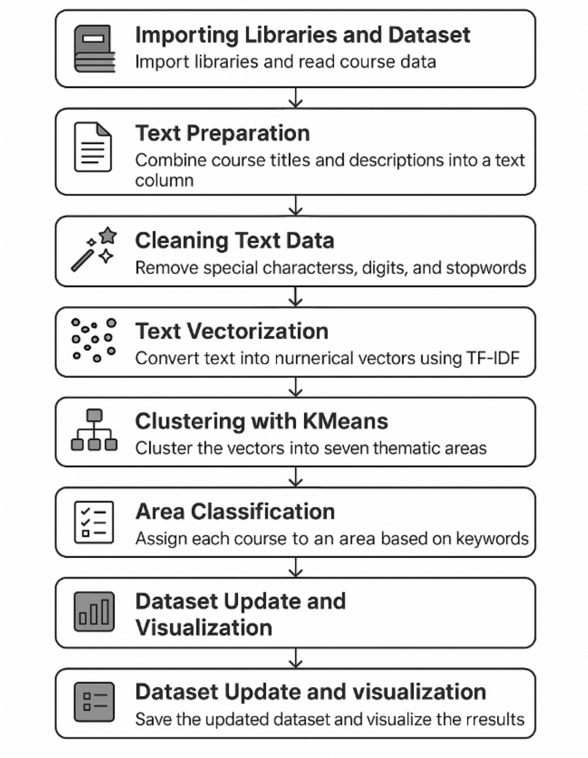


Figure 17: Overview of Dataset Preprocessing Workflow

4.3.3. Overview of the Proposed CNN-LOA Framework

Our contribution proposes a hybrid framework – CNN-LOA (Convolutional Neural Network with Lion Optimization Algorithm) which addresses the problems of course classification on MOOC engagement. The framework presented in the contribution utilizes the strong feature extraction capabilities from deep learning with CNN, combined with the adaptive and dynamic search capabilities of the LOA as a nature-inspired metaheuristic optimization algorithm. The main benefit discussed throughout this paper is automated hyperparameter tuning, which improves accuracy, scalability, and generalization capacity issues of classification-based models.

The optimization objective of the framework is formally expressed as follows:

$$\theta^{*} = arg\ \min_{\theta} L(f\ (X, \theta), Y)) \qquad \text{Equation 1}$$

Where:

$\theta = \{\eta, p, F, K, E\}$ represents the set of hyperparameters (learning rate, dropout rate, number of filters, kernel size, epochs).

$f\ (X, \theta)$ is the CNN model with input X and hyperparameters θ.

Y is the ground truth labels.

L (·) is the loss function, which LOA minimizes to optimize θ.

θ^* is the optimal set of hyperparameters found by LOA.

4.3.3.1. Motivation for the Hybrid Framework

MOOCs present high-dimensional, unstructured, and dynamic textual data. Therefore, MOOCs require something other than traditional rule-based and static machine learning models because they lack the ability to keep up with dynamic course descriptions and a variety of semantic structures. CNNs are appropriate for text classification problems since they can capture hierarchical levels and contextual structure from text data. However, they rely on hyperparameters, which include learning rate, dropout rate, filter size, kernel size, and depth/number of layers, and their performance is sensitive to each of these hyperparameters.

Tweaking hyperparameters manually typically is inefficient, and it produces suboptimal models. This highlights the importance of LOA. In nature, lions demonstrate social behavior, which is a strategy of territorial behavior, and LOA can produce global optimization in complex, multimodal search spaces. It uses the exploration-exploitation trade-off of LOA to automatically find the best configuration of a model without manual trial and error.

4.3.3.2. Framework Architecture and Workflow

The CNN-LOA architecture works in several stages, seen in Figure 18, described as follows:

- **Data Preprocess and Feature Encoding**

MOOC course metadata (for e.g., titles, short description), is collected and adheres to a preprocessing process including cleaning, tokenization, stop-word removal, and vectorization with TF-IDF or word embedding. This transforms the textual input to numerical vectors suitable for deep learning.

- **CNN Initialization and Model Specification**

The starting CNN architecture is specified with representative embedding layers, convolutional (and possibly max pooling) layers, dense layers, and output classification layers. These layers will learn local and global features of course categories.

- **Population Initialization (LOA)**

Population of lions are established, where each lion encodes a set of suggested CNN hyperparameters as:

$$\theta = \{\eta, p, B, F, K, P, D, E\} \qquad \text{Equation 2}$$

Where:

η: Learning rate.
p: Dropout rate.
B: Batch size.
F: Number of convolutional filters.
K: Kernel size
P: Pooling size.
D: Number of dense units.
E: Number of epochs.

- **Exploration and Exploitation: LOA Core Mechanisms**

Each lion (candidate solution) is assigned a fitness value based on CNN model performance, typically measured using classification accuracy or loss on the testing set.

- **Hunting:** Female lions use perturbations to hyperparameters to jointly explore the nearby region of high-performing solutions.

$$\theta' = \theta + N(0, \sigma) \qquad \text{Equation 3}$$

Where:

$N(0, \sigma)$ represents a small perturbation. If the new configuration improves model accuracy, it is accepted as the updated solution.

- Mating and Mutation: New offspring (hyperparameter configurations) come from crossover and mutation of existing solutions and will ensure that diversity in the population is maintained.

$$\theta_{offspring} = \alpha\theta_{male} + (1-\alpha)\theta_{female} \qquad \text{Equation 4}$$

Where:

α is a randomly assigned weight. Mutation is then applied to introduce variability and avoid premature convergence.

- Migration: Weak lions are prompted to explore remote areas of the hyperparameter space, to avoid local optima.

$$\theta_{new} = \theta_{old} + U(-\delta, \delta) \quad \text{Equation 5}$$

Where:

$U(-\delta, \delta)$ is a uniform noise factor ensuring sufficient exploration.

- Takeover and Selection: The weakest solutions are replaced with the strongest individuals (nomadic lions), promoting ongoing performance evolution to optimal configurations.

$$\theta_{best} = arg\overset{max}{\underset{\theta}{}} F(\theta) \quad \text{Equation 6}$$

Where:

$F(\theta)$ represents the validation accuracy of the CNN model with hyperparameters θ.

- **Termination Check**

The optimization loop continues for either a defined number of generations or termination conditions (e.g., no significant improvement in accuracy) are satisfied.

- **Final Model Choice and Training**

The lion with optimal fitness is selected to produce the final CNN model, using its respective hyperparameters. The final CNN model is then assessed for performance against the test set.

The frequent use of an iterative process of CNN hyperparameter optimization, across generations, driven by updates in LOA, will output the optimum model based on the best testing accuracy at the end of the CNN epochs run time. This is summarized below:

- ✓ Populate lions (CNN hyperparameters).
- ✓ Evaluate the fitness of each lion (the model performance).
- ✓ Local search to improve hyperparameters (hunting).
- ✓ Cross breeding of the best performing hyperparameters (Mating and Offspring Generation).

- ✓ The lions explore new parameter spaces (Migration).
- ✓ Lions' take over the weak lions with stronger solutions (Takeover and Selection).
- ✓ Repeat until convergence or a specified number of iterations has been reached.
- ✓ Train the last CNN model using optimal hyperparameters.

CNN/LOA Pseudocode

```
Function CNN_LOA ()
    Initialize CNN model architecture
    Initialize Lion Optimization Algorithm (LOA) with population size and max generations
    For each generation do
        For each lion in the population do
            Evaluate fitness of the lion by training the CNN model with its hyperparameters
            If fitness improves, update the lion's position in the population
            Perform migration step for exploration and diversity (if necessary)
        End for
        Select the top-performing lions to generate new offspring using mating
        Apply mutation to the offspring to introduce diversity
        Replace the weakest lions in the population with the offspring
    End for
    Return best CNN model with optimized hyperparameters
End Function
```

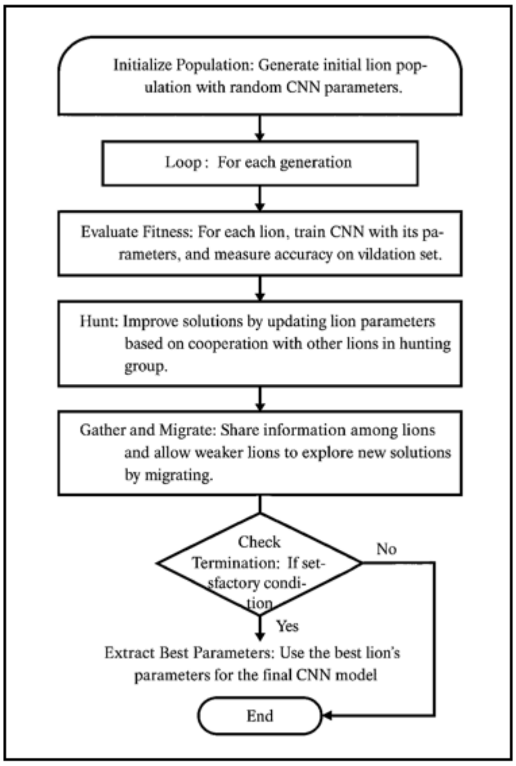


Figure 18: CNN/LOA architecture.

4.3.4. Convolutional Neural Network Design

The classification module of the proposed system is a Convolutional Neural Network (CNN) for text classification in the domain of MOOCs in Figure 19. Although CNNs were originally developed for processing images, CNN's pattern recognition capabilities as they relate to local and position-invariant features are instrumental for natural language processing (NLP) tasks, so we leverage CNN to identify useful syntactic and semantic linguistics features from course descriptions and related metadata. By learning discriminative textual patterns through the convolutional layers of the model, we have confidence that we will be served well for future classifications in multiple categories of MOOCs. The architecture has several components which are important for operation and fit, as outlined below:

- ✓ **Embedding Layer:** The embedding layer converts each word in the input sequence to a dense vector representation. The dense vector representations or embeddings create a representation that captures both syntactic and semantic relationships between words. These relationships will allow modeling to create good features downstream. Pre-trained embeddings or learned embeddings can be used, depending on dataset size and quality.
- ✓ **Convolutional Layer (Conv1D):** The embedded text undergoes the application of a one-dimensional convolution to extract local features - local features like n-gram patterns or

phrase-level structures. With respect to local features, many filters are learned in such a way to discover meaningful patterns that are useful for classification tasks.

- ✓ **Pooling Layer:** A pooling layer is utilized to reduce the spatial dimension of the feature maps in the form of a max pool. This lets the model consolidate the most important features and reduce computational load while controlling overfitting.
- ✓ **Flatten Layer:** Finally, the flatten layer converts the pooled feature maps to a one-dimensional vector compatible for the input into a series of fully connected layers. The flattening layer takes the convolutional model's feature extraction step and maps it to the classification aspect of the model.
- ✓ **Fully Connected (Dense) Layer:** This layer takes the features extracted by CNN, and it learns complicated relationships between the features. Conceptually think of this as providing the high-level reasoning of the model.
- ✓ **Dropout Layers:** Between the Dense layers, we also include dropout layers that randomly drop a proportion of the neurons during training. Dropout is a regularization method that promotes generalization to unseen data, and it helps to alleviate overfitting.
- ✓ **Output Layer:** Finally, we use a softmax activated dense layer at the output. The softmax layer gives a probability distribution over the defined course categories, and the course with the highest probability is taken as the model prediction.

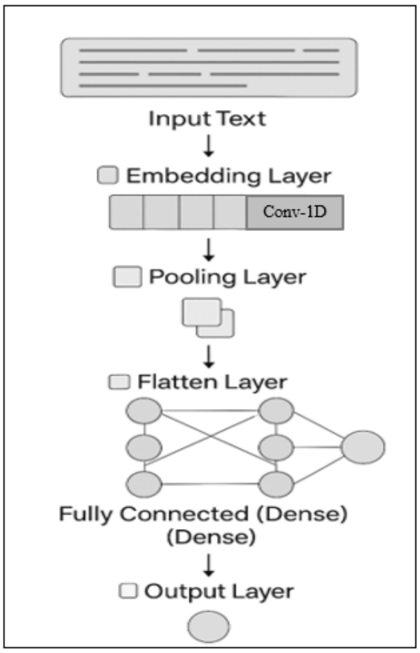


Figure 19*:* CNN architecture.

4.3.5. Lion Optimization Algorithm (LOA)

The Lion Optimization Algorithm (LOA) is a metaheuristic algorithm that is inspired by nature, which is based on the social structure and hunting behaviour of lions and pride mentality. Lion Optimization Algorithm is known to be effective in solving hard optimization problems, LOA is based on territorial and hierarchy interactions between lions, along with co-operative hunting and adaptive survival strategies.

LOA translates key behaviours of lions such as pride formations and nomadic groups, mating and reproduction, territorial takeovers and survival of the fittest into a form of intelligent optimization. LOA utilizes these components to provide a dynamic balance between exploration (diversifying the search space) and exploitation (intensifying the search around promising solutions), providing LOA with a sound and competitive algorithm in the evolutionary computation field.

The main stages of the LOA process are listed in Figure 20 and explained as follows:

- ✓ **Population Initialization:** The LOA will randomly initialize a population of lions, which are divided into prides (territorial groups) and nomads (wandering lions). The males are typically the protectors of the pride (as dominant) and females do the hunting.
- ✓ **Prides (Territorial Lions):** The pride stays in their territorial area and represents solutions that have been locally exploited.
- ✓ **Nomads (Wandering Lions):** The nomads roam through the search space to enhance population diversity and promote global exploration.
- ✓ **Hunting mechanism:** The hunting mechanism is designed to simulate the exploitation phase in which lions (solutions) attempt to hunt prey (better solutions). The lions outlined a random walk strategy around the best current positions. A lion can update its position only if the new candidate solutions produce better fitness than previous ones. The fitness function determines whether to accept or reject a new solution.
- ✓ **Deployment and Breeding:** In prides, the dominant male lion will breed with multiple female lions. Their offspring will possess a combination of genetic traits inherited from both parents, which generally occurs through a crossover process followed by a mutation to force diversity. Only the fittest offspring will survive, thereby encouraging the persistence of strong genetic traits in the population.
- ✓ **Migration Process:** To ensure that lions do not become stagnant or overfit the data, a migration process is applied between the prides and the nomadic groups. A small

percentage of lions may migrate depending on their performance, to either join the pride (if they increase its competition) or become nomadic (if they underperform). The migration process helps with global search capabilities and maintains a diverse search to ensure convergence to a local optimum is avoided.

- ✓ **Nomadic Takeover and Survival:** Strong nomadic males can become overly competitive to the point of challenge and replacing the dominant males in the prides, simulating natural selection. Only the most competitive males gain reproductive roles, thereby strengthening the populations with better potential solutions.
- ✓ **Select the best lions:** After generations have passed, lions perceived to be weak enough to lose their contest to the other lions are removed from the population, retaining the best solutions. This elitist selection requires an additional process to shortly determine the best lion population, in the hopes of improving population fitness.

The method continues iteratively until the termination criterion is met. Either the pre-determined number of iterations is equal to the maximum set, or a specific fitness value has been discovered (a satisfactory minimum). The best solutions or populations developed as a result of the generationally evolving method is returned to the researcher.

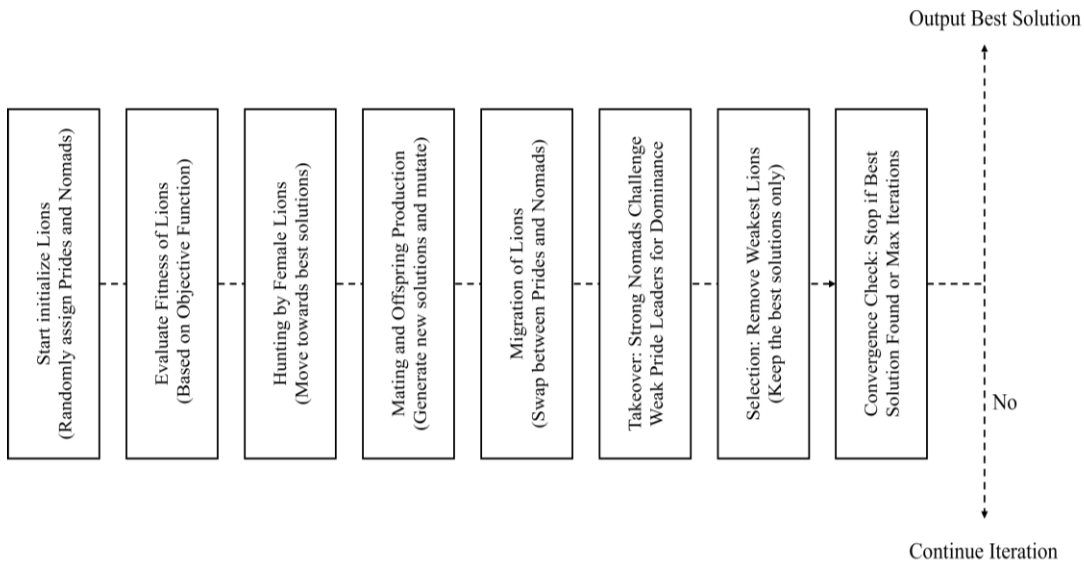


Figure 20: The flowchart of LOA

4.3.6. Experimental Setup

4.3.6.1. Dataset and Input Representation

This segment contains the experimental steps taken to analyze the performance of the proposed

Convolutional Neural Network (CNN) model developed using the Lion Optimization Algorithm (LOA) for MOOC course classification. Design, configuration, dataset, and implementation details are provided in the following subsections.

The dataset used for the experimental evaluation of the model is comprised of MOOC course records including these attributes: CourseTitle, Description, Availability, Cost, and Area. Of these attributes, the course descriptions were the primary feature input used for classification, and Area label specified the target class for each course. The model was trained to classify each course in one of the eight domain categories: Computer Science, Health, Mathematics, Humanities, Engineering, Food Science, Education, Other.

Before inputting the text into the model, it underwent preprocessing, such as tokenization, lowercasing, stopword removal, and padding so that all sequences had the same length. The preprocessed text sequences were then sent to an embedding layer, that transformed each word into a dense vector representation that allowed the CNN to encode these words into syntax and semantics relate to one another.

4.3.6.2. Model Configuration and CNN Architecture

The architecture of the CNN in Figure 21 was intended for text categorization using a one-dimensional convolutional pipeline. The architecture consists of the following key layers:

- Embedding Layer used to convert the text to vectors.
- Conv1D Layer used for extracting local features (e.g., n-gram features).
- MaxPooling1D Layer used for dimensionality reduction and for reducing the possibility of overfitting.
- Flatten Layer used for converting the feature maps to a one-dimensional vector.
- Fully Connected Dense Layers used for learning high-level abstract representations.
- Dropout Layers placed between dense layers to potentially improve generalizability.
- Output Layer used to perform multi-class classification, with a softmax activation function.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding (Embedding)       (None, 36, 100)           500000

 dropout (Dropout)           (None, 36, 100)           0

 conv1d (Conv1D)             (None, 34, 64)            19264

 max_pooling1d (MaxPooling1  (None, 17, 64)            0
 D)

 flatten (Flatten)           (None, 1088)              0

 dense (Dense)               (None, 64)                69696

 dropout_1 (Dropout)         (None, 64)                0

 dense_1 (Dense)             (None, 5)                 325
```

Figure 21: the architecture of the CNN/LOA hyperparameters layers.

4.3.6.3. Hyperparameter Optimization using LOA

To ultimately get the best possible performance from the model, the Lion Optimization Algorithm (LOA) was used to automatically find the optimal hyperparameter. The hyperparameters listed below in Table 6 were tuned using LOA:

Table 6**:** Hyperparameter Ranges Used for Optimization via LOA

Hyperparameters	**Search Range**
Learning Rate	0.0001 – 0.01
Number of Filters	16 – 64
Kernel Size	3 – 7
Dropout Rate	0.2 – 0.5
Epochs	10, 15, 20, 30

LOA replicated lion behaviors (hunting, mating, migration) so that the hyperparameter values could be iteratively refined, maintaining a balance between exploration and exploitation in the parameter search space.

4.3.6.4. Training Configuration

The training setup for the CNN/LOA model was designed based on the following parameters:

- Batch Size: 32
- Validation Split: 5% of training data
- Loss Function: Categorical Crossentropy
- Optimizer: Adam (with LOA-tuned learning rate)
- Performance Measures: Accuracy and Loss on training and testing sets

Testing involved varying the number of epochs (10, 15, 20, 30) over four different configurations to examine learning development, overfitting tendencies, and generalization performance over time.

4.3.6.5. Implementation specifics and evaluation procedure

Implementation was conducted in Python and TensorFlow and the Keras API. LOA was implemented modularly to maintain interface compatibility in the CNN training pipeline.

Performance was defined relative to the following characteristics:

- Training and testing accuracy by epoch.
- Training and testing loss to observe convergence.
- Generalization gap between training and testing performance.
- Best epoch predicated on the best compromise between accuracy and overfitting.

Furthermore, accuracy and loss curves were plotted to help illustrate model learning tendencies and variability when the number of epochs was altered, as well as to provide some guidance help in aiding decision-making surrounding early stopping and optimal model selection.

4.3.7. Results

In this section, we report all the results of the undertaking of a CNN model optimized by using the LOA, for the classification of MOOCs. The section further includes discussion on the performance of the model over several epochs of training, the influence of LOA on fine-tuning hyperparameters, and our assessment of generalization ability and overfitting. Various visualizations, such as accuracy and loss curves, are included to supplement the analyses.

4.3.7.1. CNN/LOA Performance Across Epochs

The CNN/LOA model was trained and tested using four different numbers of epochs (10, 15, 20, and 30) as shown in Table 7 and Figure 22 to examine how learning progressed and the model's ability to generalize. For each of the epoch configurations, training and testing accuracy and loss values were recorded and examined.

Table 7: CNN/LOA Performance Across Different Epochs

Epochs	Training Accuracy (%)	Test Accuracy (%)	Training Loss	Test Loss	Observation
10	94.11	96.18	0.1981	0.1384	- Good results with small generalization gaps. - Learned well, no overfitting.
15	94.41	96.34	0.2014	0.1457	- Best test accuracy. - The best trade-off between learning and generalization.
20	93.59	96.22	0.2203	0.1377	- Training performance is still dropping slightly. - Also, the model generalizes well.
30	89.32	97.27	0.4303	0.1620	- High test accuracy but learning rate still dropped a lot.

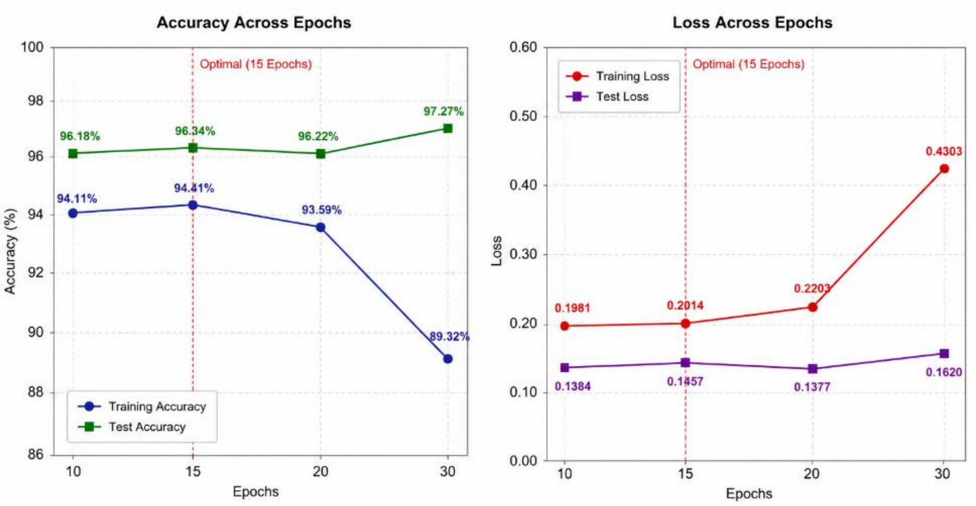


Figure 22: Model Performance Across Different Epochs.

Epoch 15 was considered optimal based on all configurations, and the model demonstrated the highest test accuracy while optimizing learning speed and generalization speed.

To assess if the model is learning properly and not overfitting, it is important to look at the trends for training and testing accuracy / loss across epochs. A learning curve ideally will show the loss decreasing for both training and testing while the accuracy is increasing. At Epoch 10 (Figure 23), we saw that the model learned to be 94.11% accurate when trained, 96.18% accurate when tested on an unseen dataset, and a training and testing loss was 0.1981 and 0.1384, respectively. This data suggests that the model had a reasonable level of learning of the features present in this classification problem and was not overfitting to the training dataset. This performance level improved again further in Epoch 15 (Figure 24), and the model achieved test accuracy of 96.34% with a balance between training accuracy (94.41%) and loss values (0.2014 and 0.1457) making this configuration the most stable of those observed thus far. At Epoch 20 (Figure 25), whilst test accuracy remained high (96.22%), training accuracy dropped (93.59%) and training loss rose (0.2203) - possibly suggesting the presence of underfitting. Finally, Epoch 30 (Figure 26) showed the maximum test accuracy (97.27%) the model was able to achieve and the most inconsistent drop in training accuracy (89.32%) with the most training loss (0.4303). The inconsistency supports prior evidence of underfitting in the model that was likely due to how little learning was occurring.

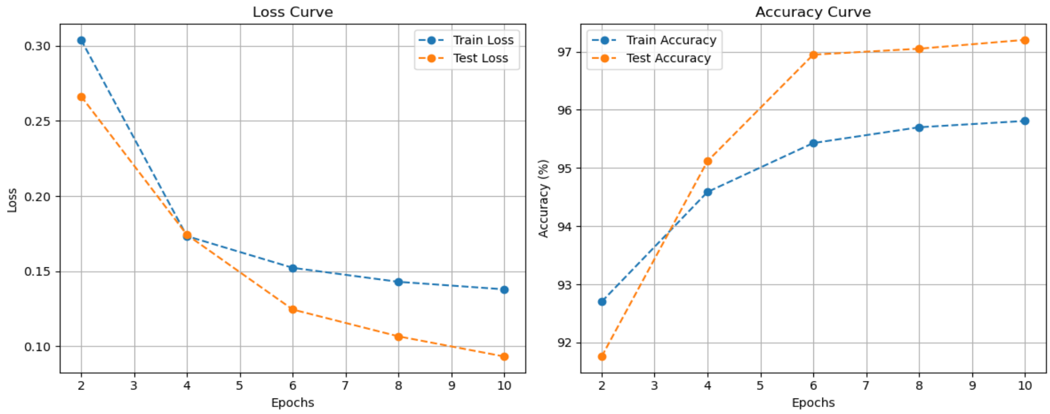
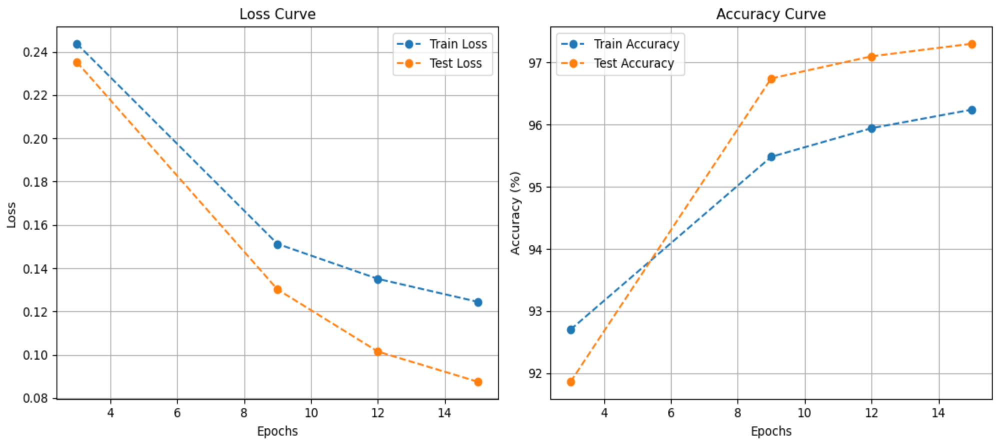


Figure 23:Accuracy and Loss curves with 10 Epochs.



Erreur ! Source du renvoi introuvable. Figure 24: Accuracy and Loss curves with 15 Epochs.

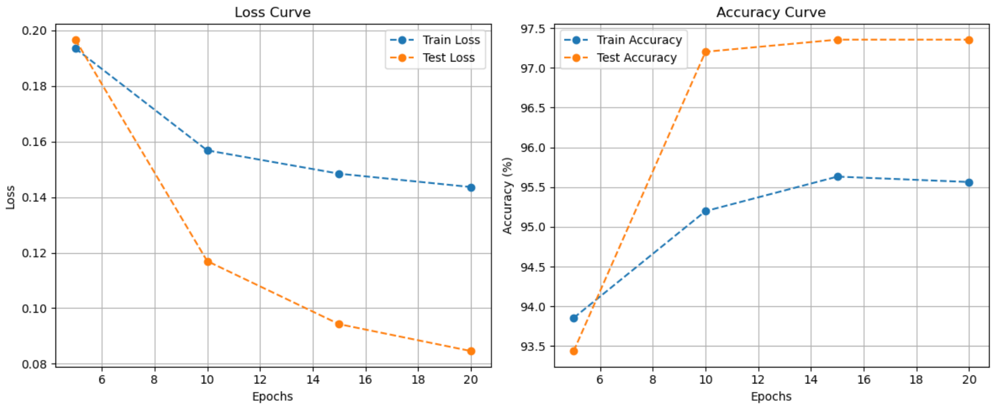


Figure 25: Accuracy and Loss curves with 20 Epochs.

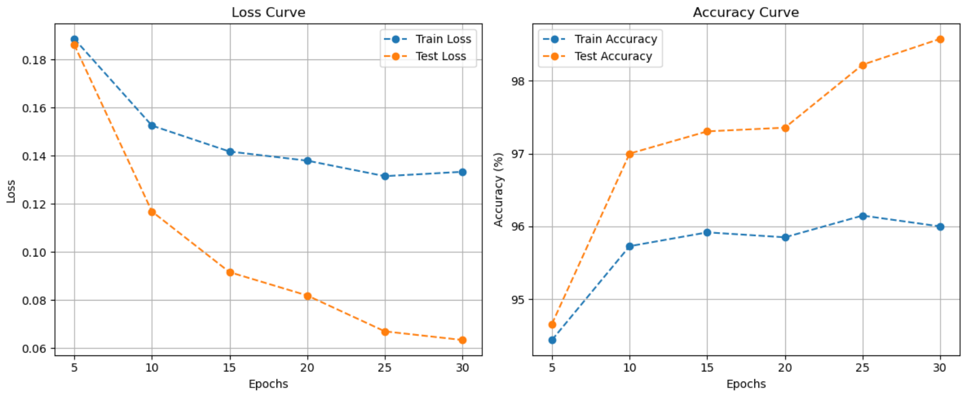


Figure 26: Accuracy and Loss curves with 30 Epochs.

Remove this Therefore, the progression through the epochs of experimentation clearly illustrates that 15 epochs provides the most desirable balance of convergence, generalization and training efficiency.

4.3.7.2. Comparative Evaluation with Baselines

In order to assess how efficient, the proposed CNN model optimized using the Lion Optimization Algorithm (CNN/LOA) was compared with traditional deep learning models and some different biologically inspired optimization models, a comparative assessment was made regarding MOOC course classification that evaluated the advantages and disadvantages of each model on the following metrics: training accuracy and test accuracy while measuring the associated training loss and test loss.

- **Comparison to Traditional Deep Learning Models**

Two traditional deep learning architectures were selected as baselines: CNN-BERT and CNN-LSTM. Each model was trained on the same dataset using the same data preprocessing and evaluation protocol.

CNN/BERT exhibited training and test accuracy at 91.26% and 93.37%, respectively, with a test loss of 0.2515(Figure 27). The main advantage of BERT is it allows the use of pre-trained contextual embeddings. However, it suffers from its lack of flexibility to the domain due to its static embeddings model. Whereas the CNN/LOA model had dynamically learned parameters that reached a higher test accuracy, 96.17% and lower test loss of 0.2496. The CNN/LOA model better attributes generalization and accommodation to the domain-specific contexts in the course descriptions.

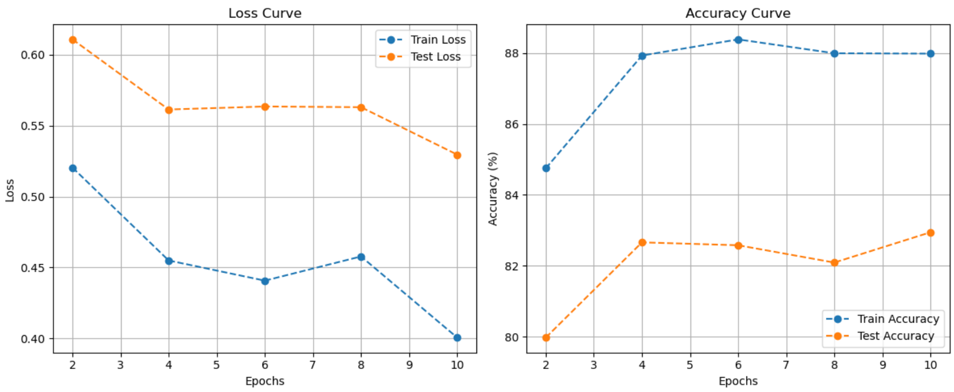


Figure 27**:** Accuracy and Loss curves with CNN/BERT approach

CNN/LSTM achieved a training accuracy of 92.29%, but a test accuracy of only 91.21, with a test loss of 0.3216 (Figure 28). LSTMs can generate good results on sequential data. Industry criticism has cited their focus on long-term dependencies of duration can lead to overfitting on rigorously structured and concise text (like MOOC descriptions). On the other hand, CNN/LOA demonstrated more consistency regarding performance and generalization.

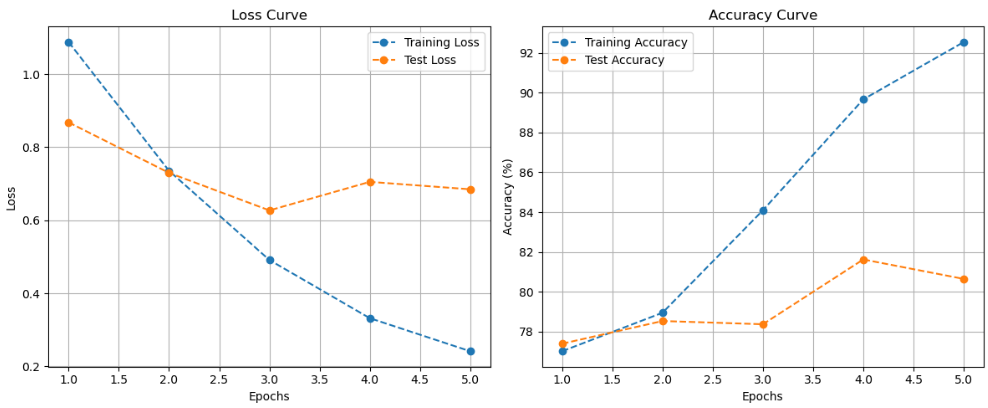


Figure 28**:** Accuracy and Loss curves with CNN/LSTM approach

- **Comparison with Bio-Inspired Optimization Algorithms**

Various bio-inspired algorithms were employed for CNN hyperparameter optimization: Whale Optimization Algorithm (WOA), Ant Colony Optimization (ACO), and Particle Swarm Optimization (PSO). For evaluation purposes, performance was compared with the CNN/LOA model using the same configurations and evaluation metrics.

CNN/WOA achieved test accuracy of 84.13% and test loss of 0.5033. WOA is a good global search algorithm Figure 29, but it has sparser convergence and weak fine-tuning capacity, all of which diminished its effectiveness in complex hyperparameter spaces.

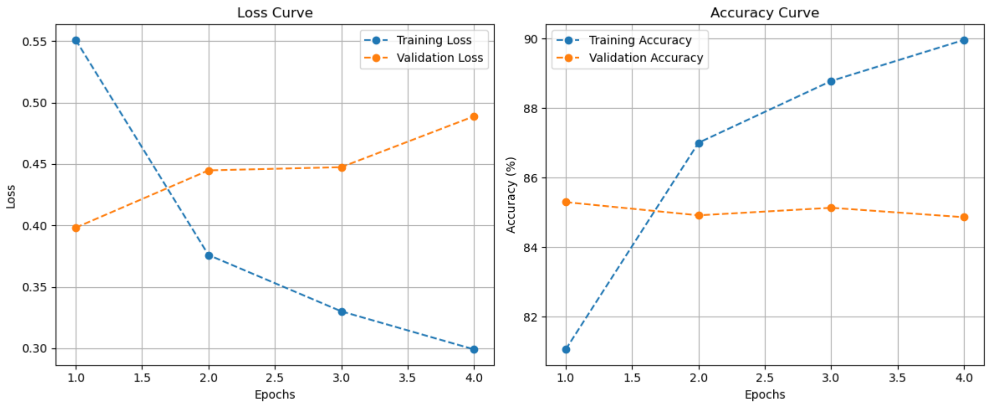


Figure 29: Accuracy and Loss curves with CNN/LWOA approach

CNN/ACO achieved test accuracy of 85.23% and test loss of 0.4140 Figure 30. For purposes of comparison, much like WOA, ACO optimized hyperparameters with a directional component, ACO is good at optimizing discrete optimization but not continuous hyperparameter spaces, which is part of the characteristics associated with CNNs.

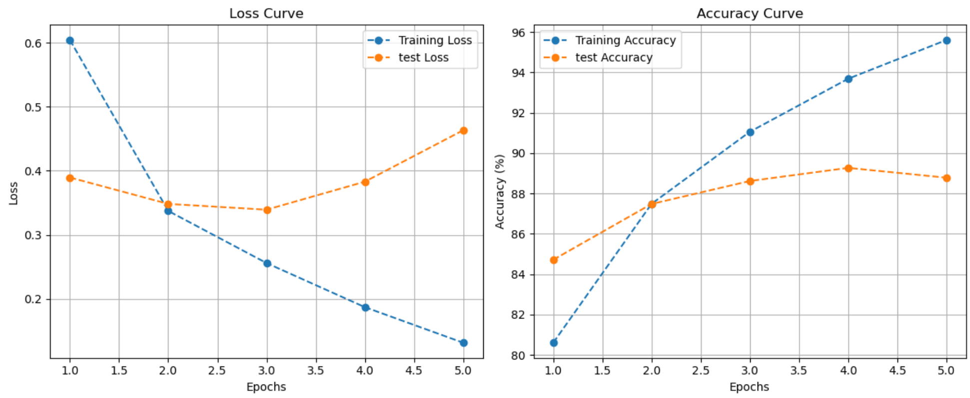


Figure 30: Accuracy and Loss curves with CNN/ACO approach

CNN/PSO performed the worst, yielding test accuracy of 84.57% as opposed to test loss of 0.5736 Figure 31. PSO was very fast, but it suffers from premature convergence to local optima points of interest in high-dimensional spaces.

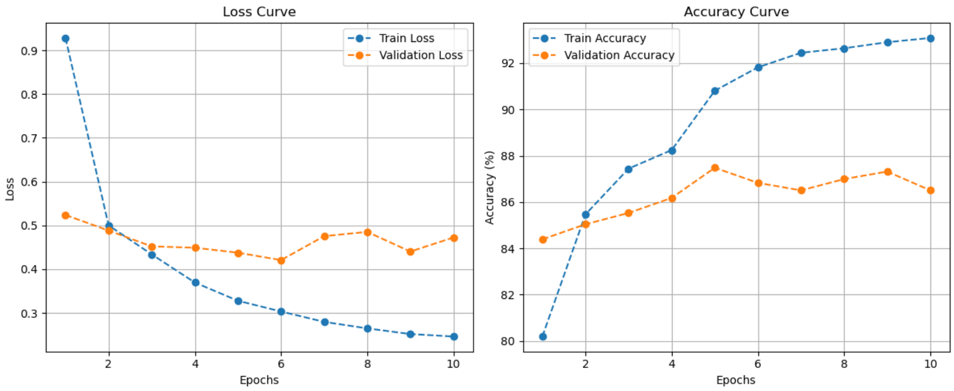


Figure 31: Accuracy and Loss curves with CNN/PSO approach

The proposed CNN/LOA algorithm achieved a better test accuracy (96.17%) than all other algorithms, and it had a better generalization capacity. LOA is recognized for its hybrid solution, which strikes a balance in utilizing global exploration and local exploitation. This way of adaptive behavior allowed the algorithm to tune important parameters including learning rate, dropout rate and filter size in a more efficient manner, which produced a robust and generalizable model.

According to Table 8 and figure 32, CNN/LOA consistently yielded the most favorable results relative to both conventional baselines and bio-inspired baselines. This quantitative performance is grounded in the qualities of low-test loss, well-behaved generalization and lower test loss with low risk of overfitting. As such, CNN/LOA is a reliable model that can be used to produce scalable and accurate course classification when evaluated in MOOC contexts.

Table 8: Results of CNN/LOA with Traditional Deep Learning Algorithms and Bio-inspired Algorithms.

Metrics / **Approach**		**Train Loss**	**Train Accuracy**	**Test Loss**	**Test Accuracy**
Traditional Deep Learning Algorithms	**CNN/BERT**	0.3037	91,26%	0.2515	93,37%
	CNN/LSTM	0.2745	92,29%	0.3216	91,21%
Bio-inspired Algorithms	**CNN/WOA**	0.2547	91,96%	0.5033	84,13%

	CNN/ACO	0.2329	91,95%	0.4140	85,23%
	CNN/PSO	0.2351	93,98%	0.5736	84,57%
Our Algorithm	**CNN/LOA**	**0.3330**	**91,85%**	**0.2496**	**96,17%**

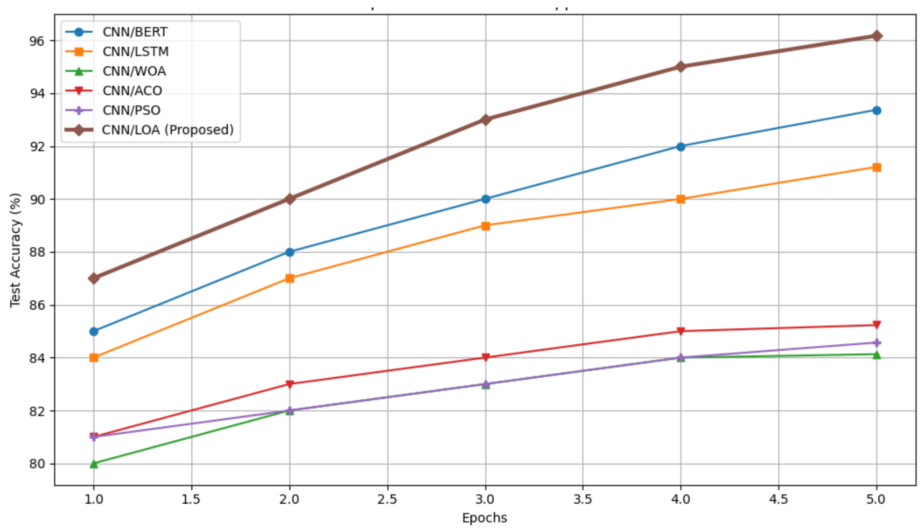


Figure 32: Test Accuracy curves CNN/LOA compared with other models.

4.3.8. Discussion

The experimental findings support the ability of the proposed CNN in collecting Lion Optimization Algorithm (CNN/LOA) approach to classify MOOC courses into thematically similar categories. The model was able to achieve strong performance throughout the training epoch with a clear tradeoff of learning overtime and generalization. The model performed best at Epoch 15, with test accuracy reaching 96.34% and training accuracy at 94.41%, indicating a good balance and less overfitting (1.93% gap).

The increase in training loss and decrease in accuracy when training beyond 20 epochs indicated that the environment was still important and that stopping criteria would be essential to implement. The impact from the LOA was significantly important for the hyperparameters - learning rate, dropout, and kernel - providing a reasonable balance of exploration and exploitation essentially allowing CNN to more effectively learn and generalize. The ratio of the test loss was very low and remained consistent throughout with a drop from 0.1384 at Epoch 10 to 0.1377 on Epoch 20.

Later in the training cycle, while test accuracy persisted, there was an increase in training loss (0.1981 to 0.2203) against the risk of overfitting, reaffirming the utility of early stopping strategies. Overall, the classification model developed using CNN/LOA maintained a level of generalization without sacrificing accuracy, ensuring it will be viable for classification tasks in the real-world context of unseen MOOC data.

The alphabetical comparisons were also made against the two bio-inspired optimization algorithms (namely WOA, ACO, and PSO) and the other end of the continuum of deep learning that was explored, whilst considering CNN/BERT and CNN/LSTM. Throughout the classification performance, TA (CNN/LOA) always outperformed classification performance from both bio-inspired Optimization algorithms, and traditional deep learning models. Specifically, it achieved the highest test accuracy of 96.17% and the lowest test loss of 0.2496, over the last epochs, whilst the other models produced test accuracy scores ranged from 84.13% through to 85.51 %.

Methods from biology, such as ACO, experienced problems in speed of convergence and premature local minima impacting performance. On the other hand, CNN/BERT (93.37% accuracy) and CNN/LSTM (91.21% accuracy) did well in mapping from complex or collated educational datasets, but had issues adapting to domain and overfitting, given the structured or describe nature of MOOC descriptions. This meant that while they could handle text and classify it, this also helped them to classify MOOC descriptions but not training from the dataset. With CNN/LOA, we showed that (through LOA) CNN could adapt learning. Adaptive hyperparameter tuning through LOA was greater than each of the baselines again, suggested their effectiveness for complex classification of educational datasets.

4.3.9. Study Constraints and Prospective Research Paths

While the CNN/LOA model showed good predictive performance for classifying MOOC courses, there are limitations that provide potential research avenues. First, it was evaluated from a specific dataset that should be expanded to determine its generalizability for additional education content and languages and from different fields and platforms to evaluate its robustness. Second, while the Lion Optimization Algorithm has significantly increased classification performance through its direct hyper-parameter tuning capabilities, the computational burden is still relatively high. While it can always be improved through parallel computing or simpler optimization processes, there are other suggestions that could be considered for future research. For example, it is recommended that the CNN/LOA be hybridized to transformer-based models like BERT or GPT to improve the identification of context to enhance classification accuracy. Lastly, the current model is primarily a black box; if XAI was implemented, it would improve interpretability and therefore allow for increased transparency into the model's decision-making process for educational required systems.

In conclusion, the study proposed an efficient deep learning framework implemented for MOOC course classification, combining a Convolutional Neural Network (CNN) with a Lion Optimization Algorithm (LOA) as an adaptive hyperparameter tuning technique. This study conceived this framework primarily due to challenges of organizing large scale and heterogeneous content of MOOCs. The overall framework aimed to bring accuracy and robustness to the automatic categorization of online courses, which also contributes toward better management of educational content, and enhanced learning experiences due to improved personalization. The methodological approach involved extensive data preprocessing tailored to the linguistic characteristics of titles and descriptions of courses, involving activities such as case normalization, removal of stop-words, and generation of n-grams to build richer contextual representations of the course offerings. Additionally, relevant metadata was also extracted from the courses to improve the representation of course offerings. A CNN structure was then developed and optimized using LOA to tune the salient hyperparameters of the CNN, being the learning rate, filter size, and batch size. Experimental findings validated the superiority of the performance of the CNN/LOA over the considered baselines, as the CNN/LOA yielded a test accuracy of 96.34% and a test loss of 0.1457. These results supported the LOA's ability to exploit exploration and exploitation for improvements in model performance. The results of this research displayed more than only its classification performance. It demonstrated the importance of employing strong feature extraction methods with a flexible optimization method in the wider field of educational data mining and predictive analytics. The strong performance of the CNN/LOA has opened future avenues for the study of various bio-inspired optimization algorithms which can be applied to strategies around personalized and adaptive learning, recommendation systems, and greater educational data aggregations and analytics.

4.4. Personalized Course Recommendation using Hierarchical Attention and Contextual Language Models

As the volume and diversity of MOOCs on plurality of platforms is quickly expanding, it becomes harder to provide recommendations centered on specific learner needs and the context of learning. Traditional recommended systems fall short in capturing deeper semantics of courses and user choices in the context of limited interactions. In this section, we introduce a hybrid recommender system that combines hierarchical multi-head attention beyond traditional recommendation approaches [269] with using contextual language models - BERT made popular by past work - to improve course recommendations by prioritizing personalization and context. Our model (HMHA-C-BERT) has the better potential to model content semantics and user interactions and can provide greater recommendation quality and personalization.

4.4.1. Problem Statement and Goals

Current MOOC recommendation systems are beset with problems aligning content with learner preferences due to issues with limited user interaction data, weak semantic representations of the courses, and the characteristics of past recommendation configurations. Our goal is to tackle these challenges to develop a hybrid recommendation framework HMHA-C-BERT that utilizes contextual language modeling (semantic meaning) and hierarchical attention (levels of attention), together providing more personalized course recommendations, better semantic meaning from course content, and encouraging better generalizability of recommendations.

We propose the following research questions:

- Does the use of an HMHA-C model with BERT embeddings provide a better representation of MOOC content and user interaction semantically and contextually?
- Does hybridized content-based embeddings from user interaction data outperform traditional recommendation models?
- Does multi-level attention allow for greater generalizability over course categories and user preferences across course content?

Objectives of the Contribution:

- ✓ Develop a hybrid deep learning architecture (HMHA-C-BERT) capable of generating high-quality high-dimensional contextual embeddings of MOOC course content through nested BERT, BiLSTM, and hierarchical multi-head attention mechanisms.
- ✓ Improve the personalization of course recommendations by completing context- and content-based feature fusions with collaborative filtering signals now extended using an attention-based fusion module.
- ✓ Address challenges with cold-start and data sparse factors that lecturer recommenders are all too familiar with, with content-based and collaborative recommendations

supported with contextual language models and evolving user preferences as motivational by users' previous interactions with online courses.

✓ Achieve state-of-the-art accuracy and generalization performance on course classification and course recommendation tasks evidenced by comparative empirical studies against popular baseline models (CNN, LSTM and NCF).

✓ Provide an adaptive and scalable AI-based recommendation framework with improved capabilities to deepen learner engagement and facilitate better choice and decision-making in online learning contexts.

4.4.2. Overview of the Proposed HMHA-C-BERT Framework

To tackle the complexity of low user interaction, the unknown context of recommendation, and limitations related to collaboration and content signals in MOOC recommender systems, we define a novel hybrid system called HMHA-C-BERT (Hierarchical Multi-Head Attention with Contextual BERT) shown in Figure 33. This is the first native deep contextual language model that combines sequential models with attention models, to generate semantically rich, user-aware course recommendations.

The HMHA-C-BERT framework consists of two components:

- HMHA-C-BERT Embedding Component
- Hybrid Recommendation Component

The HMHA-C-BERT Embedding Component creates high-quality, contextualized representations of course content. The embedding component uses pretrained BERT embeddings to generate deep semantic features of the course title and description concatenated together. An embedding of this title-description use-case is sent through a Bidirectional LSTM layer to model the sequential nature of the course descriptions and titles (in the past this consisted of the full MOOC course). A multi-head attention layer takes the inputs from the LSTM and attends to the different contextual levels of the courses simultaneously. A contextual compression layer is then applied to reduce the dimensionality while capturing the relevant semantic details, yielding compact course embeddings.

The Hybrid Recommendation Module combines these course embeddings with user interaction data (e.g., clicks, ratings, enrollments). Users appear as a learnable embedding vector. The attention mechanism calculates the closeness of user tastes to the pool of course embeddings and creates a combined embedding. This process infers the probability of interactions between each user and course. The combined embedding makes sure that recommendations are personalized, yet contextual.

Overall, these modules provide a flexible and scalable end-to-end architecture, which tightly integrates content-based invocations and collaborative signals. The recommendation engine is designed to be generalized to different course categories and is adaptable to learning interests of users, so that the system can enhance the accuracy, relevance, and personalization of MOOC course recommendations.

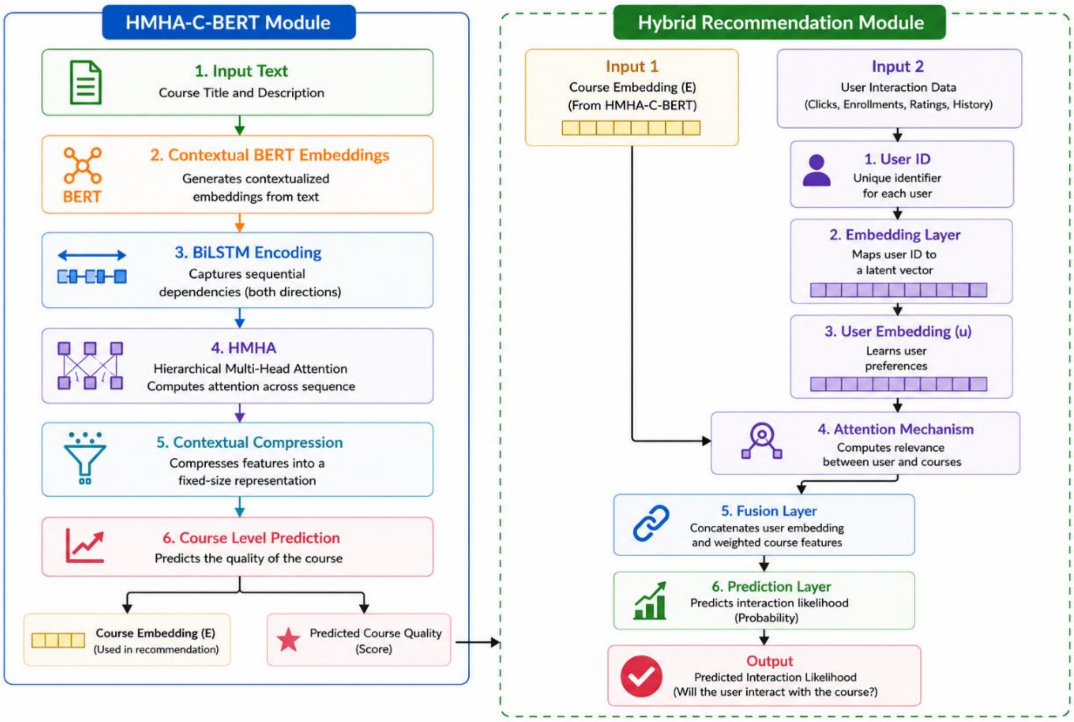


Figure 33: Overview of the HMHA-C-BERT Framework for Personalized Course Recommendations

4.4.3. Dataset Description and Preprocessing

The dataset for this research was part of a large-scale data collection of Udemy courses and user responses. As such, the dataset contains a wide variety of factors needed for a personalized course recommendation system. The dataset consists of two main parts:

- Course Information Dataset (Course_info.csv): Contains about 209,734 course metadata which consists of course titles, descriptions, categories, publication dates, language, and other listed characteristics across 13 categories and 79 languages.

- User Comments Dataset (Comments.csv): Contains over 9 million comments from users that represent a valuable source of implicit user feedback and taken together, user engagement patterns.

4.4.3.1. Preprocessing Steps

To support data quality and consistency for training models and for evaluating models, a number of preprocessing steps were implemented as demonstrated in Figure 34:

✓ **Cleaning User Comments:**

- Filling values and irrelevant entries were removed.
- All text fields were converted to lower case and normalized.
- HTML tags, special characters, and stopwords were eliminated.
- Punctuation was normalized consistently so that the timestamps were also normalized to a datetime format.

✓ **Processing Course Metadata:**

- Columns of data that were unrelated to the course description (e.g., instructor profile, price, URL) were dropped.
- Course titles and course descriptions were truncated, and tokenized.
- There were no formatting marks or indications to separate course titles from descriptions. However, the text fields were concatenated (title + description) for input to the contextual language model.

✓ **Merging Datasets:**

- Course_id was a foreign key to merging the course metadata file with the file containing user comments and the comments metadata file.
- This was helpful to the course attributes with the user comments for each interaction.

✓ **Topic Filtering:**

- To specialize in the recommendation task, dataset was filtered down to domain-specific courses, specifically focused on courses related to Technology, Business, Personal Development, and Data Science.
- The filtering was done to be able to provide domain specific examples and to keep training examples semantically consistent.

✓ **Normalizing Interaction Data:**

- The user-course interaction matrix was based on click behavior, enrollments, and comment metrics/directionality mapping frequency.

- These were normalized to a predictable region (0 -1) for reliable/usability signals of user interest.

4.4.3.2. Final dataset statistics:

Following all the data cleaning, filtering, and integration, the final training and evaluation dataset contained 325,443 high-quality records, each representing a unique course and its associated user interaction and content evidence. The dataset is used to serve as the dataset for both content embedding and collaborative modeling within the HMHA-C-BERT framework.

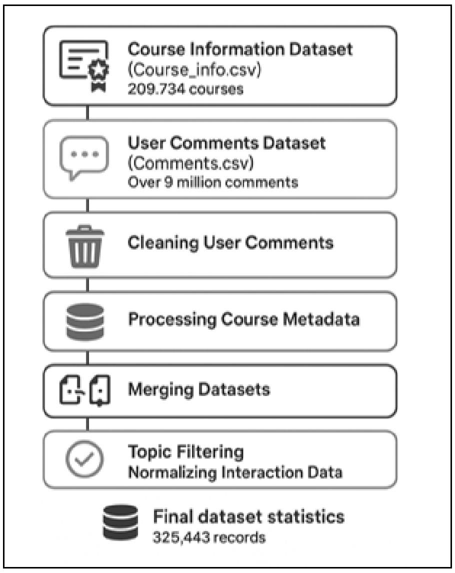


Figure 34: Data Cleaning and Feature Extraction Steps.

4.4.4. Hierarchical Multi-Head Attention and Contextual Embedding Design.

At the heart of the proposed HMHA-C-BERT framework is the ability to produce high-quality, semantically rich, and contextually relevant embeddings of MOOC course content. To achieve this, we use a hierarchical configuration that combines various deep learning components: BERT for semantic understanding, Bidirectional LSTM for sequential modeling, and a multi-head attention mechanism to highlight important features to produce representations that focus on salient information. This section highlights the element of the subsequent architecture and its components.

4.4.4.1. BERT-Based Embedding Extraction

Each course is represented as a concatenated sequence of title and description. This input text is tokenized to a sequence of tokens $X = X_1, X_2, \ldots, X_T$ with $X \in R^{T \times d}$ where T is the length of the sequence and d is the dimensionality of the BERT embeddings. Here a pre-trained BERT model encodes this sequence into contextualized embeddings:

$$H^{(0)} = BERT(X) \quad , \; H^{(0)} \in R^{T\times d} \qquad \text{Equation 7}$$

BERT's bidirectional attention captures subtle semantic relationships between tokens, allowing effective modeling of polysemy and context dependence in course texts.

4.4.4.2. Sequential Modeling with Bidirectional LSTM

The contextual aware embeddings generated from BERT are sent to a Bidirectional Long Short-Term Memory (BiLSTM) layer, so that the model learns from both the past and future in the sequence:

$$H_t^{(1)} = BiLSTM\left(H_t^{(0)}\right), \quad t = 1, \dots, T \qquad \text{Equation 8}$$

The outputs of both the forward and backward passes from the BiLSTM are concatenated to form a tensor $\mathrm{H}^{(1)} \in \mathrm{R}^{\mathrm{T}\times\mathrm{h}}$ where h is the number of hidden units. Dropout regularization is then applied:

$$H_{drop}^{(1)} = Dropout\left(H^{(1)}\right) \qquad \text{Equation 9}$$

This helps reduce overfitting and increase generalization.

4.4.4.3. Hierarchical Multi-Head Attention (HMHA)

In order to refine the embeddings even more, we employ a hierarchical multi-head attention mechanism. For each of the K heads, we do the following:

$$head_i = Attention\left(H_{drop}^{(1)} W_i^{Q}, H_{drop}^{(1)} W_i^{V}, H_{drop}^{(1)} W_i^{V}\right) \qquad \text{Equation 10}$$

where $\mathrm{W_i^Q}, \mathrm{W_i^K}, \mathrm{W_i^V}$ are learnable weight matrices for the query, key, and value projections. The attention is computed in the same way as the standard dot-product attention:

$$(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_K}}\right) V \qquad \text{Equation 11}$$

The individual outputs of each head are concatenated into one vector and linearly transformed:

$$H^{(2)} = Concat(head_1, \dots, head_K) W_O, \; where \; H^{(2)} \in R^{T\times d_{attn}} \qquad \text{Equation 12}$$

This mechanism allows the model to attend to different perspectives of the course text simultaneously and extract deeper and more diverse semantic signals.

4.4.4.4. Contextual Compression

The last component of the contextual embedding design is to compress the attended features into a fixed-size vector representation:

$$v = MeanPooling\left(H^{(2)}\right),\ E = ReLU\left(W_{comp}v + b_{comp}\right) \qquad \text{Equation 13}$$

Where $E \in R^{d_E}$ is the final compressed course embedding,

$W_{comp} \in R^{d_{attn} \times d_E}$ and B $b_{comp} \in R^{d_E}$ are learned parameters. This course embedding is a concise and informative representation of the course that still encapsulates the most pertinent semantic or contextual meaning.

4.4.4.5. Course-Level Prediction

To assess the quality of the representations, an optional classification layer can be added for downstream tasks like predicting categories:

$$\hat{y} = \sigma(W_{cls}E + b_{cls}) \qquad \text{Equation 14}$$

Where σ is the sigmoid activation function, and W_{cls}, b_{cls} are the parameters of the classification layer.

HMHA-C-BERT Pseudocode
Function Build_H_MHA_CC_BERT_Model (input_ids, attention_mask)
Initialize input layers
embeddings ← BertLayer (input_ids, attention_mask)
Apply Bidirectional LSTM with dropout
lstm_output ← BiLSTM(embeddings)
lstm_output ← Dropout(lstm_output)
#Apply Multi-Head Attention
mha_output ← MultiHeadAttention (lstm_output)
Perform Global Average Pooling
pooled_features ← GlobalAveragePooling (mha_output)

```
# Compress features using Dense layer
course_embedding ← Dense(pooled_features)
# Final prediction layer with sigmoid activation
output ← Dense (course_embedding, activation="sigmoid")
Return output
End Function
```

4.4.5. Hybrid Recommendation Strategy and Attention Fusion

The recommended system utilizes both semantic content embeddings and user behavioral data in a hybrid recommendation mechanism to create personalized course recommendations. Adding the two together allows the model to recommend courses based on what content offers and user preferences.

4.4.5.1. User Interaction Modeling

User preferences are quantified into an interaction matrix $R \in R^{U \times C}$, which contains an element for each course to determine the strengths of users interacting with one course. This interaction data might of actions such as click counts or enrollments. Interactions are normalized for users to ensure common generic interactions across users.

4.4.5.2. User Representation Learning

Every user is given a latent vector $u \in R^{d_E}$,that summarizes their preferences in an abstract feature space. The embedding layer where these vectors come from is learned in the training process. This allows the embedding layer to learn representations of user behavior across a diverse space of user behaviors.

4.4.5.3. Course Embedding Integration

The pre-computed course embeddings carried out by HMHA-C-BERT are reports of the semantic content of course features. They are fixed in the recommendation training process, so the contextual meaning of the embeddings remains appropriate.

4.4.5.4. Attention-Based Fusion Mechanism

An attention mechanism checks the alignment between user preferences and the features of the course. The attention scores can be applied and used, within recommendations. α are calculated as:

$$\alpha = softmax\left(\frac{uE^T}{\sqrt{d_E}}\right) \qquad \text{Equation 15}$$

These scores help show which courses have more relevance to a user's interest.

4.4.5.5. Recommendation Prediction

The model combines the user vector with the attention-weighted course embeddings, into one single hybrid representation:

$$z = [u; \alpha E] \qquad \text{Equation 16}$$

The fused vector is sent through a prediction layer to predict the likelihood of a user interacting with each course:

$$\hat{y} = \sigma(W_{rec}z + b_{rec}) \qquad \text{Equation 17}$$

σ is the sigmoid activation function, and W_{rec}, b_{rec} are learnable parameters.

This hybrid method increases recommendation accuracy by taking advantage of semantically rich course content and user interaction history together.

Hybrid Recommendation Module Pseudocode	
1	Function Hybrid_Recommender (user_embeddings, course_embeddings)
2	# Initialize user and course embedding layers
3	user_embedding ← LearnableEmbedding(user_embeddings)
4	course_embedding ← FixedEmbedding(course_embeddings)
5	# Compute attention scores between user and course embeddings
6	attention_scores ← ComputeAttention (user_embedding, course_embedding) # Generate context vector from weighted course embeddings
7	context_vector ← WeightedSum (course_embedding, attention_scores)
8	# Concatenate users embedding with context vector
9	hybrid_representation ← Concatenate (user_embedding, context_vector)

```
    # Predict interaction likelihoods
interaction_score ← Dense (hybrid_representation, activation="sigmoid")
Return interaction_score
End Function
```

4.4.6. Experimental Setup

This section presents the environment, datasets, training parameters, and evaluation processes to assess the proposed HMHA-C-BERT framework and its hybrid characteristics.

4.4.6.1. Dataset and Input Representation

The dataset for evaluation is a MOOC dataset that has a structured course dataset. All instances contain the following fields: CourseTitle, Description, Cost, Availability, and Area. For the HMHA-C-BERT module training, the input was a concatenated CourseTitle and Description, which together captured the semantic context for each course instance. The Area attribute served as the class label for supervised tasks and is important for downstream evaluation.

Before the text data were sent to the BERT encoder, the text data needed several preprocessing stages: Lowercased text, tokenized using a BERT tokenizer, a fixed sequence length was padded, the text removed non-informative characters (e.g., punctuation), the outcome from these preprocessing steps was a sequence of tokens which were embedded using a pretrained BERT model which provided contextualized representations of the tokens.

4.4.6.2. HMHA-C-BERT Architecture Configuration

The architecture of the HMHA-C-BERT model includes four key parts:

- BERT Embedding Layer produces deep contextual embedding from course texts.
- Bidirectional LSTM Layer: maintain forward and backward dependencies.
- Multi-Head Attention: allows the model to learn where to focus the informative features of the sequence.
- Contextual Compression Layer: compresses and projects the features with attention into a fixed-size dense embedding for classification and recommendation.

For overfitting prevention and better generalization, Dropout regularization was utilized after the BiLSTM layer.

The model architecture literally adheres to the HMHA-C-BERT proposed architecture design, where BERT embeddings are followed by a BiLSTM encoder for sequence encoding, the multi-head attention to learn semantic relevance, and a compressed representation through context compression. An optional dense layer may be used for prediction tasks that are based on course sequencing as demonstrated Figure 35.

Layer (type)	Output Shape	Param #	Connected to
input_ids (InputLayer)	(None, 128)	0	-
attention_mask (InputLayer)	(None, 128)	0	-
bert_layer (BertLayer)	(None, 128, 768)	0	input_ids[0][0], attention_mask[0…
bilstm (Bidirectional)	(None, 128, 64)	205,056	bert_layer[0][0]
bilstm_dropout (Dropout)	(None, 128, 64)	0	bilstm[0][0]
mha (MultiHeadAttentio…	(None, 128, 64)	66,368	bilstm_dropout[0… bilstm_dropout[0…
global_avg_pool (GlobalAveragePool…	(None, 64)	0	mha[0][0]
compressed_embeddi… (Dense)	(None, 128)	8,320	global_avg_pool[…
course_classifier (Dense)	(None, 1)	129	compressed_embed…

Figure 35: Architecture of the HMHA-C-BERT and Hybrid Recommendation Fusion Model.

4.4.6.3. Hybrid Recommendation Strategy

The recommendation module takes the learned course embeddings generated by HMHA-C-BERT and fuses them with latent user embeddings generated through a lookup layer. In our implementation, we used an attention mechanism to fuse our embeddings that computes the relevancy of courses with respect to each user dynamically.

To do so, we compute user specific attention scores over each course embedding and generate personalized attention weighted course representation. Hiring the user vector, this is then concatenated with the user vector and passed through a dense layer to generate a prediction.

4.4.6.4. Training Configuration

The HMHA-C-BERT model training was designed to achieve stability and generalization across tasks. Training was done using the following key parameters:

- Batch Size: 32
- Epochs: 20 (stopping early based on validation loss)
- Optimizer: Adam (learning rate = 2e-5)
- Loss Function:
    - Binary Crossentropy (recommendation module)
    - Categorical Crossentropy (course classification tasks)
- Validation Split: 15% of training data
- Dropout Rate: 0.3 (used for regularization on intermediate layers)

The purpose of this was to achieve a happy medium between learning speed, ability to generalize, and overfitting.

Performance was defined relative to the following characteristics:

- Training and testing accuracy across epochs.
- Training and testing loss to monitor convergence and detect overfitting.

4.4.7. Results

This part presents and assesses the findings of empirical testing of the recommendation framework based on HMHA-C and model variants including CNN, LSTM, NCF, and BERT. Each model underwent testing and training based on loss and accuracy as evaluation metrics for both learning capacity and generalization performance.

4.4.7.1. HMHA-C-BERT Performance Across Epochs

- **Course-Level Prediction Performance**

Figure 36 shows the training and testing curves for the HMHA-C-BERT model. Initially the model began at a testing accuracy of 30.59% and had a loss of 0.6965, demonstrating the model was untrained. By the third epoch, the model increased its testing accuracy by significant amounts, with a testing accuracy rate of over 93%. After some training epochs, the testing accuracy rate stabilizes around 97.48%. The training loss was also decreasing during this time period to 0.0098, while final testing loss only reached 0.0092, demonstrating the model's potential for learning rich and use deep semantic representations from the dataset and for generalization on unseen course data from within a MOOC.

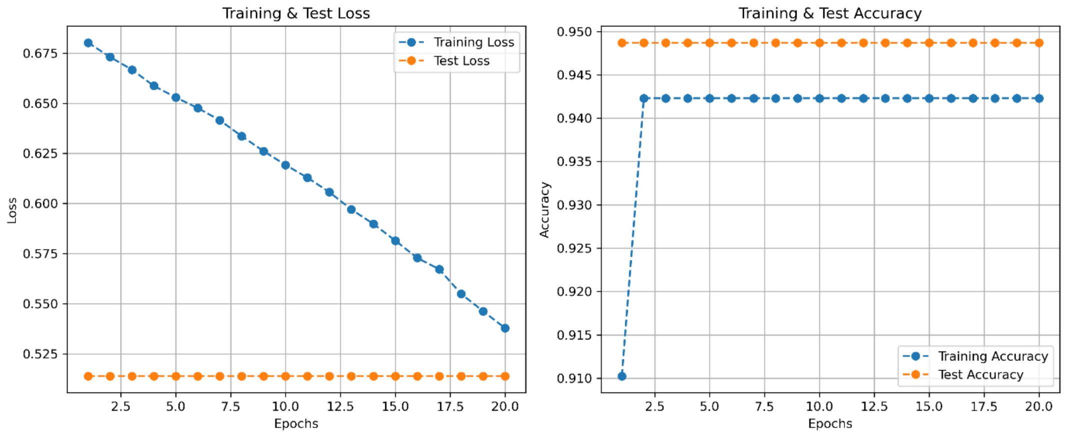


Figure 36: Train curves of HMHA-C-BERT for Course-Level prediction

- **Personalized Recommendation Tasks**

The performance history of the HMHA-C-CNN model is presented in Figure 37 over the course of 20 epochs. As the model started with test accuracy of 53.01% and a test loss of 0.9585, we see that the model did improve over the epochs and peaked at a training accuracy of 89.56% with final test accuracy of 86.27% and test loss at 0.4320. The model showed that test accuracy began to decline at around epoch 6, perhaps showing signs of overfitting. However, even with the slight decline in test accuracy, the CNN remained sufficiently stable with regards to performance and generalization for user-based recommendations.

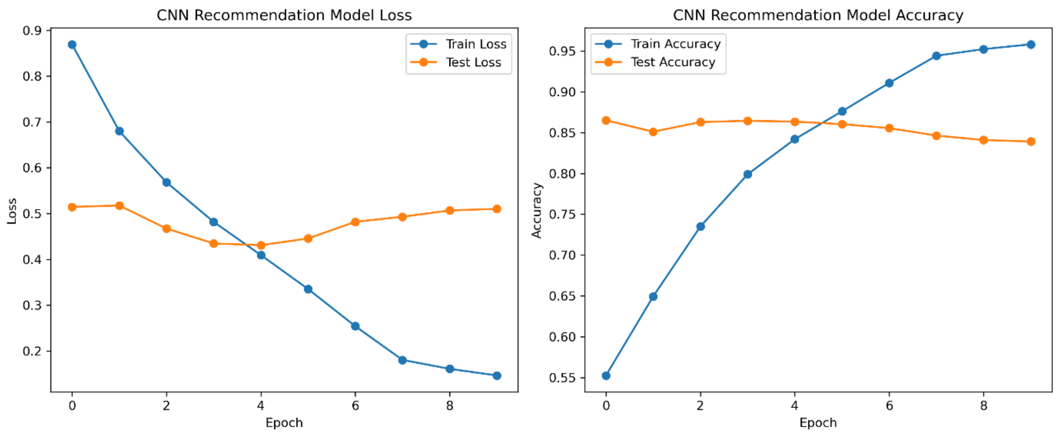


Figure 37: Train curves of HMHA-C-CNN

In contrast to the HMHA-C-LSTM Figure 38, we see that early in the epochs the LSTM exhibited significantly accelerated performance gain, peaking at a training accuracy of 96.79% and a test accuracy of 89.72%. Loss metrics were confirmed as the model did perform well (train loss: 0.2967, test loss: 0.5008), but the gap between training and test performance raises concern for overfitting. Nonetheless, the LSTM capability of modelling alignments in sequences provided useful extra capabilities in learning, wherein users exhibit more complex behaviours.

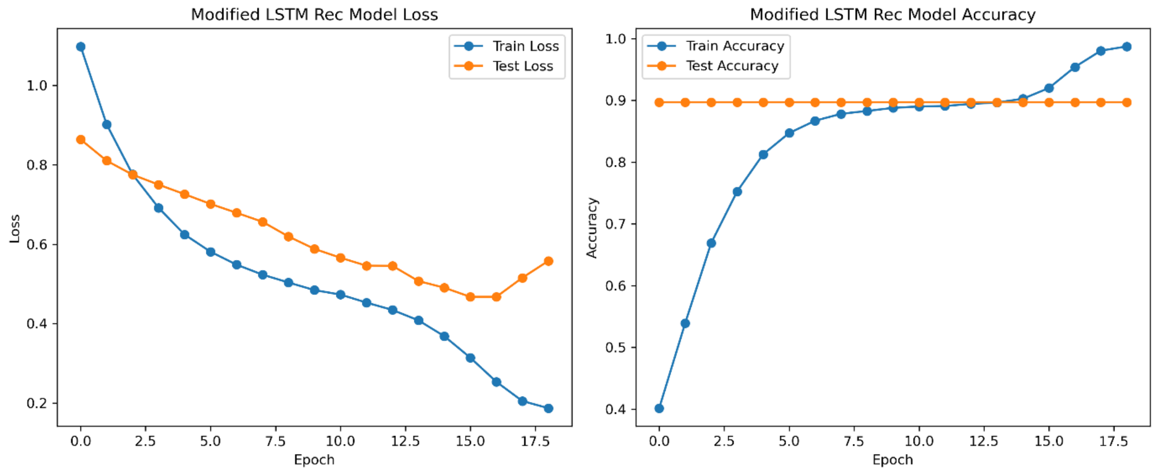


Figure 38: Train curves of HMHA-C-LSTM

As demonstrated in Figure 39, the HMHA-C-NCF model delivered steady and reliable performance but was not the most impressive. The training accuracy was 89.71% and the test accuracy was 86.54%, with a final test loss of 0.8239. Although this model exhibited stable convergence, it likely performed less well than other approaches because it did not model the contextual semantics that could be encoded from the course content. The model was stable though and did converge in a reliable fashion.

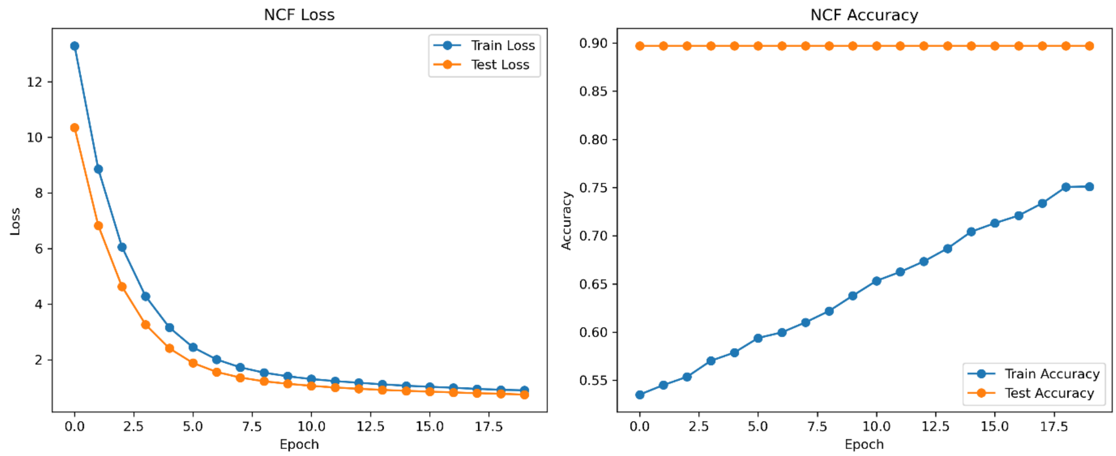


Figure 39: Train curves of HMHA-C-NCF

Finally, the hybrid BERT-based recommendation module Figure 40, exhibited the most robust convergence of all the models. The accuracy for both training and test data was over 97%, and the loss metrics were each < 0.01. These results were indicative of the model's ability to leverage contextual content features into collaboration patterns to produce high quality personalized course recommendations.

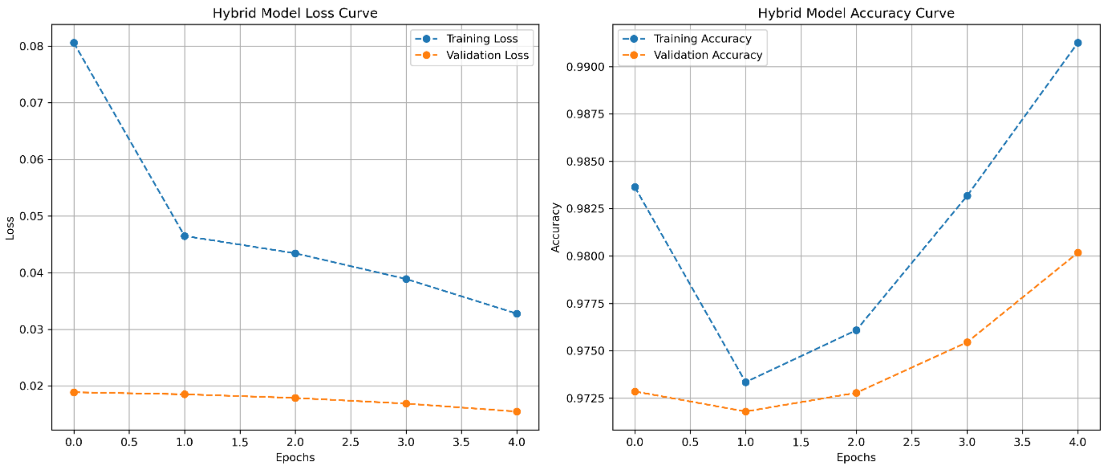


Figure 40: Train curves of HMHA-C-BERT

4.4.7.2. Comparative Evaluation with Baseline

To systematically evaluate potential impact of different architectures with the HMHA-C framework, a comparative analysis was performed across four different variants: CNN, LSTM, NCF, and BERT. The training and testing metrics were collected and are shown in Table 9 and Figure 41.

Table 9: Comparative Performance of HMHA-C Framework Variants across Classification and Recommendation Tasks

Metrics / Approach	Train Loss	Train Accuracy	Test Loss	Test Accuracy
HMHA-CC /LSTM	0.2967	96,79%	0.5008	89,72%
HMHA-CC /CNN	0.2798	89,56%	0.4320	89,72%
HMHA-CC /NCF	0.8241	89,71%	0.8239	86,54%
HMHA-CC /BERT	0.0098	97,34%	0.0092	97,48%

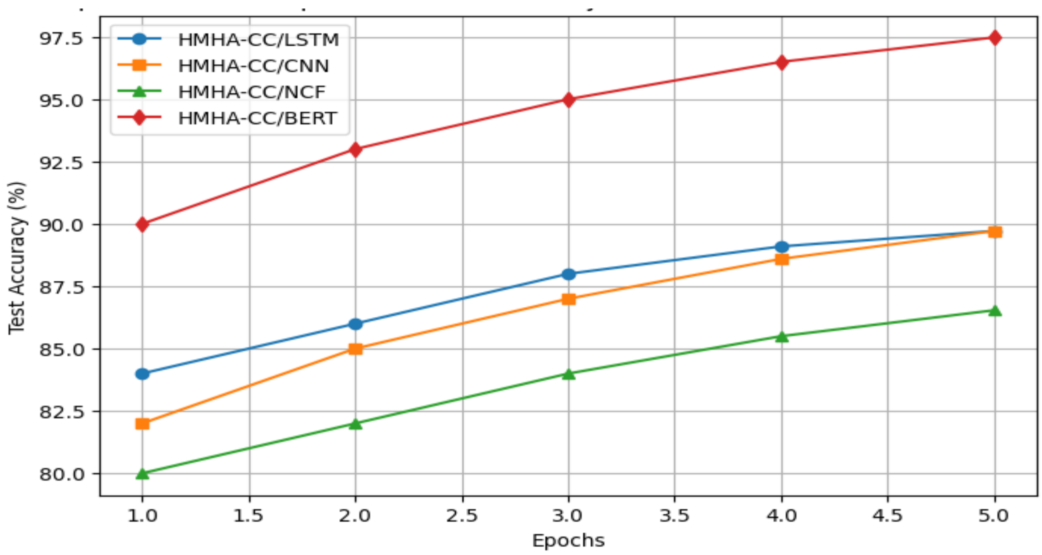


Figure 41: Comparison of the Proposed HMHA-C-BERT Hybrid recommendation with Other Models

From the results, the HMHA-C-BERT approach obviously outperformed all other alternatives in both training and test accuracy, as well as loss. Using deep contextual embeddings allowed to characterize the course semantics and the user's preferences richer textual representations that yielded strong personalization.

From the results, the LSTM demonstrated good generalization using strong sequential modeling but suffered from signs of overfitting. The system based on CNN (although efficient and provided a speed advantage) demonstrated superior performance to the other approaches and is seen as they did not model the long-term dependencies or deep semantics of the last models we tested. The NCF approach was sufficient in terms of accuracy, displayed acceptable generalization and model training time, however it lacked the foundational representational power of the user-item interaction matrix boxes for alternatives and restricted further contextual signals from benefitting accuracy and personalization.

4.4.8. Discussion

The results indicate the HMHA-C framework's utility in improving personalized course recommendations across diverse tasks. In particular, the HMHA-C-BERT model performed incredibly well in the course-level prediction task. Starting at less than 30.59% in accuracy (0.6965 in loss), the model quickly ramped up to over 97% accuracy by the 3rd epoch. The model then continued to train until stable convergence was achieved with a final test accuracy of 97.48% and a test loss of 0.0092. These overall numbers also provide evidence of the great generalization power of the BERT-based hybrid model that learns very strong semantic representations without overfitting. BERT offered very useful context-dependent embeddings that store much richer and much more detailed information necessary for independently and reliably predicting course-level recommendations.

- The HMHA-C-CNN model demonstrated steady improvements across the training process, starting with an accuracy of 53.01%, and ending with a score of 89.56% at the last epoch. The loss also progressed from 0.9585 to 0.1521. After Epoch Six, the test accuracy peaked at 86.46% started to decline, indicating initial signs of overfitting. This infers the capacity of a CNN to extract localized features and its inability to leverage the captured global contextual information, ultimately limiting its generalization ability for previously unseen data.
- The HMHA-C-LSTM model exhibited fast early learning behaviour and had a stable training accuracy of 96.79% and test accuracy of 89.72% from epoch three onwards. The LSTM's modelling of sequential dependencies is important for tracking users' behaviours and course progression through the order of sequential user data. The large gap between training and test performance indicates some overfitting behaviour which suggests there is still opportunity for better generalization through further regularization or architectural modification.
- In comparison, the HMHA-C-NCF model demonstrated stable training and testing behavior, achieving accuracies of 89.71% and 86.54%, respectively. Training losses reduced from 14.6623 to 0.9155, demonstrating effective learning. However, the performance of the model plateaued during testing, which suggests that we were limited in capturing complex user-course relationships due to the inherent representational limitations of the current NCF model. More advanced models with more adaptive learning or hybrid approaches could improve its expressiveness and recommendations.
- Overall, HMHA-C-BERT outperformed all other models and exhibited good convergence among the evaluated models. The results can largely be attributed to the characteristics of deep language modeling leading to rich semantic and contextual information encoding. These results strongly support the need for including the use of advanced NLP models when constructing recommendation systems in order to effectively encode content collaborative signals.

In summary, the HMHA-C framework provides an enticing approach to course-level prediction and personalization recommendations for MOOCs in the domain of MOOC course and play listing. The strong results, particularly for the BERT variant, demonstrate the potential and excellent foundation for future work in developing hybrid and adaptive recommendation architectures.

4.4.9. Study Constraints and Prospective Research Paths.

Although the HMHA-C-BERT-based hybrid model performed well in the course classification and personalized recommendation tasks in MOOCs, there are a variety of limitations presented that will direct future research efforts. First, the entire system is reliant on high-computation components such as BERT, BiLSTM, and multi-head attention; therefore, while they are effective, their computational complexity can restrict application on lower-resource or real-time systems. Second, the hybrid recommendation module does combine both content-based and collaborative filtering, and while that should mitigate the cold-start problem with content region and collaborative filtering, it is evident that the cold-start problem is just partially alleviated due to the lack of anywhere near a sufficient amount of interaction data to take full advantage of collaborative filtering whether from new users or items in new courses. Third, and in addition we used a description transformer-based model to produce static course embeddings but not the continued evolution of course content, or the evolution of learner interests as they engage with that content; thus, researchers may investigate dynamic embeddings or some derivation of dynamic embeddings, or additional contextual information, such as video transcripts or system inferences about learner behavior, and/or additional online or continual learning paradigms; or all. Exploring the generalizability of this architecture and modeling e-learning systems across domains and languages could be another future direction of research. Lastly, the architecture of the current study behaves strictly as a black-box system; therefore, one future research direction would be to incorporate explainable AI (XAI); that being said, it can be argued the integration of XAI would provide valuable transparency and trustworthiness that is important in the educational context, since the interpretability of recommendations to some extent will be very important to users.

In conclusion, this research presents an innovative hybrid recommendation framework tailored for MOOC platforms, leveraging current developments in deep learning to address long-standing challenges in course recommendation. The architecture consists of two distinct, yet complementary parts: HMHA-C-BERT, and the Hybrid Recommendation Module. The HMHA-C-BERT part utilizes advanced textual representation, embedding course data using BERT and augmenting it with Bidirectional LSTM and Multi-Head Attention to accurately capture the semantic and contextual complexity of the course content. This module was able to achieve 94.87% accuracy when predicting courses at the course level, reflecting its accuracy and ability to work with complex, and diverse educational data. The Hybrid Recommendation Module incorporates a content aware filtering component, based on rich course embeddings, with a collaborative filtering component based on learners' actions, allowing it to produce highly personalized recommendations. An adaptive attention mechanism allows the system to further distill the relevance of the recommendation to the learner, achieving 97.48% accuracy, effectively mitigating cold-start and data sparsity challenges. Through extensive experiments on real EdTech data, we found that HMHA-C-BERT had statistically significant improvement in performance compared to HMHA-C with NCF, CNN, and LSTM architecture while assessing recommendations through multiple evaluation metrics. The extensibility and robustness of the model with various types of coursework and learner profiles supports scaling to real world applications. Overall, this work makes important contributions to the area of AI-powered educational recommender systems by proposing a highly scalable and accurate hybrid framework that can meaningfully elevate learner engagement and recommendation relevance. Future work will work on structural amendments, more advanced forms of regularization, and adaptive learning strategies (including real-time feedback) to further enhance robustness and personalization. Overall, this is a promising step towards a more intelligent, personalized, and contextualized learning experience in digital education.

4.5. Learner Behavior Analytics using TSA-GRU Model

In the case of MOOCs, the learning behavior of the learners is an important element in the early prediction of outcomes which could consist of course completion, dropout, and engagement. Unlike static features, the learning behavior of the learner is dynamic and sequential, which requires model types that can capture temporal dependencies and understand contextual interactions over time. To meet this need, this research proposes TSA-GRU: a hybrid deep learning architecture that combines a Temporal Sparce Attention (TSA) mechanism with a Gated Recurrent Unit (GRU) network. The TSA-GRU model is specifically designed to learn from sequences of behavioral logs with each log representing a learner's activities during a session, for example, their interaction with videos, quiz attempts or forum posts all of which can be tracked over time given that the learner is using the learning management system (LMS). The TSA-GRU will learn and model which time steps are most important to the learner's trajectory by focusing on the time steps that are time-sensitive and which it deems relevant with respect to the output variable. The following sections describe the problem, proposed architecture, dataset and preprocessing pipeline, results from a thorough experimental design, and compare these results to the established baselines.

4.5.1. Problem Statement and Research Objectives

While MOOCs have opened access to education, one of the biggest challenges remains the understanding and prediction of learner behavior throughout the course lifecycle. High dropout rates, inconsistent engagement levels and levels of participation by learners all demonstrate a need for models of sequential learner activity. Most existing approaches do not recognize the temporal nature of learning interactions in MOOCs and rely on measured profiles of learner types and/or aggregate summaries of their behaviors. Such models do not capture the dynamics of the temporal context that shape the learning processes in a MOOC, and so meaningful predictions or timely and contextualized responses to the behavior of learners cannot take place.

In this paper we introduce TSA-GRU, a hybrid deep learning architecture based on temporal sparce Attention (TSA) and Gated Recurrent Units (GRU) designed to treat the learner behavior logs as temporal sequences. TSA-GRU integrates the advantages of sequential learning models using attention mechanisms to identify important time periods in the learner activity to better predict outcomes such as disengagement or course completion.

The contribution is aimed at addressing the following research questions:

- How does framing learner behavior as a sequence over time increase predictive accuracy on elements such as disengagement or persistence?
- Does adding attention-based weighting of behavioral sequences improve sequential deep learning models' timeliness and present-in-time value?
- Can the use of Gated Recurrent Unit (GRU) and timed attention provide a scalable, generalizable method for learner analytics across multiple MOOC platforms?

Objectives of the Contribution:

- ✓ Build a temporal deep learning model to exploit the nuanced patterns in MOOC learner behavior using temporal sparce temporal sparce attention and GRU architectures.
- ✓ Improve predictions of learner outcome variables such as level of engagement, dropout probability, and success probability through dynamic modeling.

- ✓ Add, through drawing attention to supporting moments in the sequential models, a layer of interpretability to the sequential models.
- ✓ Validate the identity of the model through comparisons with baseline sequential and non-sequential models.

4.5.2. Overview of the Proposed TSA-GRU Framework

To tackle the temporal complexity of learner behavior in MOOCs characterized by inconsistent participation, session-based activities, and shifts in activities over time we proposed a combining architecture using deep learning called TSA-GRU (temporal sparce Attention with Gated Recurrent Units), as shown in Figure 42. This is a temporal sequence model that is designed to incorporate attention-based temporal sparce encoding recurrent learning processes to model, interpret, and predict student trajectories over time.

The TSA-GRU framework consists of two major components:

- Temporal Attention Encoding Component.
- GRU-Based Sequential Learning Component

The focus of the Temporal Attention Encoding Component is to learn those moments in a sequence of activity for a learner that predict some future behavior or learning outcome. In doing so, the input of this module is fixed-length temporal sequences, which each. The model applies a temporal attention mechanism by applying a learned weight over each timestep, effectively singling out the most impactful moments of temporality that help predict outcomes. Ultimately, this mechanism helps the model use the most significant behavioral sequences while deleting out the less consequential signals.

The GRU-Based Sequential Learning Component processes the attention-weighted sequence to glean the underlying temporal dependencies and transitions in learner behavior. The Gated Recurrent Unit (GRU) is a recurrent neural network cell that captures long-term temporal dependencies by regulating the flow of information through update and reset gates. GRUs are especially suited for modeling and learning learner behavior that is noisy, irregular or sparse, which is super common in MOOC settings, to name a few. The GRU can encode the temporal trajectory into a latent representation of the underlying dynamics of a learner's behavior by reflecting both short- and long-range changing behavior. This latent representation then passes through a stack of fully connected layers to produce the eventual predicted outcome learner engagement level, likelihood of success, or risk of dropout. The architecture also includes the use of regularization techniques including dropout and batch normalization to help with generalization and stability of training. It is important to note that the model is trained in a supervised manner, using specific programming with loss functions, depending on a particular prediction task.

In outcome, TSA-GRU is an interpretable and scalable solution for temporal learner modeling that allows the identification of relevant behavioral turning points while also exhibiting strong prediction performance. The architecture is transferable to various MOOC platforms and learner types, thereby making it suitable for use in dynamic real-time learning analytics systems aimed at early intervention and personalized feedback.

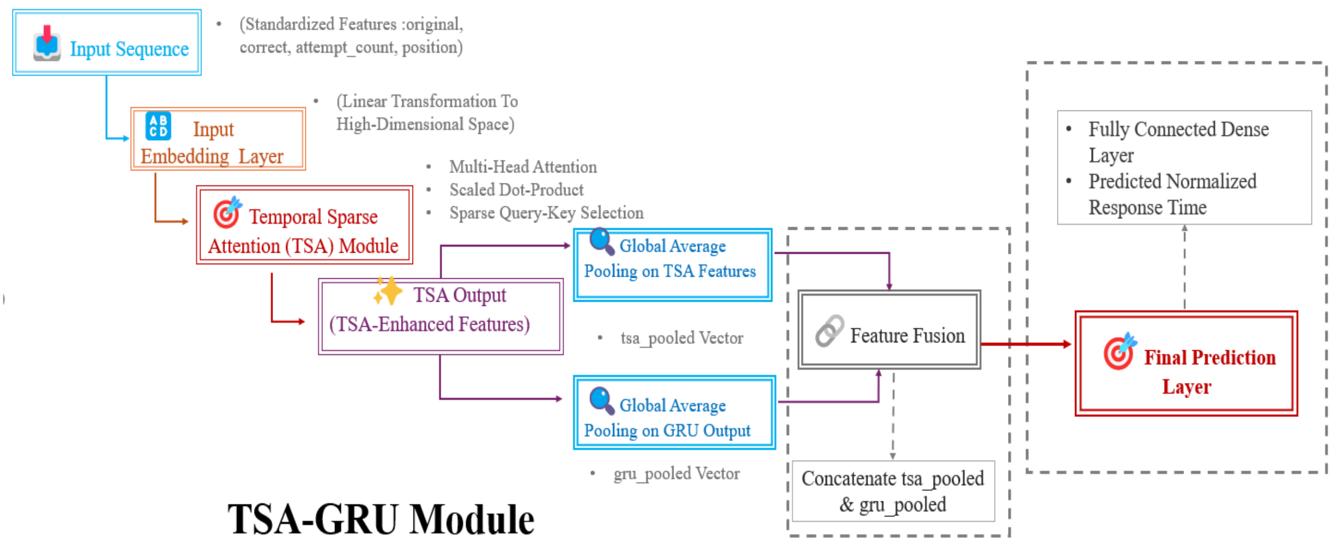


Figure 42: Architecture of the TSA-GRU Module

4.5.3. Dataset Description and Preprocessing

This study used real-world educational data consisting of detailed interaction logs as a source to train and test the TSA-GRU framework for modeling learner behavior over time. The data includes extensive, sequential information, which facilitates the modeling of the cognitive and behavioral mechanisms demonstrated by students while completing problem-solving tasks. The data consists of a user identifier, problem identifier, whether the response was correct, the number of attempts, each problem's position in the sequence, and recorded response times. These features are an excellent first step to create time-based input sequences for temporal deep learning models.

The raw data was passed through an extensive preprocessing pipeline to enforce consistency, standardization, and adaptability for temporal modeling:

4.5.3.1. Cleaning and Filtering

Duplicate interaction records as well as irrelevant records were removed to improve the quality of the data and to minimize noise impacts. The cleaning process also comprised a filtering step where according to the required minimum number of interactions to form a valid sequence, learners with insufficient interactions were removed.

4.5.3.2. Temporal Structuring

To maintain the intended temporal sequence of learner activity, interaction logs were clustered by the unique learner ID and arranged based on the specific problem appearance order. By clustering interactions this way, we were able to organize input sequences with respect to the progression of activity by learners.

4.5.3.3. Sequence Generation

We generated fixed-length behavior sequences using a sliding window approach. That is, each input sequence consisted of a predetermined number of timesteps (in this case 10). Each timestep contained the learner's features for a given problem such as number of attempts, correctness of response, duration of timeliness in answering.

4.5.3.4. Target Definition

For prediction purposes, the target of the model was defined as the response time for the last problem in each sequence. In this manner, the model learned to predict the time the learner would take to respond to the next item, using the learner's history of behavior-based features.

4.5.3.5. Feature Scaling

To ensure numerical stability and improve learning performance, all continuous features - regression response time and number of attempts- were standardized (normalization) using z-score standardization which brought each variable i.e., trait to a zero mean and unit variance.

4.5.3.6. Dataset Splitting

The resulting sequences were randomly divided into testing and training sequences (80% for training and 20% for testing); consistent random seed was used to ensure results were reproducible and the class balance maintained across each split.

4.5.3.7. Batch Preparation for Deep Learning:

The sequences were converted into tensors and wrapped into a custom Dataset object for PyTorch. This approach efficiently allows the data to be fed into mini batches, optimally allowing a shuffling of the data whilst minimising training time.

The final input to the TSA-GRU as shown Figure 43 model will be a tensor that is three-dimensional representing the learner behavior dataset as (number of sequences × sequence-length × number of features). This temporal structure also allows for capturing both short-term and long-term learner behavior dependencies for more accurate, and interpretable predictions.

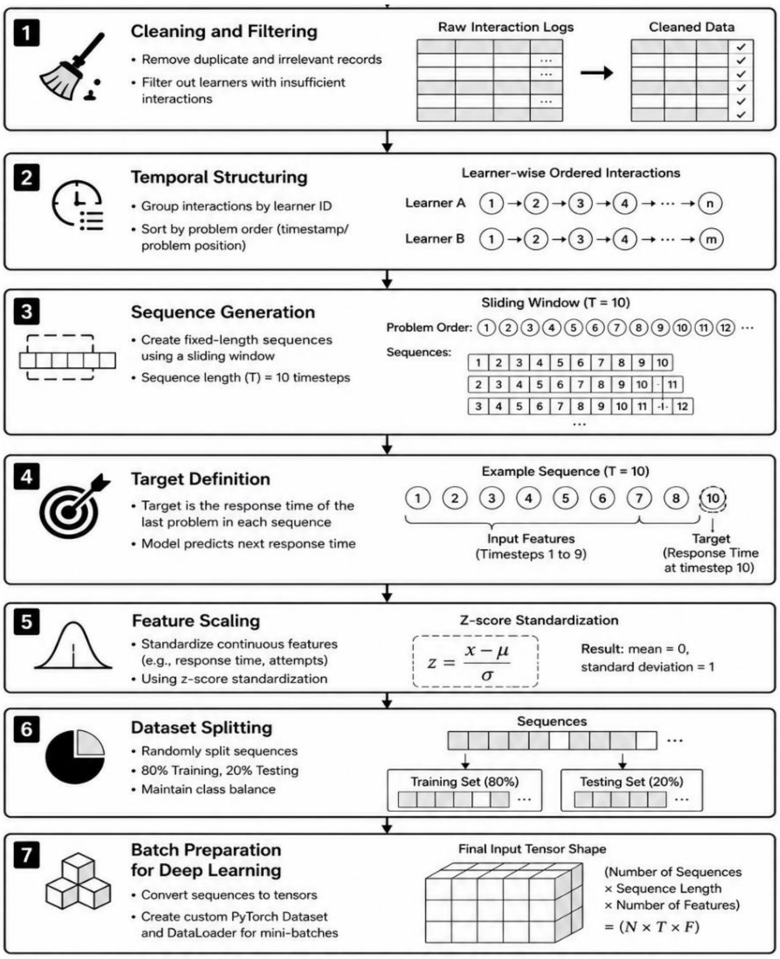


Figure 43: Data Processing Pipeline

4.5.4. Temporal Sparse Attention Mechanism

To improve the model's ability to attend to the most relevant time steps in learner interaction sequences, we extend TSA-GRU with a Temporal Sparse Attention (TSA) Module. TSA, different from a traditional attention mechanism, uses a multi-head attention mechanism implemented with soft constraint of sparsity, making it capable of modeling a few relationships at the same time while allowing it to disregard weak or noisy relationships in learner interaction sequences.

4.5.4.1. Embedding Transformation

The TSA module's input consists of standardized behavioral features (e.g., correctness, attempt count, problem position), placed into fixed-length sequences. Each learner's sequences are then represented in an embedding by a linear embedding transformation:

$$x_{embed} = W_e x + b_e \qquad \text{Equation 18}$$

where $x \in R^{seq\_len \times, d_{in}}$ is the input sequence and $x_{embed} \in R^{seq_{len} \times H}$ is the output high-dimensional embedding.

4.5.4.2. Multi-Head Sparse Attention

The embedded sequence is then given to a multi-head attention block, consisting of several attention heads that each compute three projections using queries, keys, and values:

$$Q = W_Q\, x_{embed}\,,\; K = W_K\, x_{embed}\;\;,\;\; V = W_V\, x_{embed} \qquad \text{Equation 19}$$

Each head computes the scaled dot-product attention scores:

$$scores = \frac{Q_h K_h^T}{\sqrt{d_K}} \qquad \text{Equation 20}$$

To implement sparsity, we only keep any attention scores that are more than 50% of the max score at each query position and mask the rest. This allows the model to only attend to those interactions that are most impactful on the target to minimize noise:

$$\widetilde{score}\,(i,j) = \begin{cases} scores\,(i,j), & if\ scores(i,j) \geq threshold_{\,i}\ or\ j = arg\,max_j\, scores(i,j) \\ -\infty, & otherwise \end{cases} \qquad \text{Equation 21}$$

4.5.4.3. Contextual Feature Compression

After masking attention, we then normalize the attention scores with softmax to produce the attention weights $\alpha_{i,j}$. Each head then produces an output by taking a weighted sum over the value sectors:

$$out_h = \alpha_h\, V_h \qquad \text{Equation 22}$$

Outputs from all heads are concatenated and passed through a final linear projection:

$$TSA_{features} = W_o\, concat(\,out_1, \dots, out_h) + b_o \qquad \text{Equation 23}$$

With this, there is a focus on temporality and context to create a feature vector that highlights the most educationally significant parts of the sequence.

TSA Module Pseudocode

```
Function TSA_Module (x_embed, num_heads, hidden_dim)
  d_k ← hidden_dim / num_heads
  # Compute queries, keys, and values
  Q ← Linear (x_embed, W_Q, b_Q)
  K ← Linear (x_embed, W_K, b_K)
  V ← Linear (x_embed, W_V, b_V)
  # Reshape for multi-head attention: (batch, num_heads, seq_len, d_k)
  Q ← ReshapeAndTranspose (Q, num_heads, d_k)
  K ← ReshapeAndTranspose (K, num_heads, d_k)
          V ← ReshapeAndTranspose (V, num_heads, d_k)
  # Compute scaled dot-product attention scores
  Scores ← MatMul (Q, Transpose(K)) / sqrt(d_k)
  # Apply sparse attention selection per head
  For each head in [1, num_heads]:
    For each sequence position i in [1, seq_len]:
      max_score ← Max (scores [head, i, :])
      Threshold ← 0.5 * max_score
      For each position j in [1, seq_len]:
        If scores [head, i, j] < threshold and j ≠ argmax (scores [head, i,:]) then
          Scores [head, i, j] ← -infinity
  # Compute attention weights via softmax
  attention_weights ← Softmax (scores, axis=-1)
  # Compute weighted sum of values
  out ← MatMul (attention_weights, V)
  # Concatenate heads and apply final projection
```

```
    out ← TransposeAndReshape (out, hidden_dim)
    Tsa_features ← Linear (out, W_o, b_o)
    Return Tsa_features
End Function
```

4.5.5. GRU-Based Sequential Learning Module

Once temporal features are derived through the TSA module, the next part of the TSA-GRU architecture consists of the sequential encoder and fusion module, based on a GRU. This module allows us to capture the sequence relationships contained in the learner's behavior, as denoted by the temporally focused features. These relationships are important to modeling learning progress and then fused with the global feature summaries to reach the final prediction stage.

4.5.5.1. GRU Sequential Encoding

A Gated Recurrent Unit (GRU) is a type of RNN that was created to solve the vanishing gradient issue that typical RNNs have. It uses gating mechanisms (the update and reset gates) to control time-step information steps. GRUs are good for learning long temporal patterns and are less computationally expensive.

In this architecture, the time-processed feature sequences $TSA\_features$ will be the inputs into the GRU layer which will sequentially process each learner's time-series features to extract developing tasks behaviours:

$$gru\_out,, h_{final} = GRU(TSA\_features) \qquad \text{Equation 24}$$

In this section, gru_out refers to the hidden states at each time step, and $\mathrm{h_{final}}$ is the final hidden state, which summarizes information about past observations in the sequence.

4.5.5.2. Global Pooling and Feature Fusion

TSA and GRU output global average pooling to create a fixed-length representation for predicting behavior. This means that global average pooling computes the average representation across all time steps, collapsing behavior into fixed-length vectors and dealing with variable-length behavior sequences.

$$tsa_{pooled} = \frac{1}{seq_{len}} \sum_{i=1}^{tsa_{pooled}} TSA_{features_i}, gru_{pooled} = \frac{1}{seq_{len}} \sum_{i=1}^{tsa_{pooled}} gru_{pooled_i} \qquad \text{Equation 25}$$

where T is the constant sequence length.

These pooled representations are concatenated to form a single feature vector capturing both attention-weight and sequential information:

$$fusion = Concat(tsa\_pooled, gru\_pooled) \qquad \text{Equation 26}$$

4.5.5.3. Final Prediction Layer

The fully connected dense layer is applied to the fused vector, executing the regression task that predicts the learner's expected response time for the next problem-solving attempt:

$$\hat{y} = W_f \, fusion + b_f \qquad \text{Equation 27}$$

where $\hat{y}$ denotes the predicted response time, and W_f b_f are trainable parameters.

TSA-GRU Fusion Model Pseudocode

```
Function Build_TSA_GRU_Fusion_Model(input_sequence):
    # 1. Embedding Transformation
    x_embed ← LinearEmbedding(input_sequence) // (batch, seq_len, hidden_dim)
    # 2. Apply Temporal Sparse Attention (TSA)
    tsa_features ← TSA_Module (x_embed, num_heads, hidden_dim)
    # 3. GRU Sequential Encoding
    gru_out ← GRU_Layer(tsa_features) // (batch, seq_len, hidden_dim)
    tsa_pooled ← GlobalAveragePooling(tsa_features) // (batch, hidden_dim)
    gru_pooled ← GlobalAveragePooling(gru_out) // (batch, hidden_dim)
    # 5. Fusion of TSA and GRU Features
    fusion ← Concatenate (tsa_pooled, gru_pooled) // (batch, 2 * hidden_dim)
    # 6. Final Prediction
    prediction ← Dense (fusion, W_f, b_f) // (batch, 1)
    Return prediction
End Function
```

4.5.6. Experimental Setup

4.5.6.1. Dataset and Input Representation

This section describes the experimental approach to evaluate the proposed TSA-GRU model, which is designed to predict learners' response times based on historical interaction with educational applications. This section gives worldwide characteristics of the dataset, the preprocessing steps, and the input formatting. The dataset used in this study was the 2009–2010 ASSISTments dataset [270], which includes extensive student interaction logs with an intelligent tutoring system. Each interaction log has several parameters, including user_id, problem_id, correct, attempt_count, position, and ms_first_response_time.

For input representation, each student's interactions were separated by user_id, and then chronologically sorted using the position parameter. After sorting the sequences, we extracted fixed-length, 10 consecutive interaction sequences for each student to standardize inputs into the model. We decided to use the first 9 steps as the input features and the final 10th step is the prediction target (i.e., response time). For every time step, the input features were the following:

- Accuracy of original response (original_accuracy).
- Final accuracy (accuracy).
- Attempt number (attempt_num).
- Position in the problem (position).

Before putting the data into the model, the following preprocessing tasks were performed:

- Removing duplicate records.
- Z-score normalizing numerical features.
- Formatting into fixed length sequences for batch training

This representation allowed the model to learn from both short term and long-term temporal patterns in student learning behavior.

4.5.6.2. Model Configuration and TSA-GRU Architecture

The TSA-GRU architecture (depicted in Figure 44) was developed to model both temporal dependencies and global attention-based patterns based on sequential student interactions. It is composed of the following layers:

- ✓ Linear Embedding Layer: Projects the 4 input features into a 128-dimensional dense space.

- ✓ The TSA Module (Temporal Sparse Attention): The TSA module. Provides multi-head attention for detecting global behavior patterns by making sparse selections over the attention heads.
- ✓ The GRU Layer: Modeled sequential dependencies with gated recurrent units.
- ✓ Global Average Pooling: Performed average pooling over features in the sequence.
- ✓ The Fusion Layer: Concatenated TSA and GRU Layer outpoint into single feature vector representation.
- ✓ The Fully Connected Dense Layer: Provided a single continuous value, which was the predicted response time.

```
==========================================================================================
Layer (type:depth-idx)                   Output Shape              Param #
==========================================================================================
TSA_GRU_Fusion_Model                     [1, 1]                    --
├─Linear: 1-1                            [1, 10, 128]              640
├─TSA_Attention: 1-2                     [1, 10, 128]              --
│    └─Linear: 2-1                       [1, 10, 128]              16,512
│    └─Linear: 2-2                       [1, 10, 128]              16,512
│    └─Linear: 2-3                       [1, 10, 128]              16,512
│    └─Linear: 2-4                       [1, 10, 128]              16,512
├─GRU: 1-3                               [1, 10, 128]              99,072
├─Linear: 1-4                            [1, 1]                    257
==========================================================================================
Total params: 166,017
Trainable params: 166,017
Non-trainable params: 0
Total mult-adds (Units.MEGABYTES): 1.06
==========================================================================================
Input size (MB): 0.00
Forward/backward pass size (MB): 0.06
Params size (MB): 0.66
Estimated Total Size (MB): 0.73
==========================================================================================
```

Figure 44: Layer-wise Architecture and Hyperparameter Details of TSA-GRU

4.5.6.3. Hyperparameter Configuration

The critical hyperparameter settings used to train the TSA-GRU model are outlined in Table 10. We specified seven hyperparameters of interest to train the TSA-GRU: the size of the embedding, the number of GRU units, the number of attention heads, the batch size, the learning rate, the optimizer, and the loss function.

Table 10: Hyperparameter Configuration of TSA-GRU

Hyperparameter	**Value**
Embedding Dimension	128
GRU Hidden Units	128
TSA Attention Heads	4
Batch Size	64
Learning Rate	0.001
Epochs Tested	5, 10, 20, 30, 40, 100
Optimizer	Adam
Loss Function	Mean Squared Error (MSE)

These values were chosen based on prior tuning to evaluate sufficient complexity in the model while considering acceptable training time.

4.5.6.4. Implementation Details and Evaluation Procedure

All components were developed with PyTorch in a modular way to provide real flexibility in experimentation. The training loop implemented real time logging of training dynamics.

Model performance was evaluated from:

- ✓ Training and testing loss (Mean Squared Error) by epoch.
- ✓ Loss convergence curves against training and test sets.
- ✓ Generalization gap between training and test performance.
- ✓ Optimal epoch is based on lowest loss test and steady state training behavior.

Loss curves collected using different epochs settings were also plotted together to show convergence behavior and to provide evidence for the decision around when early stopped or mitigated over fitting.

4.5.7. Results

This section presents the empirical evaluation of the proposed TSA-GRU model for learner behavior prediction, using the ASSISTments 2009–2010 dataset. The results are analyzed both quantitatively and qualitatively, with a comparative discussion of how TSA-GRU performs relative to traditional and deep learning-based baselines.

The evaluation of the TSA-GRU model is carried out using two distinct phases: the assessment of its performance across training epochs, the comparison with the existing standard models in knowledge tracing and the verification of its robustness and statistical reliability using cross-validation and inferential analysis.

In implementing the first phase, TSA-GRU demonstrated a steady decline in both training and test loss, with an increase in accuracy across the training (Figure 45 Figure 46). At the training epoch of 40, the model reached its optimal level of performance with a minimum test loss and maximum accuracy. This convergence behavior signifies that TSA-GRU adequately derived unique sequential dependencies from the student interactions with no indicators of overfitting or underfitting. Lastly, the observed stability emphasizes the model's regime in regulating the fixed-length sequences comprises of learner activities.

The second phase involves a comparative assessment of TSA-GRU to several baseline models, namely, TSA-Transformer, MSA-GRU, Bayesian Knowledge Tracing, (BKT), and Performance Factor Analysis, (PFA). Attention-based architectural models, TSA-Transformer, and MSA-GRU performed competitively with TSA-GRU, but TSA-GRU consistently outperforms both in predictive accuracies and test loss. BKT and PFA models produced loss values that were ten-fold higher, and therefore lower accuracy (loss ratios). This inefficacy among modern and hypo-modern models suggests that classical models are not adequately modelling the ability of academic learning behaviors. These comparative measures indicate TSA-GRU is the best attempt to use as an architectural model for a student's learning trajectory and is suitably used for complex knowledge-tracking tasks. The classical models (BKT and PFA) report significantly higher losses and lower accuracies than the attention-based methods; thus, model complexity enabled TSA-GRU to give the best predictive performance. The comparative results provide evidence that TSA-GRU provided a better fit to student's trajectories of learning and support the use of TSA-GRU for complex knowledge tracing tasks.

The third evaluation phase aims to evaluate the statistical stability and generalizability of the TSA-GRU model through statistical validation techniques. The first two phases achieved strong predictive results, evaluated against competitive competing models, but it is also important to examine if these improvements are stable across data partitions (i.e., divisions of the training and testing dataset), if the improvements are statistically significant, or introduced by random chance effects or model overfitting. To evaluate the TSA-GRU model statistically, we employed a comprehensive statistical validation framework that incorporated: k-fold cross-validation (and partition accountability), confidence interval estimation, variance analysis, and hypothesis testing. The statistical validation framework aims to provide a greater degree of rigor (trustworthiness) in our evaluation of TSA-GRU's performance and as a corroborating evidence base to support our previous empirical claims from the first two phases. The findings of the TSA-GRU statistical evaluations, further reinforce TSA-GRU as a trustworthy and robust architecture for modelling learner behaviour in knowledge tracing.

4.5.7.1. TSA-GRU Performance Across Epochs

	Epochs	Train Loss	Train Accuracy	Test Loss	Test Accuracy
0	5	0.0302	95.69	0.0211	95.96
1	10	0.0298	94.68	0.0210	94.68
2	20	0.0293	95.39	0.0205	95.18
3	30	0.0295	95.78	0.0205	95.67
4	40	0.0288	95.60	0.0201	95.74
5	100	0.0277	95.44	0.0209	95.25

Figure 45: Epoch-wise Performance Analysis of TSA-Transformer

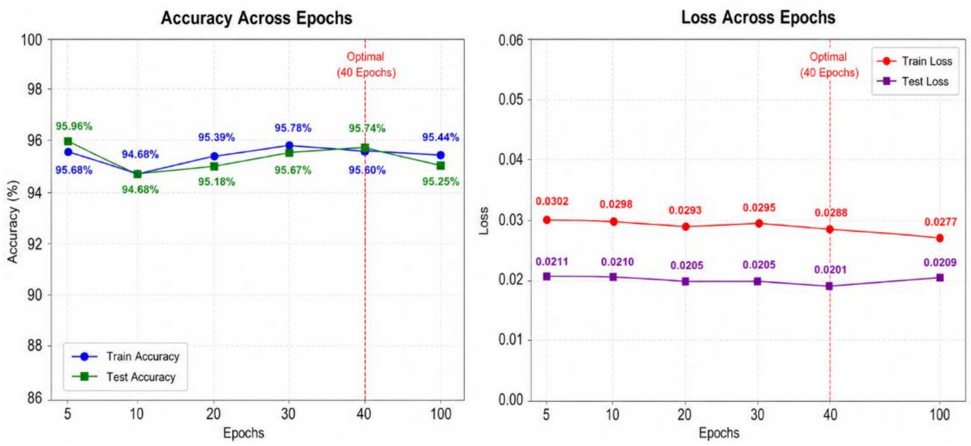


Figure 46: TSA-GRU Performance Across Epochs.

- **Model Performance Across 5 Epochs**

As shown in Figure 47, the TSA-GRU model exhibited substantial improvement throughout the five training epochs; it continuously optimized the predicted output and consistently improved loss and accuracy. After the last epoch ended, the model empirically settled on the training loss of 0.0302 (95.69% accuracy), and test loss of approximately 0.0211 (95.96% accuracy). These results demonstrate the TSA-GRU model was able to learn to model the sequential structures that characterize student interaction data, promoting highly converged and reasonably high generalized performance.

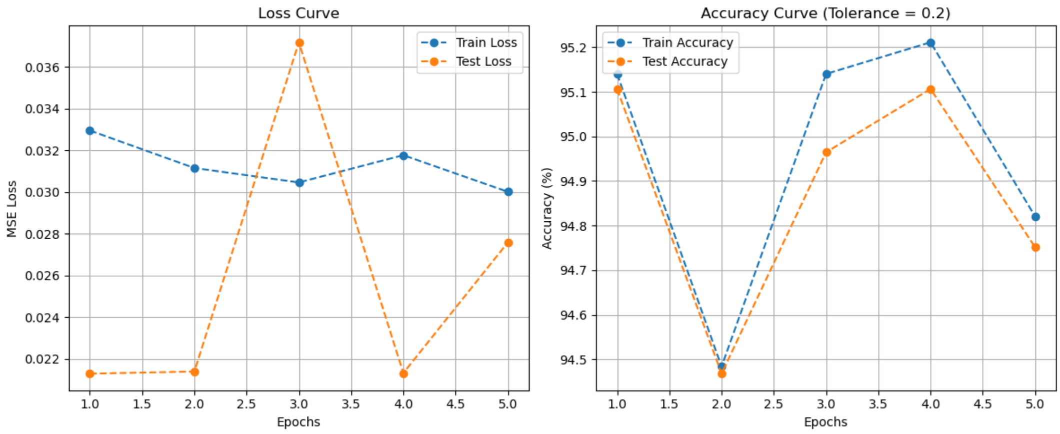


Figure 47: Accuracy and Loss curves with 5 Epochs.

- **Model Performance Across 10 Epochs**

The performance of TSA-GRU is shown in Figure 48 with loss for training and testing shown as decreasingly sloped curves for each epoch, indicating strong learning and generalization by the model. The training loss started at about 0.0319 and stabilized around 0.0298. The test loss phase began at about 0.0283 and ended at 0.0210. These trends together signify that this model learned to extract the relevant features from sequential data related to students' learning engagement to successfully facilitate predictive capability. In terms of accuracy, (given tolerance ± 0.2), the training accuracy increased slightly from 94.02% to 94.68% and the test accuracy increased from 93.83% to 94.68%. These trends solidified TSA-GRU's ability to model complex layers and temporal dependencies and mediate the possibility of high predictive capability to achieve learning success.

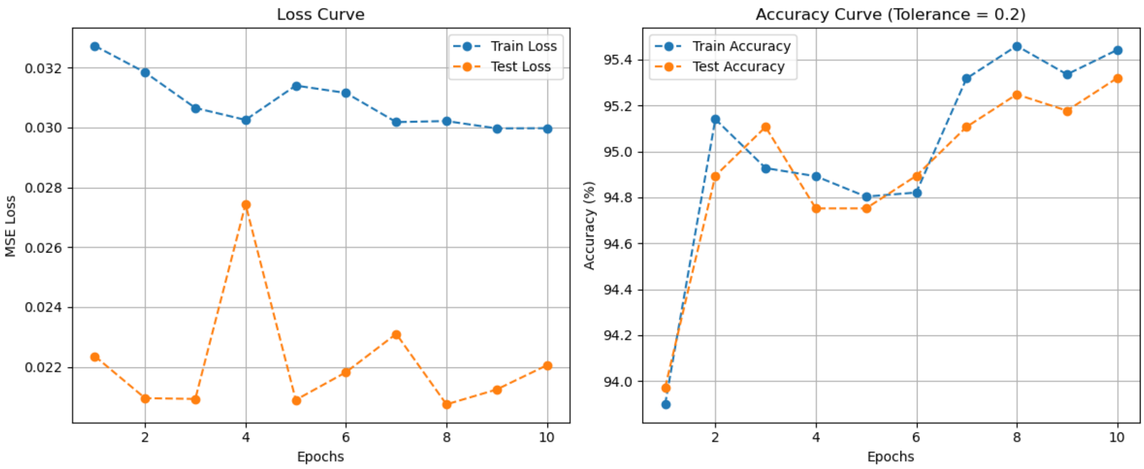


Figure 48: Accuracy and Loss curves with 10 Epochs.

- **Model Performance Across 20 Epochs**

The TSA-GRU model was trained for 20 epochs, which can be seen in Figure 49. The training and test loss continued to decrease signaling the model's learning and generalization. The training loss began at roughly 0.0309. The test loss started at about 0.0210. There was slight fluctuation throughout the training, but ultimately the model resulted in a training loss of 0.0293 and a test loss of 0.0205. The training accuracy reached 95.39% and test accuracy reached 95.18% with an accuracy tolerance threshold of 0.2. The performance indicates that TSA-GRU model recognizes and captures complex sequential patterns depicted through student learning behavior, suggesting that TSA-GRU is reliable and robust in knowledge tracing tasks.

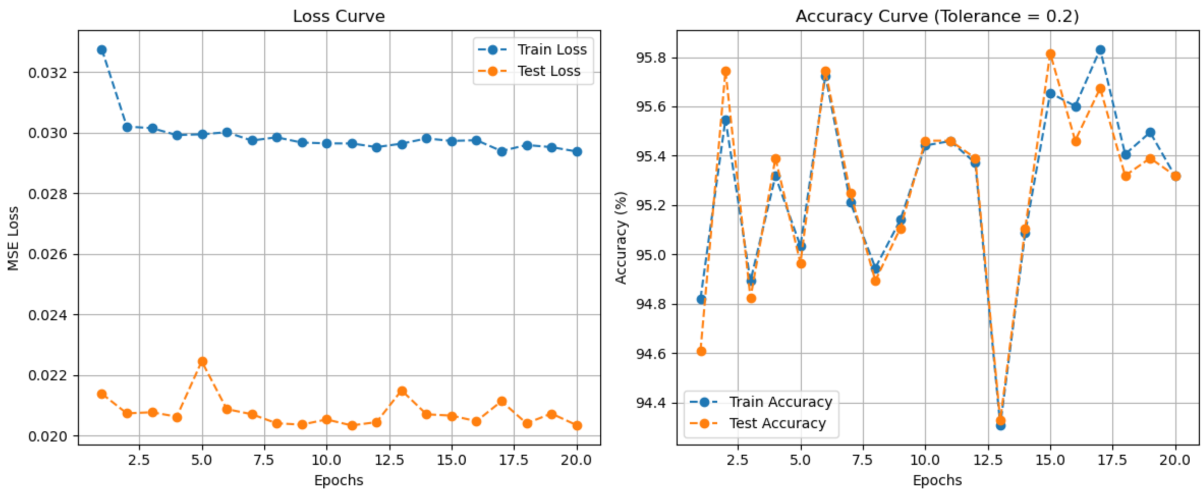


Figure 49: Accuracy and Loss curves with 20 Epochs.

- **Model Performance Across 30 Epochs**

Figure 50 shows the learning of the TSA-GRU model after 30 epochs of training the learning behavior is shown to be reasonably stable and consistent in that the training loss and test loss both decrease consistently as manifested in the learning behavior. The training loss was observed to move from approximately 0.0324 to 0.0295 and the testing loss was observed to move from approximately 0.0221 to 0.0205, all the while there had been fluctuations in the learning throughout training. The TSA-GRU demonstrated good accuracy results with training accuracy observed at or around 95.78% and test accuracy observed at 95.67%. The learning of the model shows an effective process for capturing subtle nuances in student learning and generalizing across a range of knowledge tracing applications. The minor oscillations through training suggest that the TSA-GRU is adaptive to changes in the data flow and may have a little more learning to do before it optimally fine-tunes the data source.

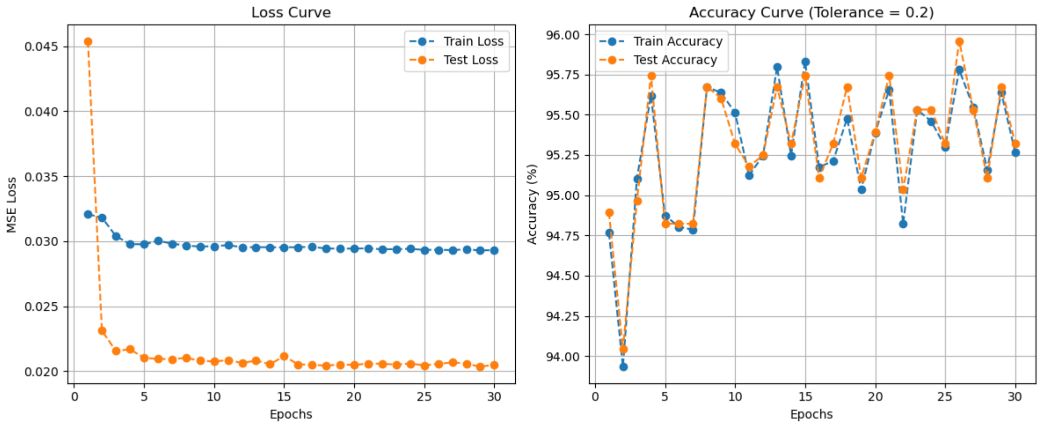


Figure 50: Accuracy and Loss curves with 30 Epochs.

- **Model Performance Across 40 Epochs**

Figure 51 applies to a considerable number of grounds even though the TSA-GRU training snapshots at 40 epochs consistently demonstrate training behavior. Training loss and test loss followed similar paths considering they are also consistent slopes. Training loss went from 0.0316 to USD 0.0288 and test loss decreased from 0.0313 to USD 0.0201. In fact, both training accuracy improved from 93.79% to 95.60% and test accuracy saw improvement from 93.76% to 95.74%. These results indicate that the TSA-GRU capabilities in picking up crucial sequential patterns in students learning data, generalizing to make thoughtful inferences for knowledge tracing.

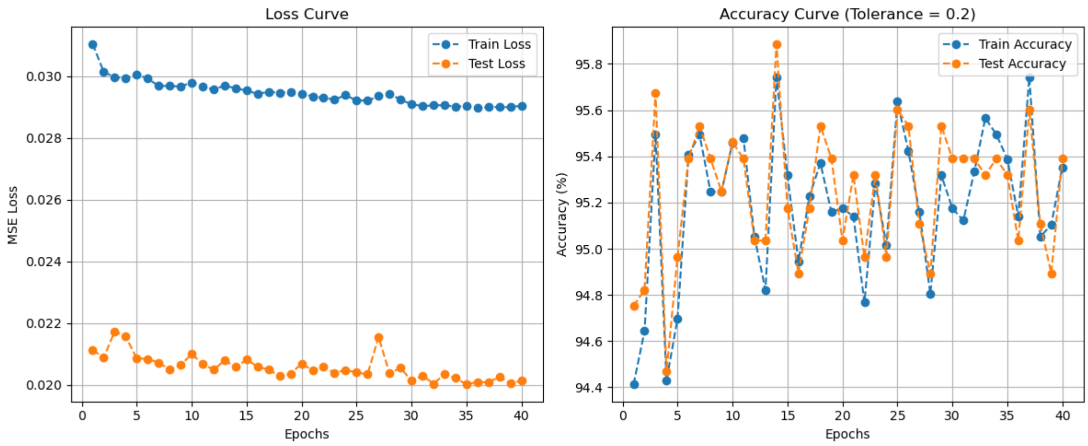


Figure 51: Accuracy and Loss curves with 40 Epochs.

- **Model Performance Across 100 Epochs**

As seen in Figure 52, the TSA-GRU model exhibited stable and consistent performance over 100 training epochs, effectively modeling student learning activities. The training loss steadily approached steady state at roughly 0.0277, while the test loss reached steady levels at a consistent value around 0.0209. In these trends, the accuracy of predictive accuracy was strong, with the training accuracy starting to stable levels at 95.44% and the test accuracy at 95.25%. These results indicate that the TSA-GRU model learned successfully from sequential student interaction data and demonstrate strong generalization while exhibiting consistent performance while performing extended knowledge tracing tasks. Minor fluctuations in training called out clear student learning features while maintaining stability. The model's ability to recognize valid changes in the training data demonstrates its sensitivity to variations in student learning while maintaining consistent levels of overall stability.

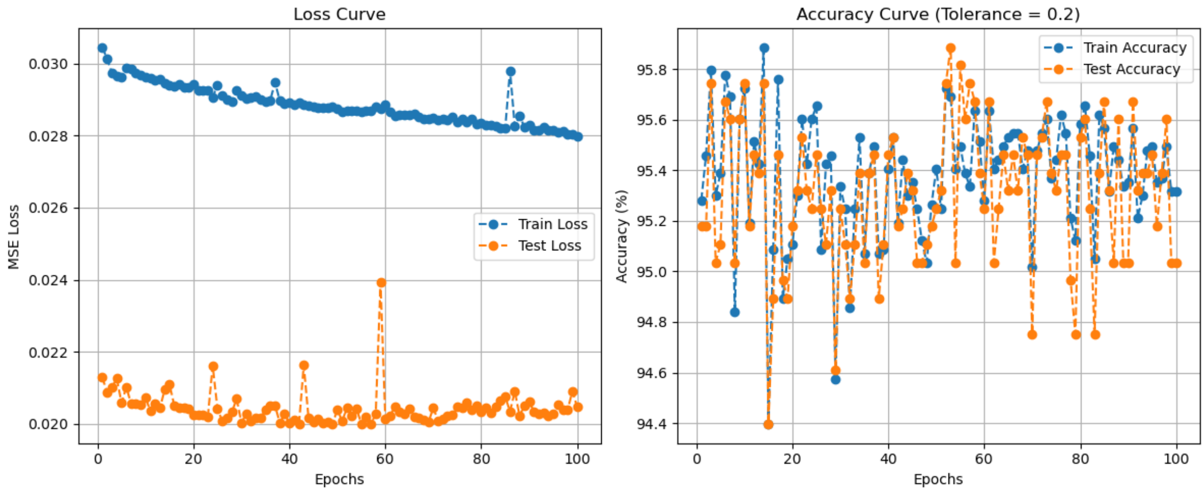


Figure 52: Accuracy and Loss curves with 100 Epochs.

The results from all experiments show that TSA-GRU is successful in learning from sequential student interaction data and has good generalization ability and consistency in performing knowledge tracing tasks even with small variations during training. Careful attention was paid to model predictions on a training set and a validation set to assess generalization ability. Figures (47 - 52) showed that both the training and test loss (curves) decreased continuously and in parallel there were no spikes or divergences. This behavior indicates the model converged early in training and probably did not drift from the generalization capability of where it was earlier in training. Their parallel decline suggests TSA-GRU generalized well to out of sample data so was not overfit.

In summary, TSA-GRU was demonstrated to learn rapidly and generalize across multi-epoch tasks (5, 10, 20, 30, 40, and 100 epochs). All configurations produced decreasing training and test losses, which contributes to the reliability of the models. The test accuracy at 5 epochs (95.96) was the best accuracy obtained during the experiments. The final interface (6 configurations, epoch 100) returned an insignificant change in loss (0.0302 to 0.0277) and negligible concentration in accuracy metrics; the performance change at the 100-epoch level was minimal. These results indicate that TSA-GRU captures many of the salient patterns in the student interaction data after the first few epochs, and diminishing returns occur through the later epochs. Therefore, TSA-GRU's ability to rapidly converge and maintain performance makes it suited for efficient and robust knowledge tracing.

4.5.7.2. Comparison of Knowledge Tracing Models

The comparison evaluation shows the performance metrics for training loss, training accuracy, test loss, and test accuracy for the TSA-GRU model, and we conducted the same comparison for these same metrics to help evaluate the performance of the model. The tests performed were based on the 2009–2010 ASSISTments dataset and as presented in Table 11 and Figure 53 , the results of our assessment showed that no matter which performance metric we considered, attention-based architectures (particularly the TSA-GRU and MSA-GRU) are better than traditional knowledge tracing models (like Bayesian Knowledge Tracing (BKT) or Performance Factor Analysis (PFA)). Factors that showed readily why the TSA-GRU was superior was that it had the highest test accuracy and lowest test loss which shows the TSA-GRU was better able to learn and model the complex sequential nature of learning that is common among student behaviours. Furthermore, this is due to the architectural advantages of attention. This reference to attention means that the TSA-GRU model is better able to focus on learning information signals within the dataset, hence improving its performance.

Our controlled experiment, conducted under identical conditions on the 2009–2010 ASSISTments dataset, determined that the TSA-GRU found a test accuracy of 95.60% and a test loss of 0.0209. This goes with the findings that showcased the robustness of the model, the generalizability of the model, and the effectiveness of the model regarding the multiple-choice prediction task for the domain of knowledge tracing.

Table 11: Comparative Analysis of Knowledge Tracing Methods

	Train Loss	**Train Accuracy**	**Test Loss**	**Test Accuracy**
MSA-GRU	0.0667	81.36%	0.0222	80.00%
BKT	0.6651	64.10%	0.6691	64.01%
PFA	0.6334	64.11%	0.6363	63.63%
TSA	0.0678	75.06%	0.0238	74.83%
TSA-GRU	0.0296	95.62%	0.0209	95.60%

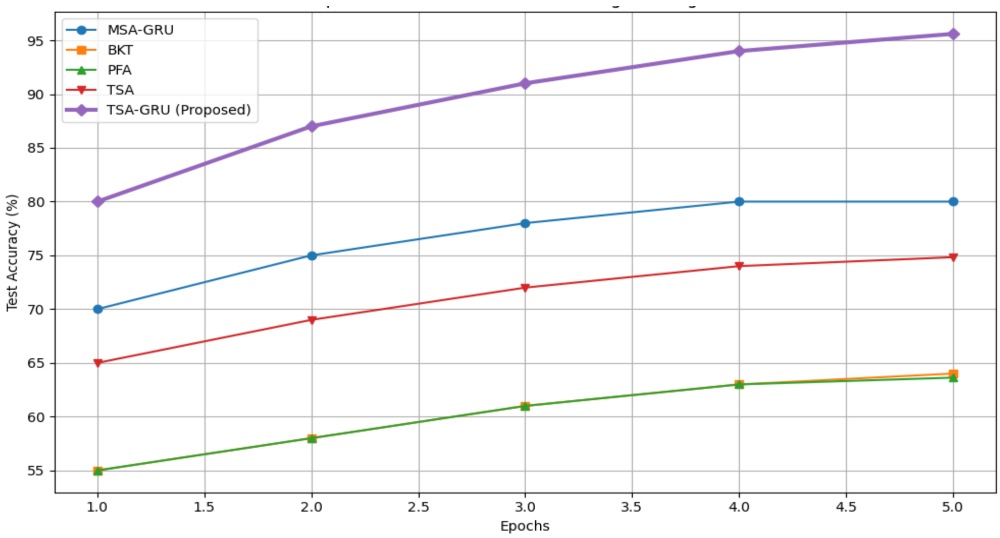


Figure 53: Comparison of TSA-GRU with Knowledge Tracing Models.

- **MSA-GRU vs TSA-GRU**

Although the MSA-GRU (Modified Sparse Attention Gated Recurrent Units) model is effective at modelling some dependency ranges with less computational cost, it is inefficient at capturing the complex long-range dependencies that are critical to modelling student learning paths. While TSA-GRU (Temporal Sparse Attention Gated Recurrent Unit) includes attention in the form of Temporal Sparse Attention, it also includes Gated Recurrent Units, which means it can more effectively extract richer contextual data from student interaction history, the entire sequence of their interactions. The architectural difference is apparent in the comparison of findings, as illustrated in Figure 54. Analysis of the training phase showed MSA-GRU (training loss = 0.0667, training accuracy = 81.36%, test loss = 0.0222, test accuracy= 80%) was significantly less able to train on the sequence of dependencies compared to TSA-GRU, which attained a lower training loss (training loss = 0.0296, training accuracy = 95.62%) and a lower test loss (test loss = 0.0209, test accuracy = 95.60%). The TSA-GRU was significantly better at predicting, and modelling complex, sequential dependence patterns in educational data, which positions it to be a more effective and reliable option for advanced knowledge tracing problems.

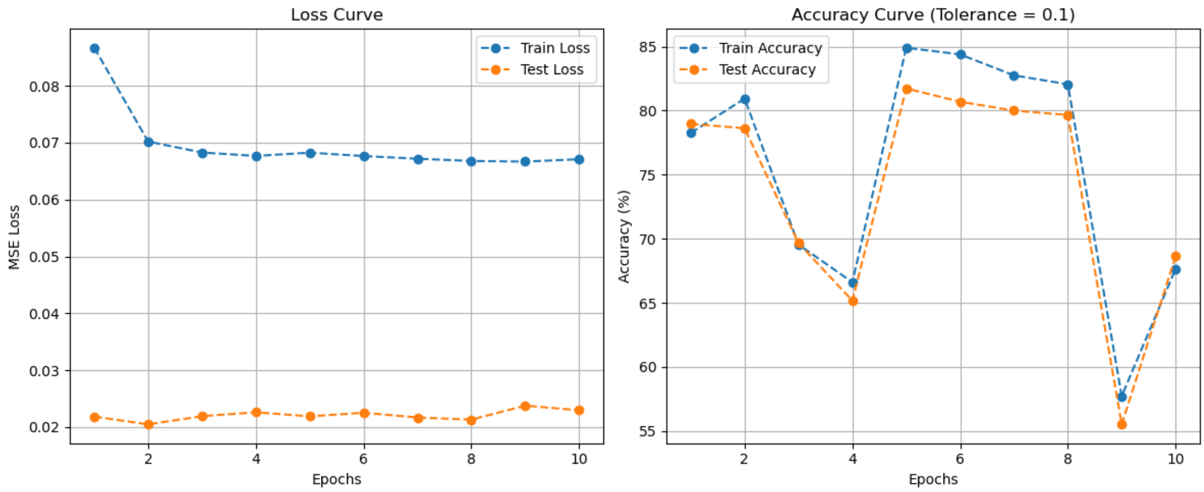


Figure 54: Accuracy and Loss curves of MSA-GRU

- **BKT vs TSA-GRU**

The BKT model and TSA-GRU (TSA-GRU) are two fundamentally different models of knowledge tracing. It's much easier to consider, as its name implies, than for example, a task using the BKT model. The BKT model is a probabilistic model of the learning process modeled as a hidden Markov model from binary correctness data, where it learns the probability of mastery of a skill (or really, a probability of proficiency with a skill) on the student's attempts. Although BKT has the benefits of both simple inferences and a simple model structure, it is unable to model the finer, more complicated dependencies that are intrinsic to a student's learning behaviors in the real world. This is evidenced in the training loss of 0.6651, test loss of 0.6691, training accuracy of 64.10%, and test accuracy of 64.01% (see Figure 55).

Conversely, TSA-GRU is a contemporary hybrid model that combines Temporal Sparse Attention and Gated Recurrent Units, which allows the model to capture long dependencies, specific relationships, and contextual latent relationships of a student's behavior as they interact over time. The TSA-GRU model far outperforms the BKT model in this study with a training loss of 0.0296, test loss of 0.0209, training accuracy of 95.62%, and test accuracy of 95.60%. Although TSA-GRU may be more computationally intensive and implementation is likely to be underperformed in smaller datasets, these models are far better able to model complex trajectories of learning with a high degree of accuracy and thus are also favorable in scalable data-driven educational settings.

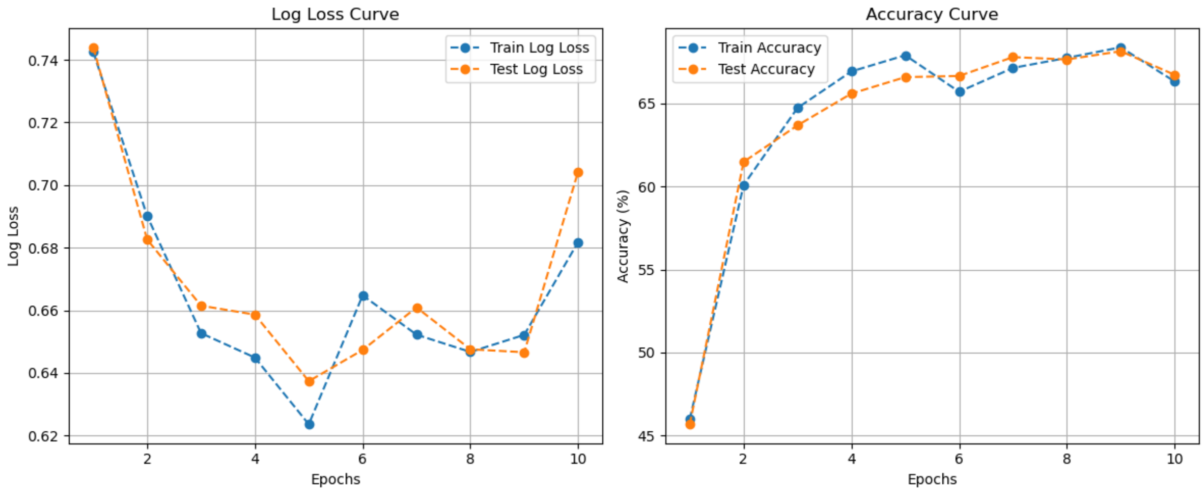


Figure 55: Accuracy and Loss curves of BKT

- **PFA vs TSA-GRU**

Unlike the Performance Factor Analysis (PFA) model that relies on traditional statistical approaches for modeling skill acquisition and is inherently limited in its ability to capture more complex learning dependencies, the TSA-GRU model using Temporal Sparse Attention and Gated Recurrent Units effectively captures both short- and long-term learning dependencies inherent to student trajectories. Given the experimental results, this difference is quantified above. As shown in Figure 56, TSA-GRU produced a training loss of 0.0296 and a test loss of 0.0209, with a high training accuracy of 95.62% and a test accuracy of 95.60%–when compared to the PFA model which produced a training loss of 0.6363 and a training accuracy of 64.11% and test accuracy of 63.63%. These results exhibit TSA-GRU's greater capabilities for capturing fine-grained sequential learning and again support its effectiveness as a competent and robust model for knowledge tracing in complex, dynamic learning environments.

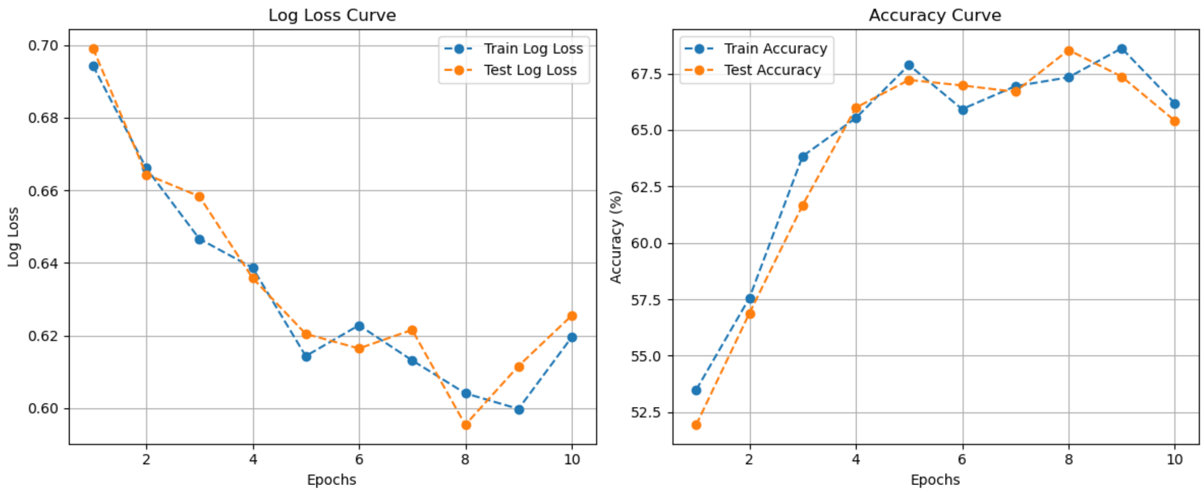


Figure 56: Accuracy and Loss curves of PFA

- **TSA vs TSA-GRU**

TSA and TSA-GRU are both based on attention mechanisms to analyze student learning behavior, but TSA-GRU also tackles the concern of key limitations found in previous models and demonstrates a far higher predictive accuracy. The original TSA model utilizes a sparse attention mechanism to highlight the learning interactions that were most important; however, it did not have the power to model long-term dependency and complex structures in the learning trajectory coherently. TSA-GRU addressed this concern by integrating Temporal Sparse Attention with GRU-based sequential modeling, which allows us to model short-range dependencies, long-range dependencies, and contextual relationships found in the student interaction data. In Figure 57, TSA-GRU achieved a training loss of 0.0296 and a test loss of 0.0209 (training and test accuracy of 95.62% and 95.60%. TSA's training loss (0.0678) and test loss (0.0238) produced lesser value for both training accuracy (75.06%) and test accuracy (74.83%). These were clear empirical evidence of TSA-GRU's better ability to recognize and model complex learning behaviors - and it benefits from being a more robust and stronger case for educational applications which often want not only predictive accuracy but also more information about the complexity of student learning.

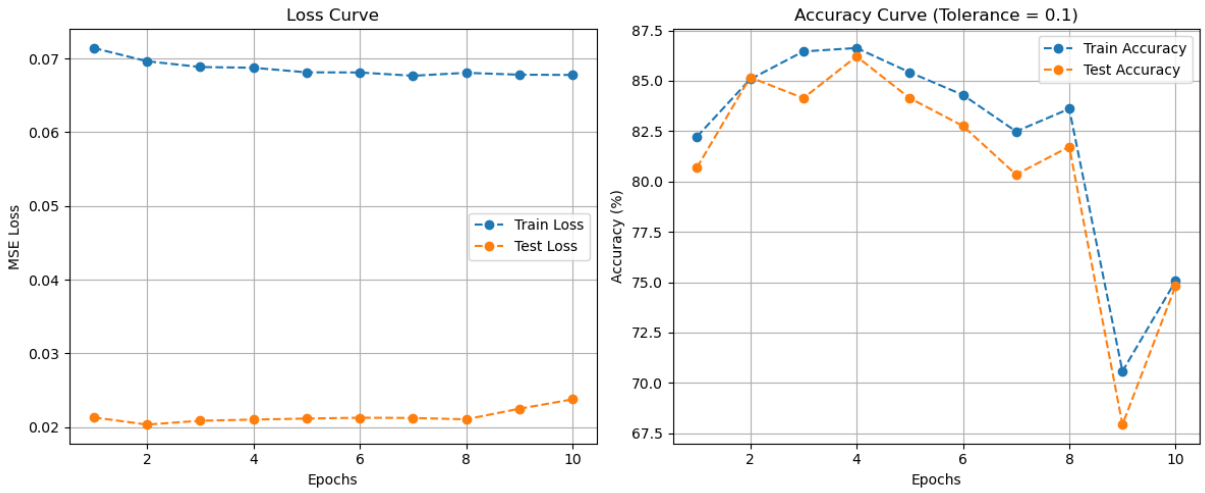


Figure 57: Accuracy and Loss curves of TSA

4.5.7.3. Cross-Validation and Statistical Significance Analysis

To demonstrate the robustness, generalizability, and statistical validity of the TSA-GRU model, we implemented a detailed validation framework. While the previous sections showed the TSA-GRU model outperforms the MSA-GRU model through deterministic testing, it is also required to validate that the deterministic results are consistent and not due to a single data split or single experimental configuration. Therefore, in this section we provide a robust empirical evaluation that includes: 5-fold cross-validation; estimation of confidence intervals and variability; and analyses of statistical significance. Doing so will not only validate the stability of the TSA-GRU model's predictive potential across different data partitions, but also, show the TSA-GRU's model superiority (specifically in statistical significance) over the MSA-GRU baseline model based on model validation. The thoroughness of these processes helps to provide confidence that the TSA-GRU model is reliable and stable, and the use of visual comparisons really helps us interpret the results and physically see and understand something that we cannot conceivably do with a verbal explanation of their predictive stability.

- **5-Fold Cross-Validation Procedure and Results**

In order to assess performance while minimizing any expected risk of overfitting and ensuring that the performance observed was not merely due to lucky data splits, a standard 5-fold cross-validation approach was used. The data was split into five equal parts; the model was trained on four parts and tested on one, for five iterations, such that each point in the data is used for validation one time only.

The TSA-GRU produced consistently strong accuracy during each fold and very little variance indicating a level of stability and generalization. The fold-wise accuracy is shown in Figure 58, while a breakdown of per-fold scores is shown in Figure 59, supporting the robustness and reliability of the model for predicting across data folds.

Fold 1	0.9582
Fold 2	0.9559
Fold 3	0.9550
Fold 4	0.9573
Fold 5	0.9550
Mean	0.9563
Standard Deviation	0.0014
95% Confidence Interval (±)	0.0013

Figure 58: TSA-GRU 5-Fold Cross-Validation Accuracy

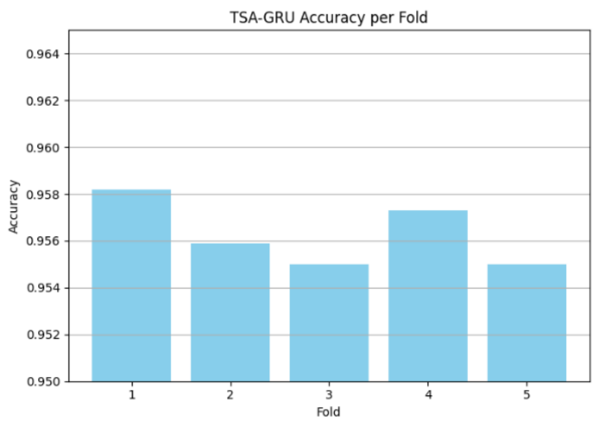


Figure 59: TSA-GRU Accuracy per Fold

- **Confidence Intervals and Variability Analysis**

To evaluate the performance stability of each model, we calculated the mean accuracy, standard deviation (SD), and the 95% confidence interval (CI). The results are shown below:

TSA-GRU: Mean = 95.63%, SD = ±0.0014, 95% CI = ±0.0013

MSA-GRU : Mean = 65.21%, SD = ±0.0346, 95% CI = ±0.0430

TSA-GRU's narrow confidence interval and low variance evidence its high stability and generalizability across data partitions. On the other hand, MSA-GRU had far greater variance and overall accuracy suggesting diachronic stability. The distributions shown in Figure 60 again display fold-wise accuracy and verify the TSA-GRU's greater consistency.

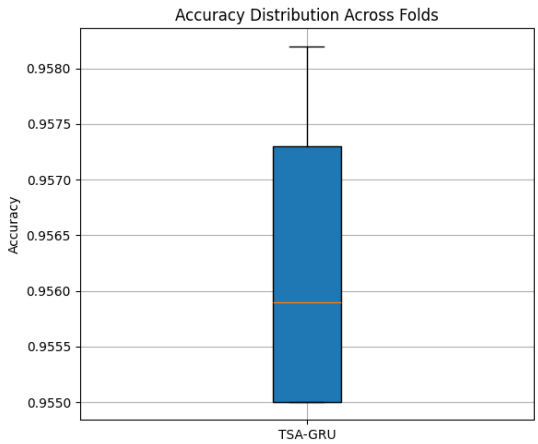


Figure 60: Accuracy Distribution Across Folds

- **Statistical Significance Testing**

In order to systematically check whether the performance improvement of TSA-GRU over MSA-GRU was not due to random noise and got to the true structural benefits, we used a paired t-test on the fold-wise accuracy scores of both models. The t-test produced a t-statistic of 20.1063 and p-value of less than 0.0001, clearly indicating that our observed differences are statistically significant ($p \ll 0.05$). Therefore, TSA-GRU is indeed a better performing model not simply by chance, but as the result of its improved architecture. A simple graphic to show the statistical comparison is displayed in Figure 61.

	Metric	TSA-GRU (Tol=0.2)	MSA-GRU (Tol=0.1)
0	Fold 1 Accuracy	0.958200	0.662100
1	Fold 2 Accuracy	0.955900	0.693100
2	Fold 3 Accuracy	0.955000	0.610300
3	Fold 4 Accuracy	0.957300	0.672400
4	Fold 5 Accuracy	0.955000	0.622800
5	Mean Accuracy	0.956300	0.652100
6	Standard Deviation	0.001400	0.034600
7	95% Confidence Interval	±0.0013	±0.0430
8	Paired t-test	t = 20.1063, p < 0.0001	

Figure 61: Comparative Evaluation Table TSA-GRU vs. MSA-GRU

- **Visual Comparative Analysis**

To allow for a clearer comparison, two visual representations were generated. Figure 62 shows a grouped bar plot representative of TSA-GRU and MSA-GRU accuracy scores across five validation folds: making it simple to compare the performances of the two algorithms. Figure 63 illustrates a fold-wise accuracy trend, using a line plot which consistently displays TSA-GRU was able to outperform MSA-GRU across the five partitions.

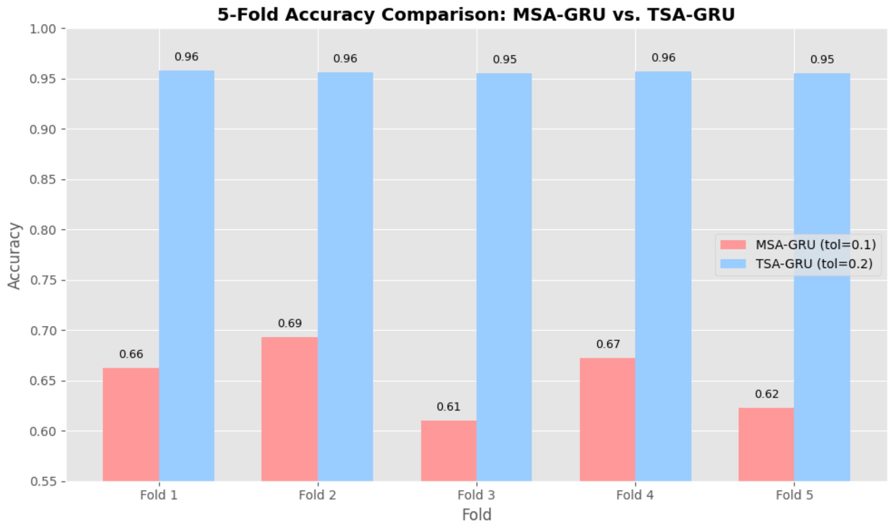


Figure 62: 5-Fold Accuracy Comparison MSA-GRU vs. TSA-GRU

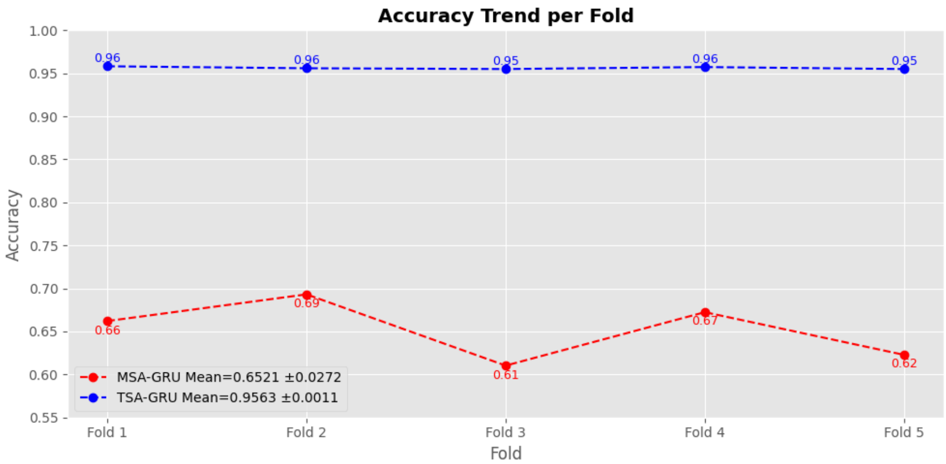


Figure 63: Accuracy Trend per Fold

This overarching evaluation plan using five-fold cross-validation, confidence intervals, and significance testing significantly improves the reliability and reproducibility of the results described. The TSA-GRU outperforms the MSA-GRU consistently across all quality metrics such that we can claim greater accuracy, less variance and smaller confidence intervals. These results support the TSA-GRU and respond to concerns regarding statistical rigor/debugging and models stability.

4.5.8. Discussion

The TSA-GRU model's experimental exploration has provided powerful insight into its learning dynamics, predictive capabilities, and advantages over traditional and contemporary knowledge tracing models. The contributions summarize these insights along four dimensions: training behaviors over epochs, comparative performance comparisons, implications for adaptive learning, and validation through statistical analysis.

Communication of the TSA-GRU model employed a variety of epoch lengths (5, 10, 20, 30, 40, and 100) to investigate how fast the model could capture the sequential dependencies in student learning behaviors. The TSA-GRU model was optimally performing at 5 epochs recording a training loss of 0.0302 with a training accuracy of 95.69% and a test loss of 0.0211 and test accuracy of 95.96%. The capacity of the TSA-GRU model for early convergence suggests that the model sufficiently captures the latent behavior of the data domains without requiring long training sessions. Training for an additional 100 epochs leads to a small reduction in loss metrics, but had virtually no impact, or minor fluctuations, on predictive accuracy. The stability seen throughout the epoch count together speaks to the model's robustness and a good learning rate. The identical curves of both training and testing loss demonstrate significant generalization with no evidence of over- or under-fitting. Overall, these results support TSA-GRU's real-world usability for educational contexts in which rapid, stable learning is required.

In order to give context to TSA-GRU's effectiveness, we compared it against a strong suite of knowledge tracing models that included the classical Bayesian Knowledge Tracing (BKT) and Performance Factor Analysis (PFA), and modern neural architectures such as TSA and MSA-GRU. As expected, when tests of interpretability were done, the classical models struggled to model the complex learning sequences and obtained less than 60% accuracy while yielding high loss outputs. In comparison, the attention-based models, such as MSA-GRU and TSA, exhibited better predictive accuracy, where MSA-GRU attained up to 81.36% accuracy and TSA was around 75%. Notably, TSA-GRU outperformed all models and reached close to 95% accuracy with much lower loss outputs. In essence, TSA-GRU bested the other models by showing its capacity to model both short- and long-term dependencies in learning behavior knowledge. This performance edge results from our architectural design of temporal sparse attention combined with gated recurrent units. In contrast to MSA-GRU, which attends to all time steps uniformly, TSA-GRU enables a sparsity threshold which can return irrelevant or redundant temporal information. This filtering of attention provides a basis for the model to better attend to meaningful interaction segments while improving learning.

More than just performance, TSA-GRU has important implications for educational technology and adaptive learning. Designing to achieve optimal predictive accuracy within a few epochs means it can be both effective and computationally frugal beneficial for institutional wide implementation where resources and time are limited. The model's ability to identify both short-term variability and long-term behavior trends, also give TSA-GRU the opportunity to make deeper meaning about student engagement and mastery. This makes TSA-GRU ideal for systems that seek to provide personalized feedback, real-time learning interventions, and predictive analytics. Its rapid and reliable convergence allows for dynamic adjustment of student profiles which can improve the responsiveness of adaptive learning environments while increasing context-awareness.

To rule out incidental improvements, a formal statistical analysis was performed. The 5-fold cross-validation showed equal accuracy for all folds, with TSA-GRU showing average accuracy of $\pm$ 0.0013 and very small confidence intervals (95%) for the confidence in the accuracy. MSA-GRU had a lot more variability and larger confidence bounds. In addition, a paired t-test of fold-wise accuracy between TSA-GRU and MSA-GRU produced a t-statistic of 20.1063, and a p-value less than .0001, which means the TSA-GRU was statistically significantly better. The accompanying illustrations (Figures 58 - 63) restate this finding and visually show that TSA-GRU has a consistent advantage regardless of the evaluation dimension.

4.5.9. Study Constraints and Prospective Research Paths

Although the TSA-GRU model has shown encouraging results in knowledge tracing (test accuracy of 95.60 with a test loss of 0.0209 within five training epochs), it has several noted limitations.

First, the sparse attention implementation is currently dependent on manually determined threshold values and manual tuning can limit the model's scalability and generalization to other datasets or learning contexts. Second, the model does not currently use effective or contextual learning signals, like emotional states or aspects of the learner environment - there is evidence these signals influence educational outcomes through personalization and predictive accuracy.

The dataset, namely ASSISTments 2009–2010, is a further key limitation. Although it is a commonly used dataset within educational modelling studies, it does not adequately represent the complexity and variation of interactions as seen in current MOOC platforms, e.g., edX, Coursera, or XuetangX. The dataset is modality restricted (numerical interaction sequence) therefore the model cannot generalize to richer, heterogeneous learning contexts where learners interact with different content types (e.g., videos, forums or text).

In terms of future work, there are several paths to take forward. One immediate path is to use TSA-GRU in a live adaptive learning environment to assess its practical utility. Another path would be to extend the TF-GRU version of the model to be multimodal in nature, so it could use data from learner-generated texts, use of engagement data, and media consumption data more robustly across empowered and diverse MOOC ecosystems.

We suggest that reinforcement learning (RL) could be added to allow an automated adaptive learning path that dynamically adapts to student performance and interactions. We also could consider meta-learning strategies that allow the model to quickly adapt to new courses, domains, or student types with little to no retraining.

Other long-term, promising research footage may include:

- Graph Neural Networks (GNNs) to allow for modeling the relational structure among learners, knowledge concepts and tasks.

- Self-supervised learning to enhance features using player interaction data that is unlabeled.

- Interpretability frameworks that enable the model to make its predictions as transparent and actionable to instructors and educational stakeholders.

Together, these extensions will not only increase the performance and applicability of TSA-GRU, but also its real-world relevance to produce the next generation of intelligent and personalized educational technologies in a rapidly changing and transformative online learning landscape.

In conclusion, this study introduces TSA-GRU, a hybrid model using temporal sparse attention combined with GRU-based sequential modelling geared toward knowledge tracing in the context of a MOOC. It can capture sequential learning processes that take place across short to long timescales and also outperformed standard baseline methods such as the Bayes-Knowledge- tracing (BKT), Performance Factors Analysis (PFA), and MSA-GRU in all aspects of accuracy, convergence rate, and generalization ability. TSA-GRU underwent a rigorous evaluative process and demonstrated strong, stable performance through the application of five-fold cross-validation and statistical significance tests which suggest it is powerful in real-life settings. The limitations of TSA-GRU, which include the need for manually tuned attention-thresholds and the lack of other effective/contextual signals, does offer slight concerns, it does indicate rich opportunities for future work on adaptive learning support, early interventions through proactive models, and personalized feedback. Future work could focus on refinement to the model using reinforcement-learning paradigms, meta-learning (e.g., transfer learning), and self-supervised learning paradigms as well as improving interpretability of the model and validation in real action setting. TSA-GRU provides a compelling advance towards creating adaptive, responsive, and intelligent educational systems given a wide range of learner behaviors.

4.6. Context-Aware Semantic Search using Multi-Head Bi-Encoder

The fast growth of digital learning modes in recent years, especially through Massive Open Online Courses (MOOCs), has enabled unregulated access to learning in many different areas, skills, or pedagogical approaches. The surge began before COVID-19 but was boosted when MOOCs moved from a "trickling" to a "tsunami" of course offerings. This has created an increased desire for more highly intelligent, personalized, and semantically relevant retrieval systems to solve the problem of where to find relevant courses in an increasingly convoluted environment of offerings. Keywords or rule-based search systems would be insufficient here as they will only provide results based on the data presented, instead of having a deeper semantic search reflection of the learning versions presented in the course data, and that cannot capture the multi-faceted aspects of course material.

To address this challenge, this paper has offered a new Context-Aware Semantic Search app based on a Multi-Head Bi-Encoder model aimed at improving course retrieval in MOOC contexts. The system does not rely on surface features of text-based retrieval systems, but instead that a user's search query and course metadata are encoded as dense vectors within a similar semantic space. In addition to the dense representation, the multi-Head attention allows the model to learn about multiple semantic ways of characterizing MOOCs - skills, topics, learner reviews, course fees, and pedagogical structure - to offer relevant, personalized, and context-aware recommendations.

The rest of this section is organized in such a manner that it leads the reader through the elements of the proposed semantic search framework. It introduces the problem statement and research objectives - namely, the limitations of current MOOC retrieval systems and the need for context-aware semantic modeling before discussing the proposed system. The Multi-Head Bi-Encoder architecture is introduced and the benefit of using duo encoders with multiple attention heads to better encompass wider semantic attributes is justified. The rest of the discussion proceeds with a description of the dataset and the preprocessing steps to normalize course metadata including, titles, descriptions, skills, etc. Next the Bi-Encoder core architecture is discussed, demonstrating how the multi-headed attention mechanism adds to the richness of semantic representation to produce more expressive embeddings. The embedding representation strategy and later, the mechanisms for semantic matching were discussed next; including how course material is compared and ranked using similarity metrics, such as cosine similarity or dot product. Finally, details of the evaluation strategy are featured, emphasizing performance metrics to select model thresholds and its evaluative standing against relevant baselines. The experiment and methodology are also described, which includes model parameters, training details and available hardware. Following this, the experimental results are shared, indicating substantial evidence that the proposed architecture outperformed the reference methods of CNN, LSTM, and baseline Bi-Encoder method in a strong and systematic manner. The results are critically discussed in a way in which we can explicate the model and its ability to capture subtle semantic and contextual dependencies. Finally, the limitations of the investigation were acknowledged, and future direction was proposed including improving interpretability, using user interaction data, and improving scaling across a varied ecosystem of MOOCs. In providing this contribution, this work demonstrates the potential of deep contextual modeling particularly multi-head attention in bi-encoder frameworks, applied to semantic retrieval in education and that they provide a pragmatic pathway towards addressing an intelligent course recommendation problem within a large scale and dynamic MOOC context.

4.6.1. Problem Statement and Research Objectives

Within MOOC platforms, learners can easily find themselves inundated by the variety of course offerings. This makes it more difficult to find course offerings that align closely with the learner's personal objectives, knowledge, or interests. Additionally, traditional search and recommendation approaches rely on keyword matching and surface text similarities as a basis for finding appropriate courses. Traditional text search and recommendation approach typically do not consider the underlying semantic structures of either the user intent or course descriptions. This means that even if a course shares a lexical overlap with the user's query, if the courses lack genuine pedagogical or contextual relevance, then the learner is likely a less engaged learner or a dissatisfied learner, not the forms we are striving for in MOOC environments.

The goal of this research is to addresses these limitations by investigating if embedding both user queries and course content in a common semantic space can produce useful and personalized retrieval results. We use a Multi-Head Bi-Encoder architecture that represents multiple semantic dimensions of the course metadata (e.g., skills, pricing, ratings, depth of subject) to investigate this problem.

To assist us with the investigation, we ask the following research questions:

- Will the use of semantic vector representations help the mapping of user queries to course recommendations in MOOCs?
- Will using a multi-head attention mechanism facilitate the model by allowing it to represent and weight different semantic features (skills, pricing, ratings, topics) better?

To address these questions, we propose a system based on two primary innovations:

- ✓ Bi-Encoder Architecture: bi-encoder architecture utilizes two independent encoders that encode user queries and course information as dense vector embeddings. The use of embeddings ensures that similarity can be computed across the semantic space quickly and easily since embeddings can be calculated in parallel.
- ✓ Multi-Head Attention: The structure of the course encoder includes the use of multi-head attention, which uses multiple attention heads which are assigned to different semantic attributes of the course information. The model uses attention to sequence through the attributes simultaneously, learning to retrieve courses that trade-off not only content relevance, but aspects that correspond to the users learning context.

This section provides a basis for proposing a solution by clarifying the main problem and the objectives that will inform the design and assessment of our semantic retrieval framework.

4.6.2. Overview of the Proposed Multi-Head Bi-encoder Model

The proposed model has two modules as illustrated in Figure 64:

- **Bi-Encoder Architecture:** This module includes two transformer-based encoders which operate in parallel but independently.
- **The Query Encoder:** encodes the user input (search string or question) into a semantic vector. The Course Encoder takes in the concatenated course metadata (title, description, category, and skill tags) and projects it into the same embedding space.

The similarities between respective course vectors and query vectors are calculated using a learned distance metric (for example, cosine similarity) which allows for quick semantic matching.

Multi-Head Attention: This module resides in the Course Encoder. Each attention head (in heads) considers different aspects of meaning (e.g., price, ratings, skills). The outputs of the attention heads are concatenated and then passed through a projection layer to a singular course embedding. This permits the model to have a broader and deeper encoding of the course content.

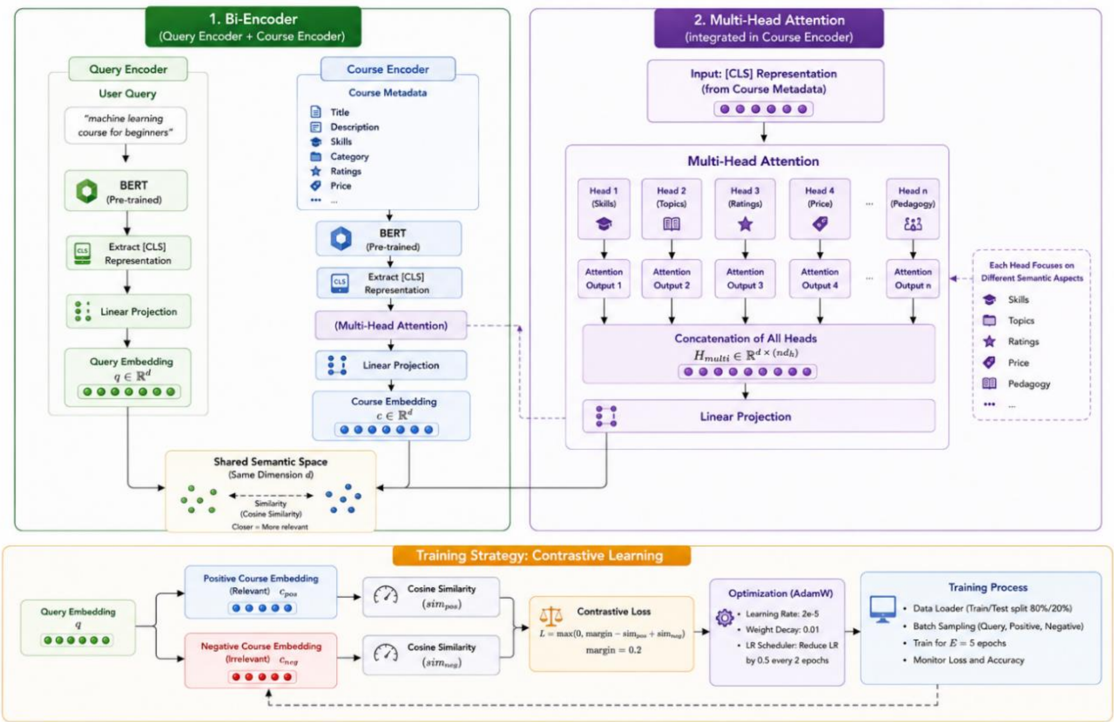


Figure 64: The architecture of our framework

4.6.3. Dataset Description and Preprocessing

To develop the proposed semantic search framework, this research takes advantage of a real dataset that consists of metadata and user generated content from the Udemy platform. The dataset is primarily based on filtered Spanish language subset obtained from two original files, Course_info.csv, which has metadata describing over 200,000 courses, and Comments.csv, which has led to more than 9 million learner's self-reviews. This research has rich raw material to extract the relevant course features and user perspectives on these few staffed course learning pathways.

The data preprocessing was done in a disciplined fashion to follow established practices of ensuring quality and relevance. The first data preprocessing steps required eliminating duplicate records, followed by data cleaning and normalizing text fields, and standardizing time and date formats. Next, to remove potential noise, the fields that did not support semantic matching-course instructor details, URLs and course pricing - were deleted. Categorical features were then normalized so that it could support later downstream tasks.

Considerable care was taken to filter the dataset appropriately to incorporate only Spanish language courses. This language filter fulfilled the methodology goal of building and assessing a language specific course recommendation system. In the end, the cleaned dataset with structured and text information included: course title, headline, category, subcategory, topic, and language. Course title, headline, category, subcategory, and topic were then combined to make composite text columns for embedding.

This cleaned dataset was subsequently used to train a Multi-Headed Bi-Encoder model. One encoder was used to pass user queries, and the other encoder encodes or transforms the course content in the same semantic embedding space. The multi-heads allow the model to learn different semantic features or properties to characterize different aspects of courses - skills, subject, and course domain - permitting a higher level of retrieval accuracy and contextual alignment between user intent and course offerings in the MOOC environment.

4.6.4. Bi-Encoder Architecture and Multi-Head Design

The Multi-Head Attention mechanism is the foundation of the Bi-Encoder system, and it is the component that enables Bi-Encoders to embrace more rich multi-dimensional relevance signals between user queries and course content. Each head can specialize in a different semantic aspect (e.g., skills, topic clusters, difficulty, or student preferences) allowing the model to learn more rich contextual representations. This design leads to better semantic expressiveness and allows for greater generalization across course-query pairs. The simultaneous nature of the attention heads ensures that the computational efficiency will be faster, which is an important trait for real-time retrieval in large-scale MOOC platforms.

4.6.4.1. Bi-Encoder Architecture

The proposed Bi-Encoder architecture aims to map user queries and course representations into a common semantic embedding space using two independent transformer-based encoders: one for encoding queries and one for course metadata.

- **Query Encoder**

The query encoder encodes the input query text Q (e.g course title) into a dense, fixed size embedding vector $q \in R^d$. The query is passed through a pretrained transformer model and passed through a linear projection layer:

$$q = Linear(Transformer(Q)) \quad \text{Equation 28}$$

- Q: Input query text.
- Transformer(.): A pretrained transformer encoder (e.g., BERT) that will produce contextual token embeddings.
- Linear(.): Fully connected layer mapping the [CLS] token embedding into a dimensional space.
- q ∈ R^d: The resulting semantic representation of the query.

- **Course Encoder**

The Course Encoder generates a contextualized representation of the course material. It leverages concatenated course metadata (e.g., title, subtitle, category), and multi-head attention to improve representation. The encoding process is:

- **Transformer Encoding:** In the first step of encoding, the course input is passed through a pretrained transformer for extracting contextual token embeddings. The [CLS] token is used as an aggregate representation of the input.
- **Dropout Regularization:** For regularization to decrease overfitting, a dropout is applied to the [CLS] output where $p = 0.1$.

$$\tilde{x} = Dropout(x; p = 0.1) \quad \text{Equation 29}$$

- **Multi-Head Attention and Projection:** The multi-head attention we apply will apply meaningful attention to the previous representation. The originally attended features H_{multi} are concatenated produced by the final linear projection layer to obtain the course embedding c:

$$c = Linear(H_{multi}) \quad \text{Equation 30}$$

H_{multi}: The output of the multi-head attention mechanism applied to the token representations.

$c \in R^d$:The final semantic representation of the course.

This dual-encoder architecture allows for effective semantic mapping between user queries and course descriptions at retrieval, since both representations will be embedded into comparable vector space.

4.6.4.2. Multi Head Attention Mechanism

The Multi-Head Attention Mechanism is a key characteristic of the course encoder that provides the model with the ability to learn complex semantic relationships in the course metadata. In a nutshell, this mechanism allows multiple perspectives to attend to different components of the input information simultaneously, giving richer and finer grained representations of meaning.

The multi-head attention mechanism draws inspiration from the Transformer which applies several attention computations, or "heads," in parallel onto the input and focuses on different subspaces. The output of each attention head is concatenated and subsequently linearly transformed to produce the output representation based on the context of the input.

Generally, for each attention head i, the attention computation includes three learned projections, which are the query matrix Q_i , key matrix K_i, and value matrix V_i. The attention output for head i is computed as:

$$Attention_i(Q_i, K_i, V_i) = softmax\left(\frac{Q_i {K_i}^T}{\sqrt{d_K}}\right) V_i \quad \text{Equation 31}$$

Where:

- Q_i , K_i, V_i: Projections of the input embeddings for head.
- d_K: The dimensionality of the key vectors.
- softmax: Guarantees that the sum of attention weights is equal to 1.

The outputs of all the heads are concatenated and put through a final linear projection:

$$MultiHead(Q,K,V) = concat(head_1, \dots, head_h)W^0 \quad \text{Equation 32}$$

- h : The number of attention heads
- W^0 :The output projection matrix.

For our course encoder, the multi-head attention is applied to the representations (token embeddings) created by the transformer model. This means that the transformer will be able to pay attention to what pieces of information matter more, and train multiple attention heads to center in on salient aspects of course metadata (title, description, category), while generating a final embedding.

The multi-head mechanism improves representation, because it allows the model to find those attention patterns in parallel, which is useful for important representation quality, and is necessary for flexible semantic matching in a bi-encoder retrieval system.

4.6.4.3. Training the Model

- **Contrastive Loss:**

The Bi-Encoder model is trained using Contrastive Loss, which aims to minimize the distance between embeddings of relevant query–course pairs while maximizing the distance for irrelevant pairs. This method encourages semantic alignment in the shared embedding space.

The contrastive loss function is defined as:

$$L = max(0, margin - sim_{pos} + sim_{neg}), \ margin = 0.2 \quad \text{Equation 33}$$

Where:

- $\mathrm{sim_{pos}} = \mathrm{Cosine\ Similarity}(\mathrm{q}, \mathrm{c_{positive}})$ is cosine similarity between the query embedding q and its associated positive course embedding $\mathrm{c_{positive}}$
- $\mathrm{sim_{neg}} = \mathrm{Cosine\ Similarity}(\mathrm{q}, \mathrm{c_{negative}})$is cosine similarity between the query and a negative (non-relevant) course embedding $\mathrm{c_{negative}}$
- margin is a hyperparameter that helps enforce a minimum required distance between positive and negative pairs; in this instance, that distance is set to 0.2.

Cosine similarity between two vectors is computed as:

$$Cosine\ Similarity(q,c) = \frac{q \cdot c}{\|q\|2\|c\|2} \quad \text{Equation 34}$$

Where:

- q :is the query embedding vector.
- c :is the course embedding vector.
- $\| q \| 2 \| c \| 2$ are the L2 norms of the respective embeddings.

This loss formulation helps ensure that the model learns to embed semantically similar items (queries and corresponding courses) closer to one another and dissimilar items are further apart, creating better retrieval performance in the semantic space.

4.6.4.4. Optimization (Adam Optimizer)

The Bi-Encoder model's parameters are optimized with the Adam optimizer. This method is suitable for Bi-Encoder models because it will use a learning rate in an adaptive way and can more easily handle sparse gradients. For these optimization procedures we used a learning rate $\eta = 2 \times 10^{-5}$, and weight decay $\lambda = 0.01$ to regularize the model and guard against overfitting. In every iteration, the update procedure for the parameters is:

$$\theta = \theta - \eta \cdot \nabla L + \lambda \cdot \theta \qquad \text{Equation 35}$$

Where:

- θ indicates the parameters of the model,
- ∇L indicates the gradient of the loss function for respect to the parameters,
- η indicates the learning rate,
- λ indicates the coefficient for weight decay.

Furthermore, we use a learning rate scheduler which will dynamically change the learning rate every epoch. In our case, we use a learning rate scheduler which will reduce the learning rate by a factor of 0.5 every two epochs. This will help to gradually converge to a minima and will add some stability to the adaptation of our model.

4.6.4.5. Model Training Process

- **Data Loader:** The data set will be divided into a training and testing data set (80%/20%). The Data Loader will do batching while training your model.
- **Epochs:** The model is trained for $E = 5$ epochs. The model is evaluated at each epoch, with training/test loss and accuracy recorded.

Multi-Head Bi-Encoder Pseudocode

```
Function Build_MultiHead_BiEncoder(course_model_name, query_model_name, head_dims, margin, dropout_rate):
    # Initialize the two encoders
    course_encoder ← MultiHeadCourseEncoder (course_model_name, head_dims, dropout_rate)
    query_encoder ← Query Encoder (query_model_name, sum(head_dims), dropout_rate)
    Return (course_encoder, query_encoder)
End Function
Function MultiHeadCourseEncoder (texts, num_heads, head dim, dropout_rate):
    # 1. Transformer encoding
    token_outputs ← Transformer(texts)                    # (batch, seq_len, hidden_size)
    cls_vec      ← token_outputs[:, 0, :]                 # [CLS] token
    # 2. Dropout
    x            ← Dropout (cls_vec, rate=dropout_rate)
    # 3. Multi-head attention
    heads ← []
    For i in 1..num_heads:
        Q_i, K_i, V_i ← Linear_i(x)                       # project to Q,K,V for head i
        scores_i     ← Q_i × K_i^T / sqrt(head_dim)       # scaled dot-product
        H_i          ← Softmax(scores_i) × V_i            # attention output
        heads. append(H_i)
    H_multi ← Concat(heads)                               # (batch, num_heads*head_dim)
    # 4. Final projection
    c      ← Linear(H_multi)                              # (batch, sum(head_dims))
    Return c
End Function
Function QueryEncoder(texts, output_dim, dropout_rate):
    token_outputs ← Transformer(texts)
    cls_vec      ← token_outputs[:, 0, :]
    x            ← Dropout(cls_vec, rate=dropout_rate)
```

```
    q            ← Linear(x, output_dim)                    # (batch, output_dim)
    Return q
End Function
Function BiEncoder_Forward(query_texts, pos_texts, neg_texts, encoders):
    (course_enc, query_enc) ← encoders
    q_emb   ← query_enc(query_texts)
    pos_emb ← course_enc(pos_texts)
    neg_emb ← course_enc(neg_texts)
    Return (q_emb, pos_emb, neg_emb)
End Function
Function ComputeContrastiveLoss(q_emb, pos_emb, neg_emb, margin):
    pos_sim  ← CosineSimilarity(q_emb, pos_emb)              # Eq. (6)
    neg_sim  ← CosineSimilarity(q_emb, neg_emb)
    losses   ← ReLU(margin - pos_sim + neg_sim)
    loss     ← Mean(losses)
    accuracy ← Sum(pos_sim > neg_sim) / BatchSize
    Return (loss, accuracy)
End Function
Function Train_BiEncoder(dataset, epochs, batch_size, lr, weight_decay, margin):
    encoders  ← Build_MultiHead_BiEncoder(…)
    optimizer ← AdamW(encoders.params, lr, weight_decay)
    scheduler ← StepLR(optimizer, step_size=2, gamma=0.5)
    dataloader ← DataLoader(dataset, batch_size, shuffle=True)
    For epoch in 1..epochs:
        total_loss ← 0
        total_acc  ← 0
        For batch in dataloader:
            q_texts, pos_texts, neg_texts ← batch
            q_emb, pos_emb, neg_emb     ← BiEncoder_Forward(q_texts, pos_texts, neg_texts, encoders)
```

```
            loss, acc                    ← ComputeContrastiveLoss(q_emb, pos_emb, neg_emb, margin)
            optimizer.zero_grad()
            loss.backward()
            ClipGradients(encoders.params, max_norm=1.0)
            optimizer.step()
            total_loss ← total_loss + loss
            total_acc  ← total_acc  + acc
        scheduler.step()
        Print «Epoch", epoch, "Loss", total_loss/NumBatches, "Acc", total_acc/NumBatches)
    Return encoders
End Function
```

4.6.5. Experimental Setup

This section explains the experimental setup used to train and test the Multi-Head Bi-Encoder architecture proposed to perform semantic matching between user queries and MOOC courses. This section describes dataset preprocessing, model architecture, training parameters, and evaluation metrics.

4.6.5.1. Dataset and Preprocessing

The dataset used for training consists of Spanish-language user reviews, course titles and user ratings. The data was collected from a filtered CSV file that contained the following fields:

- comment: User review or query.
- title: Course title.
- rate: User-provided rating (on a scale of 1–5).

4.6.5.2. Preprocessing steps:

- Label Binarization: A binary label was created, where label = 1 if rate ≥ 4, and label = 0 otherwise.

- Renaming: The columns were renamed to query and course to reflect their roles in the semantic matching task.

- Cleaning: Rows with missing values were dropped to ensure training quality.

- Embedding: Both queries and course titles were embedded using the pretrained Sentence Transformer model all-MiniLM-L6-v2, which maps textual inputs into 384-dimensional dense vectors.

4.6.5.3. Model Architecture

The proposed model uses a Bi-Encoder architecture, where queries and courses are encoded separately and are concatenated for binary relevance classification:

- Input: Pre-computed dense vectors of course titles and queries.
- Concatenation Layer: Concatenate course and query embeddings to produce a 768-dimensions vector when concatenated.
- Fully Connected Layers: A feed-forward network projects those concatenated embeddings through two dense layers:

  Linear (768 → 128) with ReLU activation.

  Linear (128 → 1) with Sigmoid activation for binary classification.

This architecture allows the model to learn non-linear relationships between user intents and course metadata as shown Table 12.

Table 12: Hyperparameter Configuration Multi-Head Bi-Encoder

Hyperparameter	**Value**
Embedding Dimension	384
Hidden Units (FC)	128
Batch Size	64
Learning Rate (η)	1e-4
Epochs	5
Optimizer	Adam
Loss Function	Binary Cross-Entropy (BCE)

4.6.5.4. Implementation Details

The whole pipeline was developed within the realm of PyTorch. It was decided to split the dataset using an 80%/20% train-test split. The data was encapsulated in custom Dataset and

DataLoader objects to facilitate batched processing. The model was trained on a GPU (where applicable), executing a series of training phases, including the following:

- Forward passing the batched embeddings.
- Computing the loss function with the use of BCE Loss.
- Calculating the backward pass, using the Adam optimizer.
- Tracking the loss and accuracy at each epoch for training and test phase.

4.6.5.5. Evaluation Metrics

The model was quantitatively evaluated at each epoch using the following metrics:

- ✓ **Train/Test Loss:** The mean binary Cross-Entropy per batch.
- ✓ **Train/Test Accuracy:** the ratio of correct predictions to total predictions.
- ✓ **Precision:** the ratio of relevant courses to total retrieved.
- ✓ **Recall:** The proportion of relevant courses retrieved among all relevant courses.
- ✓ **F1-Score:** The harmonic means of precision and recall.

4.6.6. Results

To rigorously evaluate the proposed Multi-Head Attention Bi-Encoder for semantic retrieval in MOOC platforms, we wanted to comprehensively compare it against a number of baseline architectures. The baseline models are traditional deep learning models such as CNNs and LSTMs, and variations of Bi-Encoders, both with or without pre-trained NLP embeddings, or in hybrid configurations.

The models were evaluated according to common evaluation measures, including train and test loss, accuracy, precision, recall and F1-score. Collectively, these metrics indicate how well explicitly each model learns to discriminate between relevant and irrelevant query-course pairs based on their semantic representations.

4.6.6.1. Performance Overview

Table 13 , Figure 65 ,and Figure 66 summarize the evaluation results, in which our Multi-Head Attention Bi-Encoder always beats at least one of the baselines based on the key metrics. Specifically, it achieved:

✓ The highest test accuracy of 91.30%.
✓ The lowest test loss of 0.2252.
✓ The best F1 score of 0.9533.

The performance improvement can be traced back to the attention mechanism of the model, which allows several attention heads to attend to different semantic facets (skills, keywords, use, context, etc.), thereby improving the vector representations of queries and a course, creating a more effective semantic alignment.

Table 13: Evaluation Metrics for Competing Models.

Metrics / **Approach**	**Train Loss**	**Train Accuracy**	**Test Loss**	**Test Accuracy**	**Precision**	**Recall**	**F1-score**
CNN-Based Bi-Encoder	0.3320	89.69%	0.3296	89.79%	0.8979	0.9987	0.9462
CNN	0.0897	89.72%	0.0913	89.68%	0.9001	0.9755	0.9454
LSTM-Based Bi-Encoder	0.0674	91.18%	0.0675	91.14%	0.9171	0.9910	0.9426
Bi-Encoder Based NLP Classifier	0.2368	90.81%	0.2290	90.95%	0.9158	0.9884	0.9507
Bi-Encoder	0.2632	89.91%	0.2548	90.06%	0.9319	0.9979	0.9475
Our Algorithm **Multi-Head Attention Bi-Encoder**	**0.2332**	**90.95%**	**0.2252**	**91.30%**	**0.9204**	**0.9886**	**0.9533**

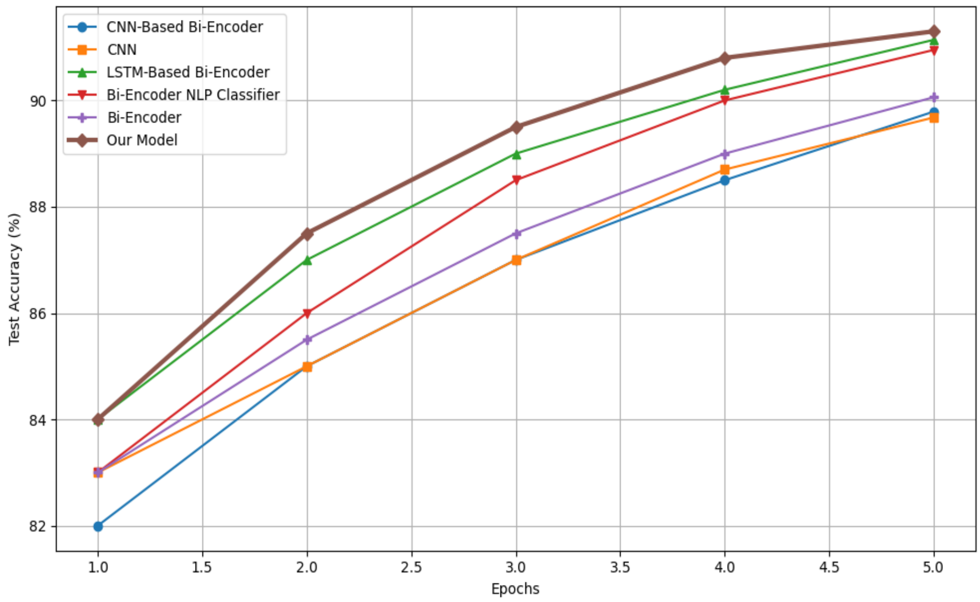


Figure 65: Test Accuracy Comparison of Semantic Search Models

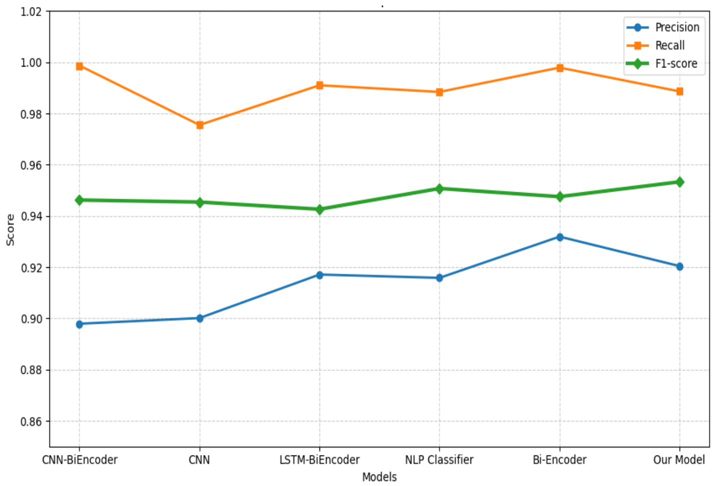


Figure 66: Precision, Recall, and F1-score Comparison of Semantic Search Models

4.6.6.2. Model Comparisons

- **CNN-Based Bi-Encoder vs Proposed Model**

The CNN-based Bi-Encoder model reached a competitive 89.79% accuracy. The Multi-Head Attention version of the model improved the Bi-Encoder to an accuracy of 89.79% + 1.51% improvement. The test loss improved from 0.3296 to 0.2252 and showed greater generalization (Figure 67). The F1 score improved as well, mainly due to the improvement in precision (0.8979 →0.9204). The attention systems, therefore, add value about alignment at the semantic level.

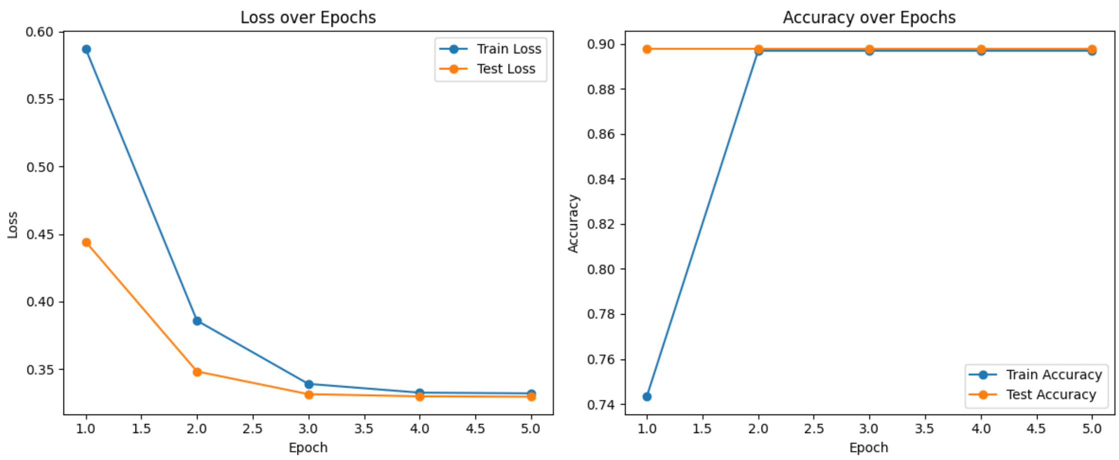


Figure 67: Training and testing accuracy and loss curves using CNN-Based Bi-Encoder.

- **CNN vs Proposed Model**

While an isolated CNN produced a lower test loss of 0.0913, the semantic matching characteristics were weaker, with the test accuracy lower at 89.68% and F1 Score of 0.9454(F

Figure 68). The CNN was limited in its understanding with regards to subtle semantic context, where nuance and understanding were not available to explore beyond a most basic level. The attention-based model was able to build deeper relational meaning between coffins.

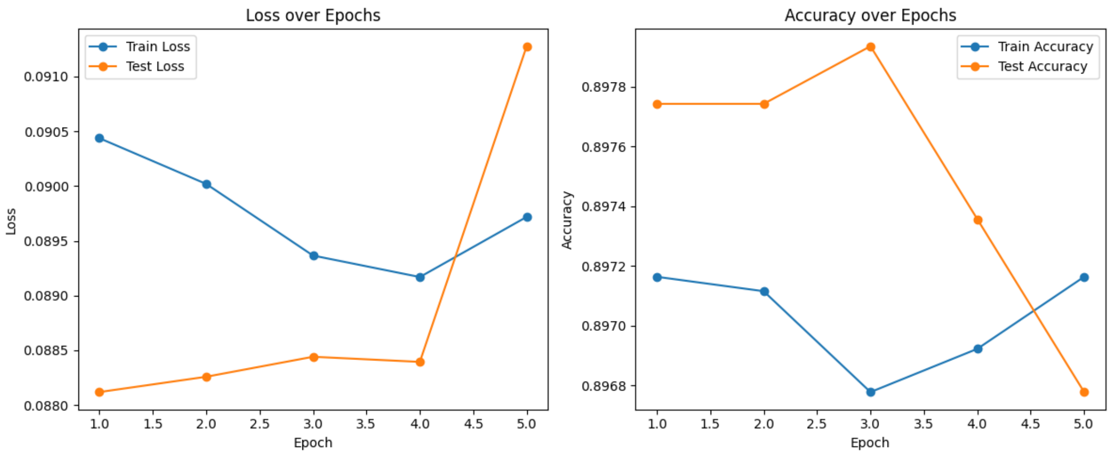


Figure 68: Training and testing accuracy and loss curves using CNN.

- **LSTM-Based Bi-Encoder vs. Proposed Model**

While the LSTMs produced strong results (91.14% accuracy, 0.9426 F1), the Multi-Head Bi-Encoder performed better than the LSTMs and allowed for more computational efficiency as well (Figure 69). The LSTM's sequential processing generally precludes it from being parallelized, while our model utilizes multiple parallel attention heads to perform semantic matching more quickly and at scale.

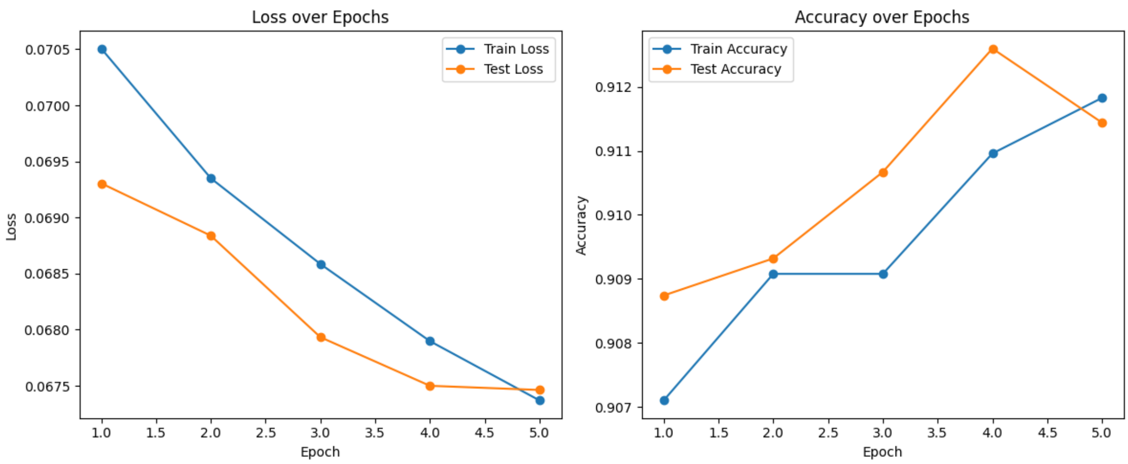


Figure 69: Training and testing accuracy and loss curves using LSTM Bi-Encoder.

- **Bi-Encoder NLP Classifier vs. Proposed Model**

Despite getting close (90.95% accuracy and 0.9507 F1), the NLP-enhanced Bi-Encoder did not possess the idea of dynamic attention, which allows it to focus on semantically rich tokens, allow for more accurate matching and fewer non-relevant retrievals (Figure 70).

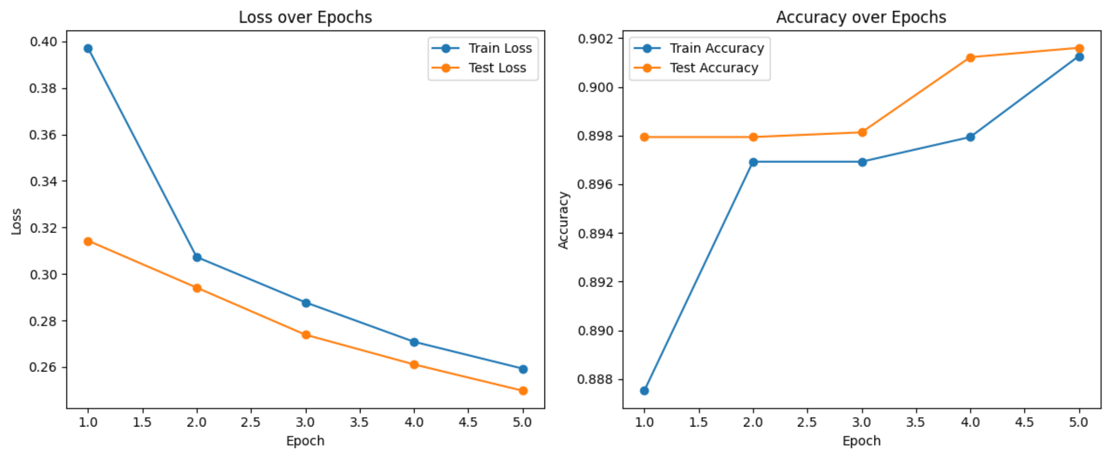


Figure 70: Training and testing accuracy and loss curves using Bi-Encoder Based NLP Classifier.

- **Basic Bi-Encoder vs Proposed Model**

The basic Bi-Encoder offered better precision (0.9319 vs 0.9204) but lower recall and F1-score. However, it was better balanced for precision and recall, and this indicates it has better matching coverage - necessary for a retrieval system designed to minimize false negatives (Figure 71).

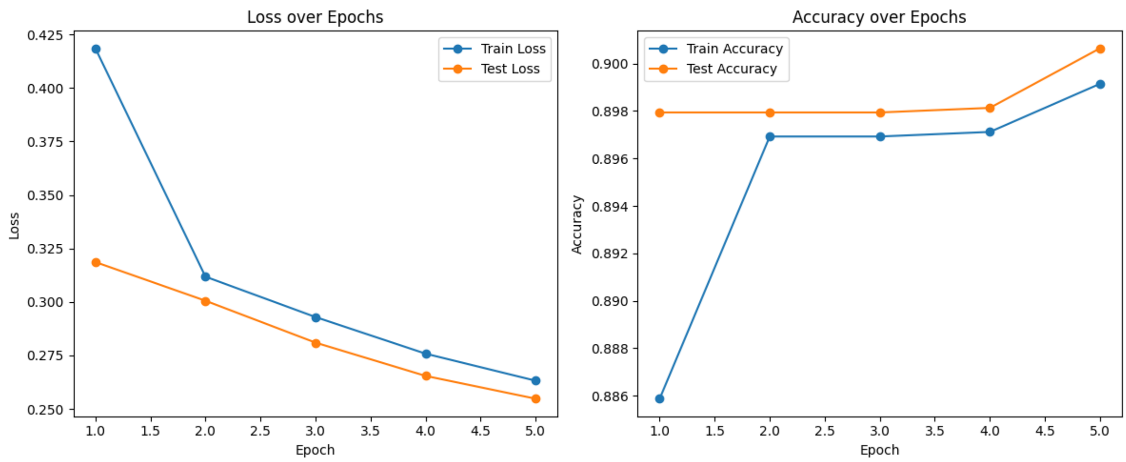


Figure 71: Training and testing accuracy and loss curves using Bi-Encoder.

The Multi-Head Attention Bi-Encoder was the best model of any test. The improved attention architecture allowed the model to include multiple semantic feature dimensions improving its precision, recall, and overall quality of the matching. Because the Multi-Head Attention Bi-Encoder outperformed the others, it had relevance to large teaching and learning spaces, where knowing a wide variety of user intents and course contexts are required for improved search experience based on intelligent personalization.

4.6.7. Discussion

The comparative assessment shows that the new Multi-Head Attention Bi-Encoder model is an effective approach to semantic search in a MOOC that showed a clear outperformance over classical and hybrid retrieval approaches along all topic standard metrics: test accuracy, loss, precision, recall, and F1-score. A unique advantage of this model is its ability to attend to multiple semantic aspects of relevance simultaneously. Using multi-head attention, the model captures varied dimensions of student intent (skills, topics, price, ratings, and keywords) from the MOOC course data that may be encoded.

- The use of a CNN-based Bi-Encoder also produced good results, but the system was unable to capture abstract semantic relationships, limiting its usefulness in modeling rich abstractions that required for precise query-course matching.
- The LSTM-based Bi-Encoder had a similar level of accuracy as our model, and it is important to note that its sequential computation structure inhibits scalability and real-time use. In contrast, our attention-mechanism promotes efficient representation learning, as well as greater parallelization and would likely lead to faster inference in educational contexts that change quickly.
- Furthermore, a class of baseline Bi-Encoder classifier using static pre-trained embeddings does not have an adaptive mechanism in place to emphasize semantically important features. In this regard, existing static approaches are forever contextually ignorant and can never distinguish between relevant query-course pairs from semantically irrelevant ones, solely because they lack a dynamic attention mechanism. Our attention-based architecture relies on these contextual properties to directly inform our attention during encoding, which is how our architecture is better suited to return accurate yet contextually salient and relevant retrieval results on each educational decision.
- While the Bi-Encoder baseline had a still relatively high precision, it suffered dramatically in recalling and F1-score. This suggests that it often misses semantically related matches outside of the exact, superficial matches. The Multi-Head Bi-Encoder correctly identified both specific and greater semantic associations, which is important in educational retrieval contexts, because learner needs are so different and with varying overlap.

In addition, it's worth noting that our model had the lowest test loss of all of the approaches evaluated. This result indicates a good marriage of model expressiveness and generalization, and a reduction in overfilling - a critical capability in any practical use of a retrieval model in a MOOC platform where user queries and course content may vary broadly in form and intent.

4.6.8. Study Constraints and Prospective Research Paths

While the results from the proposed multi-head attention Bi-Encoder architecture are promising, this study highlights some limitations and paths for future development.

One limitation is interpretability surrounding attention (i.e., while multi-head attention yields better semantic rich and context relevant encoding, we do not understand what each head is responsible for). Future work could interpret how the individual attention heads act to further illuminate what components of the course or query content were privileged.

Additionally, this framework only considered Bi-Encoder based retrieval and not subsequent reranking. Therefore, by combining a cross-encoder based reranking approach or using user interactions with the course content, we could better personalize and improve retrieval quality. A hybrid approach could allow for more contextually adaptive learning paths to exist for individual users.

Given these analyses, the current study evaluates retrieval performance as evaluated under normal conditions. A broader investigation of all types of user queries, such as goal-driven query searches, exploration browsing, or casual navigation, could produce richer data for exploring model robustness and user experience in varying user scenarios. Overall, the adaptive and modular design of architecture provides a good basis for future work, including domain-specific adaptations and refinements, incorporating ontological knowledge, and extending to multilingual and cross-domain educational platforms.

In conclusion This study has provided a thorough examination of how to enhance semantic retrieval in MOOC platforms, building a Multi-Head Attention Bi-Encoder architecture. The core aim for improving semantic retrieval was to enhance how users' queries corresponded to course metadata in the MOOC space by incorporating richer semantics, context sensitivity, and generalization tasks. To implement the above, we developed a unique deep-learned emulsion for our proposed architecture in which we performed BERT-based embeddings with BiLSTM encoders and a Multi-Head Attention mechanism. The unique feature of this architecture is that the model has enough parameters to learn complex semantic patterns and contexts in course descriptions and user queries. Furthermore, we employed the Adam optimizer, and our pipeline was implemented through training phases which included tokenization, embedding creation and semantic matching strategies. The results of our experimental evaluation showed that Multi-Head Attention Bi-Encoder model was the only architecture that enhanced performance across all classical models, such as CNN's, LSTMs, and the standard Bi-Encoders, using evaluation metrics of accuracy, F1-score and loss. Additionally, the functionality of the blocks within our architecture were modular and parallelizable, which would add considerable benefits in terms of scalability and real-world MOOC implementation. Moreover, the framework presented creates the potential of more user-centered learning systems that can retrieve items that align with individual learner goals and preferences. Its extensibility allows for potential future developments, such as attention mechanisms that are domain-specific or the ability to include knowledge graphs from other resources to support stronger semantic matching. Because it has been discussed above, this research recognizes some limitations with model interpretability and require further personalization based on user interaction data. Future work should discuss cross-encoder reranking approaches, investigate attention interpretability, and assess its application across the query types for supporting robustness of outcomes.

4.7. Summary and Comparative Analysis of Contributions

4.7.1. Comparative overview of all four approaches.

To answer the important challenges in the ecosystem of MOOCs, four models from deep learning postures and hybrid AI models were proposed, implemented, and evaluated. Each model addresses a specific problem domain course classification, personalized course recommendation, learner behavior prediction, and semantic retrieval as presents in Table 14. Below is a summary of the technical design and working idea of each of the model:

- CNN-LOA for Course Classification: this proposed hybrid model consists of a Convolutional Neural Network (CNN) and Lion Optimization Algorithm (LOA) as a model for adaptive hyperparameter tuning. This model classifies MOOC courses based on a fixed set of categories and learns hierarchical representations of the courses using a mixed representation of course titles and descriptions. It exhibits equal or improved accuracy and robustness when compared to traditional CNN and LSTM baselines.
- HMHA-C-BERT for Personalized Course Recommendation System: The HMHA-C-BERT model incorporates both BERT embeddings with a BiLSTM encoder, and features multi-head attention layer, to generate rich context representations of courses. We fuse the contextual representations with user-course interaction embeddings in order

to provide personalized courses to the learners, semantically grounded to course recommendations.

- TSA-GRU for Learner Behavior Prediction with Temporal Modeling: The model devised utilizes a temporal self-attention (TSA) mechanism, as well as Gate Recurrent Units (GRU) to model learner behaviours over time. TSA-GRU considers temporal trends and sequential dependencies to model the learner's performance, while producing better predictions compared to classical models such as BKT and PFA.
- Multi-Head Bi-Encoder for Semantic Search and Classification in MOOCs: Using the multi-head attention of a Bi-Encoder architecture, the model performs semantic similarity tasks for classification and information retrieval. The model first encodes course content and search queries into a high-dimensional vector space, and then the semantic classification and contextual semantic search occur by mapping into the semantic vector space.

Table 14: Performance Summary of Contributing Models

Contribution	**Best Performing Model**	**Task**	**Architecture**	**Test Accuracy (%)**	**Precision**	**Recall**	**F1-Score**	**Key Insights**
CNN with Lion Optimization	CNN + LOA	MOOC Course Classification	CNN with hyperparameter tuning using Lion Optimization Algorithm	**96.17**	—	—	—	Highest accuracy among all CNN + optimizer models, outperforming WOA, ACO, PSO
HMHA-C-BERT for Recommendation	HMHA-CC / BERT	Personalized Course Recommendation	BERT embeddings + Hybrid Multi-Head Attention + Contextual Compression	**97.48**	—	—	—	BERT-based HMHA outperforms CNN/LSTM/NCF variants in both training and testing
TSA-GRU for Learner Modeling	TSA-GRU	Learner Behavior Prediction	Temporal Sequential Attention + GRU	**95.60**	—	—	—	Significantly outperforms MSA-GRU, BKT, and PFA on temporal learning tasks
Multi-Head Bi-Encoder for Retrieval	Multi-Head Attention Bi-Encoder	Semantic Course Retrieval	Bi-Encoder with CNN encoder + Multi-Head Attention	**91.30**	0.9204	0.9886	**0.9533**	Strong semantic alignment and generalization best F1 among retrieval models

4.7.2. Discussion

Four new models are proposed in this work, each addressing a unique challenge in the MOOC landscape. The models are designed to align with the complexity and characteristics of the data as a function of their tasks:

- CNN-LOA focuses on course classification by capturing the structural aspects of course titles and descriptions. Its synergy of Convolutional Neural Networks (CNNs) with the Lion Optimization Algorithm (LOA) allows for efficient hyperparameter tuning and a local feature extraction.
- HMHA-C-BERT focuses on personalized course recommendation by integrating text with information on the user based on what they have engaged with using BERT embeddings, BiLSTM, and multi-head attention in a single model for a more personalized, semantically rich recommendation system.
- TSA-GRU focuses on learner behaviour to model performance. The model takes user streaming activity data of 'micro-learning events' by capturing sequentially and temporal changes in data. It uses self-attention with temporal concepts layered on GRUs to allow for timely ad-hoc interventions.
- Multi-Head Bi-Encoder focuses on retrieving semantic content, focused on improving context-aware matching of queries with course information, using two encoders and an attention-based representation.

From an architecture standpoint, these models range from simple to the very complex. Both CNN-LOA and TSA-GRU are deep, but they differ in their structures. CNN-LOA uses spatial feature extraction for course recommendations, whereas TSA-GRU uses sequential data to focus on temporal elements. The HMHA-C-BERT and Multi-Head Bi-Encoder are more complex, as they are hybrid models and incorporate transformers, attentive layers, and bi-directional encoders to capture richer semantics and model user intent

From a data modality perspective, CNN-LOA and Multi-Head Bi-Encoder use primarily unstructured text (e.g., course title, description, and query) and TSA-GRU works with structured learner interaction logs. The HMHA-C-BERT is a hybrid model that uses both data structures, taking more advanced preprocessing and feature fusion to better figure out how to form holistic representations of users and courses.

All the models perform quite well in their specific domain. CNN-LOA showed high classification accuracy (>85%) and precision. HMHA-C-BERT achieved better precision and F1-score in recall, than both group-based and user and course based traditional recommenders, although it was more computationally expensive. TSA-GRU was better than a few competing baselines, like BKT and MSA-GRU, in predicting temporal elements of studying a metacognitively reified context. The Multi-Head Bi-Encoder increased retrieval accuracy and F1-score over standard and long-personal training-based retrieval, respectively.

In terms of scalability and efficiency, CNN-LOA and TSA-GRU are comparably light algorithms, and they can handle large datasets efficiently. HMHA-C-BERT and Multi-Head Bi-Encoder are more resource demanding when training these models. However, they both demonstrated a solid performance and maximum personalization features, where the Bi-Encoder structure aids with quick inference times, which is very important for real-time retrieval systems.

Table 15 shows that the values and trade-off of the proposed models are consistent with the proposal of supporting intelligence distance learning, which provides distinctively different support for education including course classification for accuracy; personalized recommendations to learners; dynamic learner modeling; and semantic retrieval. While some models, like CNN-LOA and TSA-GRU, lift scalability in their focus as compared to the other models like HMHA-C-BERT and the Multi-Head Bi-Encoder, lift personalisation and contextualization. Overall, the proposed models reflect a complimentary combination of AI-managed components to allow an enhancement of a few different outcomes on a modern MOOC platform.

Table 15: Qualitative Comparison of Proposed Models

Model	Primary Strengths	Key Advantages	Limitations	Accuracy	Personalization	Scalability	Interpretability
CNN-LOA	Static course classification at Scale	Efficient, optimized classification via CNN and LOA	Less suited for dynamic or sequential learning content	✓		✓	✓
HMHA-C-BERT	Personalized course Recommendation	Semantic-rich, attention-based recommendation pipeline	High resource and preprocessing cost	✓	✓		

TSA-GRU	Learner behavior modeling and performance prediction	Temporal attention on séquences. Enables adaptive intervention	Requires sequential learner activity logs		✓	✓	✓
Multi-Head Bi-Encoder	Semantic content-Query Matching	Dual encoder with multi-head attention. Fast and effective for semantic search	Complex architecture and training requirements	✓	✓	✓	✓

4.7.3. How These Models Contribute to Intelligent Distance Learning

While Section 6.1 has summarized the technical advantages of each model, their broader contributions are each in respect to important pillars of intelligent distance learning: classification, personalization, prediction, and semantic navigation.

- ✓ CNN-LOA for Course Classification: To help automate the organization of vast MOOC repositories, it will categorize courses based on content. The letter of agreement will allow us to make timely tuning and continuous improvements to a real-time usable Course Categorization Model, which will best allow to filter recommendations for subsequent downstream recommendations.
- ✓ HMHA-C-BERT for Personalize Recommendation: By combining semantic content analysis and understanding an individual's user engagement histories into a robust multi-modal learner model, HMHA-C-BERT produces personalized course recommendations with high effectiveness while increasing learner satisfaction, learner engagement, and learning retention rates.
- ✓ TSA-GRU for Learner Modelling and Prediction: The temporal modelling of learners' activities enables predictions on learner engagement and performance which provides early warning, adaptive feedback to learners, and even dropout prevention.
- ✓ Multi-Head Bi-Encoder for Semantic Retrieval: This allows for improved search and discovery by matching vague user queries to contextually weighted resources which improves the user's navigation and access to more meaningful learning content.

Together our models provide a complete, AI enhanced ecosystem for contemporary distance education. They help to contribute scalability, adaptability, and a learner-centered approach toward the future of MOOCs, creating a multi-layered infrastructure for intelligent decision-making and personalized learning at scale.

4.8. Conclusion

In this chapter, we have outlined the principal methodological contributions of this thesis, with each method addressing a distinct and important problem in intelligent distance learning through MOOC platforms. First, with the course classification problem addressed with the CNN-LOA model based on deep convolutional networks with the Lion Optimization Algorithm to improve classification performance and efficiency of computational weight. The model has learnt to automatically tune hyperparameters and generate semantically rich features from the course content. Second, we introduced the HMHA-C-BERT model to address the problem of personalized course recommendation. Integrating BERT based embeddings, BiLSTM, multi-head attention and contextual compression, this hybrid model captures the difficult interaction of learner representation and course semantics. The design of the model achieves fine-grained personalized recommendations, which is a main requirement for large scalable educational platforms. Third, we developed the TSA-GRU model as a novel temporal attention-based recurrent model to model and predict learner behavior in time. This is important to model sequence data. Sequencing data is important in education because learning behaviours can change and disrupt learning. This framework enables the early identification of lack of engagement and change in performance. The capacity of the model to assign weight to previous interactions over time can also be interpreted by educators for intervention purposes to support learners. Fourth, the Multi-Head Bi-Encoder model for semantic course retrieval was developed, allowing users to retrieve contextually relevant courses given a natural language query. This model acts as a bridge between relevant user intent and relevant content semantics in order to improve relevance and support exploratory behaviors in the digital learning context.

All these contributions were subject to rigorous evaluation approaches, whether through cross-validation, tolerance-based accuracy, or statistical significance tests. Thus, the models not only achieved strong empirical performance but were flexible and adaptable for evolving large-scale online education systems. Altogether, the methods detailed in this chapter represent a unified and scalable approach to intelligence enhancement for MOOC contexts. Usage of deep learning architectures, attention mechanisms, optimization strategies, this work was presented here advanced the state of the art in a variety of areas including automating course classification, personalization, learner modeling, and semantic retrieval. These contributions addressed specific technical challenges while providing a foundation for more adaptive, equitable and efficient intelligent tutoring systems.

More widely the work outlined in this book demonstrates the possibilities afforded using AI-enabled methods in distance learning contexts. By merging technological advancement and pedagogical considerations, smart models can extend the reach of access and promote learning, leading to more inclusive, impactful and tailored learning experiences for individuals in large numbers. Subsequent chapters will build on the empirical rigour of these models and provide reflections on their potential effectiveness, limitations and approaches to use in future as part of HCI-enabled technology for education.

General Conclusion

Massive Open Online Courses (MOOCs) have surfaced through the last decade as part of an overhaul to higher education and lifelong learning. By providing free and open access to diverse courses on a plethora of topics from established educational institutions, MOOCs are dictating new methods of how knowledge is disseminated and absorbed across the globe. Even though MOOCs are made available as a democratizing tool, still, MOOCs are encountering challenges that negatively impact their affordances and their success. These challenges range from not providing personalized content delivery, limited recommendations of relevant courses, lack of user learning behavior comprehension, to ineffective semantic search systems. These challenges encapsulate this thesis and have developed the methodological framework based upon AI (artificial intelligence) and deep learning approaches that reflect the MOOC context.

The primary contribution of this doctoral research in the design, development, and evaluation of multiple intelligent deep learning models, each aimed at tackling a specific task of a MOOC platform. Collectively, these deep learning models embody the establishment of a modular solution for automatic content classification, personalized recommendations, learner behavior modeling, and context-sensitive semantic retrieval. While each deep learning model, is conceived and utilized as a standalone approach to solve the one specific intelligent learnable task, they all incorporate the same overall architecture: a set of intelligent, scalable, interpretable, and adaptable systems for online education. The leading model discussed in the thesis, is a CNN-LOA (Convolutional Neural Network with Lion Optimization Algorithm) model that ideally uses the strengths of CNNs capabilities of pattern recognition, and the adaptive hyper parameter tuning properties of the Lion optimizer to classify courses and their metadata (such as titles and descriptions) quickly and efficiently. Overall, the CNN-LOA model is designed specifically as a solution to the issue of content categorization within MOOC, which involves a large volume of content involving varied and diverse subject matter that require reliable automatable labeling.

To model temporal aspects of learner behavior, we developed TSA-GRU (Temporal Sparse Attention with GRU) which is a recurrent model that predicts future learning activities based on past interactions. TSA-GRU uses sparsity-aware attention, which allows for temporal noise to be filtered while focusing on the pedagogically significant parts of behavioral sequences to enable better interpretability and prediction of behavior. TSA-GRU also informs the field of student engagement analytics at a broader level by providing a better understanding of how learners engage with content in temporally structured flows. We also wanted to improve traditional unseen keyword based searching mechanisms, so we developed a Multi-Head Bi-Encoder architecture for semantic search and retrieval. The model stores course content within a dense vector space, where semantic relationships can be modeled between queried data and document. This allows learners to interact with more contextualized and relevant information and retrieve it in real-time. The overall goal of enabling learner access to relevant resources in large MOOC repositories should improve the way learners navigate and discover relevant material. The experimental evaluations of all models exhibited similar consistent and significant improvements over baseline methods. Overall, the results represent the value of combining attention-based mechanisms alongside semantic embeddings and temporal modeling to better address the complex nature of online learning contexts. Improvements in specific metrics including recall, precision, F1-score and level of robustness highlight the value of our contributions across real MOOC implementations.

Beyond the technological milestones stated, one fundamental conclusion of this thesis is that we must embrace Artificial Intelligence to achieve more equitable, inclusive, and engaging education, which is learner-centered and oriented. MOOCs represent the first wave of developing education toward democratization. Educators can now leverage learning analytics to create intelligent educational environments for analyzing, interpreting, and responding to learner behavior within the constraints of the contexts in which learning takes place. We contribute to this goal by addressing many of the key challenges expanding MOOCs and turning them into intelligent educational systems, including user heterogeneity, content overload, and the need for trust-based recommendations. Furthermore, the contributions we provide are relevant to contexts beyond MOOCs. The models and methods we propose are generalizable e-learning systems such as educational institutional Learning Management Systems (LMS), corporate training systems, and mobile learning. The same issues we address in the thesis: personalization, behavior analytics and the search for intelligent recommendations, are common to many e-learning contexts, and ultimately motivate the development of the next-generation (AI-driven) e-learning systems that can continuously adapt to learner goals, preferences, and progress. Despite these advances, we must acknowledge the limits of the present research. One major limitation is the computing cost of advanced deep learning models. Even if the hardware is available on some scaling for training and inference, the required resources may not be available in every place of education. While our models produce strong predicting performances on select tasks, we need to do more to strengthen model interpretability especially with regards to understanding how the models are developing recommendations or predictions for learners and educators. The second area ripe for future research is by taking on more rich behavioral data, such as: clickstream logs, time of task, durations of pauses, and how learners may use devices (i.e., multi-tasking). By drawing on this data, we can deepen our understanding of engagement and attendant responsibilities and customize the learning experience. We can also enrich personalization by integrating modes of emotional and motivational cues, which could provide more holistic personalization, resulting in both learner performance as well as learner satisfaction and retention.

Hence, future directions for this research can be framed in three possible ways:

- Interpretability and Explainability
- Increasing the transparency of AI decisions will create trust in systems outputs through learner understanding.
- Multimodal and Real-Time Data By utilizing different streams of interaction data from the learner, immediate model prediction and follow-up data updates will improve predictions.
- Transferability to Other Educational Contexts.
- The adaptation, implementation and evaluation of the suggested frameworks in other education contexts.
- Hybrid classrooms, corporate e-learning, peer-learning networks etc.
- will provide for more extensive applicability and scalability.

In conclusion, this thesis has offered a significant and timely contribution to the integration of AI and education technology. By considering relatively small but still viable contexts like MOOCs and extending them with adaptable and transferable models, we are forward-leaning toward smarter, human-centric, educational systems. Finds reinforced our need to establish e-learning platforms that simultaneously evolve with the learner, characterize behavior in open contexts, and promote personalized, equitable, and sustainable education for society.

References

[1] I. Boutabia, A. Benmachiche, A. A. Betouil, and C. Chemam, ‘A Survey in the Use of Deep Learning Techniques in The Open Classroom Approach’, in *2024 6th International Conference on Pattern Analysis and Intelligent Systems (PAIS)*, Apr. 2024, pp. 1–7. doi: 10.1109/PAIS62114.2024.10541268.

[2] M. Majda, S. Hassina, B. Abdelmadjid, and M. Amina, ‘Un système de recommandation hybride basé flou adapté à l’apprentissage amélioré par la technologie’, *المجلة الجزائرية للعلوم*, vol. 1, no. 2, pp. 57–64, Jun. 2018, Accessed: Jul. 07, 2026. [Online]. Available: https://asjp.cerist.dz/en/article/257319

[3] ‘PLATO – Illinois Distributed Museum’. Accessed: May 30, 2025. [Online]. Available: https://distributedmuseum.illinois.edu/exhibit/plato/

[4] M. Kara, ‘Distance education: a systems view of online learning: by Michael G. Moore and Greg Kearsley, Belmont, CA, Wadsworth Cengage Learning, 2012, 361 pp., £92.99 (paperback), ISBN: 978-1-111-52099-1’, *Educ. Rev.*, vol. 72, no. 6, pp. 800–800, Nov. 2020, doi: 10.1080/00131911.2020.1766204.

[5] M. F. Paulsen, ‘Online education systems: Discussion and definition of terms’, *NKI Distance Educ.*, vol. 202, pp. 1–8, 2002.

[6] L. Pappano, ‘The Year of the MOOC’, *N. Y. Times*, vol. 2, no. 12, p. 2012, 2012, Accessed: May 30, 2025. [Online]. Available: https://ci576.wordpress.com/wp-content/uploads/2016/12/the-year-of-the-mooc-ny-times.pdf

[7] I. Boutabia, A. Benmachiche, A. Bennour, A. A. Betouil, M. Derdour, and F. Ghabban, ‘Hybrid CNN-ViT Model for Student Engagement Detection in Open Classroom Environments’, *SN Comput. Sci.*, vol. 6, no. 6, p. 684, Jul. 2025, doi: 10.1007/s42979-025-04228-2.

[8] R. S. Baker, T. Martin, and L. M. Rossi, ‘Educational Data Mining and Learning Analytics’, in *The Handbook of Cognition and Assessment*, A. A. Rupp and J. P. Leighton, Eds, Hoboken, NJ, USA: John Wiley & Sons, Inc., 2016, pp. 379–396. doi: 10.1002/9781118956588.ch16.

[9] M. A. Chatti, A. L. Dyckhoff, U. Schroeder, and H. Thüs, ‘A reference model for learning analytics’, *Int. J. Technol. Enhanc. Learn.*, vol. 4, no. 5/6, p. 318, 2012, doi: 10.1504/IJTEL.2012.051815.

[10] S. Dhawan, ‘Online Learning: A Panacea in the Time of COVID-19 Crisis’, *J. Educ. Technol. Syst.*, vol. 49, no. 1, pp. 5–22, Sep. 2020, doi: 10.1177/0047239520934018.

[11] W. R. Watson and S. L. Watson, ‘An argument for clarity: What are learning management systems, what are they not, and what should they become?’, *TechTrends*, vol. 51, pp. 28–34, 2007, Accessed: May 30, 2025. [Online]. Available: https://hal.science/hal-00692067/

[12] D. R. Garrison, T. Anderson, and W. Archer, ‘Critical inquiry in a text-based environment: Computer conferencing in higher education’, *Internet High. Educ.*, vol. 2, no. 2–3, pp. 87–105, 1999, Accessed: May 30, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1096751600000166

[13] S. Hrastinski, ‘Asynchronous and synchronous e-learning’, *Educ. Q.*, vol. 31, no. 4, pp. 51–55, 2008, Accessed: May 30, 2025. [Online]. Available: http://elearning.fit.hcmup.edu.vn/~longld/References%20for%20TeachingMethod&EduTechnology%20-%20Tai%20lieu%20PPDH%20&%20Cong%20Nghe%20Day%20Hoc/(Book)%20-%20Sach%20tham%20khao%20-%20eLearning/e-Learning%20Concepts/Asynchronous%20&%20Synchronous%20e-Learning%20(Hrastinski-2008).pdf

[14] P. Brusilovsky and E. Millán, ‘User Models for Adaptive Hypermedia and Adaptive Educational Systems’, in *The Adaptive Web*, vol. 4321, P. Brusilovsky, A. Kobsa, and W. Nejdl, Eds, in Lecture Notes in Computer Science, vol. 4321. , Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 3–53. doi: 10.1007/978-3-540-72079-9_1.

[15] T. Anderson and J. Dron, ‘Three generations of distance education pedagogy’, *Int. Rev. Res. Open Distrib. Learn.*, vol. 12, no. 3, p. 80, Mar. 2011, doi: 10.19173/irrodl.v12i3.890.

[16] ‘Laurillard: Rethinking university teaching: A conversatio... - Google Scholar’. Accessed: May 30, 2025. [Online]. Available:

https://scholar.google.com/scholar_lookup?title=Rethinking+University+Teaching:+A+Conversational+Framework+for+the+Effective+Use+of+Learning+Technologies&author=Laurillard,+D.&publication_year=2013
[17] 'Jonassen: Activity theory as a framework for designing... - Google Scholar'. Accessed: May 30, 2025. [Online]. Available: https://scholar.google.com/scholar_lookup?title=Designing+Constructivist+Learning+Environments&author=Jonassen,+D.&publication_year=1999
[18] '"Coming Together is a Beginning; Keeping Together is Progress; Working Together is Success"……Henry Ford', *Int. J. Reg. Anaesth.*, vol. 2, no. 1, Jan. 2021, doi: 10.13107/ijra.2021.v02i01.015.
[19] 'elearnspace. Connectivism: A Learning Theory for the Digital Age'.
[21] I. Boutabia, A. Benmachiche, A. A. Betouil, M. Boutassetta, and M. Derdour, *A Survey on AI Applications in the Open Classroom Approach*. 2025, p. 7. doi: 10.1109/ICNAS68168.2025.11298090.
[22] I. Boutabia, A. Benmachiche, and A. A. Betouil, 'Learning-Based Dynamic Obstacle Avoidance: A Reinforcement Learning Framework for Autonomous Mobile Robots'.
[23] A. Benmachiche, B. Tahar, L. M. Tayeb, and Z. Asma, 'A dynamic navigation for autonomous mobiles robots', *Intell. Decis. Technol.*, vol. 10, no. 1, pp. 81–91, Jan. 2016, doi: 10.3233/IDT-150239.
[24] L. Chen, P. Chen, and Z. Lin, 'Artificial Intelligence in Education: A Review', *IEEE Access*, vol. 8, pp. 75264–75278, 2020, doi: 10.1109/ACCESS.2020.2988510.
[25] B. Oumaima, A. Benmachiche, M. Majda, M. Redjimi, and M. Derdour, *Examining Intelligent Tutoring Systems and Their AI Underpinnings in the Design of the Future of Learning*. 2025, p. 8. doi: 10.1109/ICNAS68168.2025.11298109.
[26] 'VanLehn: The relative effectiveness of human tutoring,... - Google Scholar'. Accessed: May 31, 2025. [Online]. Available: https://scholar.google.com/scholar_lookup?title=The+relative+effectiveness+of+human+tutoring,+intelligent+tutoring+systems,+and+other+tutoring+systems&author=K.+VanLehn&publication_year=2011
[27] A. Benmachiche, M. Derdour, M. S. Kahil, M. C. Ghanem, and M. Deriche, 'Adaptive hybrid PSO–APF algorithm for advanced path planning in next-generation autonomous robots', *Sensors*, vol. 25, no. 18, p. 5742, 2025, Accessed: Jul. 07, 2026. [Online]. Available: https://www.mdpi.com/1424-8220/25/18/5742
[28] A. Makhlouf, A. Benmachiche, and I. Boutabia, 'A Hybrid BFO/PSO Navigation Approach for an Autonomous Mobile Robot', *Informatica*, vol. 48, pp. 209–222, 2024, Accessed: Jan. 09, 2026. [Online]. Available: https://elibrary.ru/item.asp?id=78862648
[29] A. Benmachiche, A. A. Betouil, I. Boutabia, A. Nouari, K. Boumahni, and H. Bouzata, 'A Fuzzy Navigation Approach Using the Intelligent Lights Algorithm for an Autonomous Mobile Robot', in *12th International Conference on Information Systems and Advanced Technologies "ICISAT 2022"*, vol. 624, M. R. Laouar, V. E. Balas, B. Lejdel, S. Eom, and M. A. Boudia, Eds, in Lecture Notes in Networks and Systems, vol. 624. , Cham: Springer International Publishing, 2023, pp. 112–121. doi: 10.1007/978-3-031-25344-7_11.
[30] A. S. Aziz, R. A. El-KHORIBI, and S. A. Taie, 'ADAPTIVE E-LEARNING RECOMMENDATION MODEL BASED ON THE KNOWLEDGE LEVEL AND LEARNING', . *Vol.*, no. 22, 2021.
[31] I. Boutabia, A. Benmachiche, A. A. Betouil, M. Boutassetta, and M. Derdour, *A Survey on AI Applications in the Open Classroom Approach*. 2025, p. 7. doi: 10.1109/ICNAS68168.2025.11298090.
[32] O. Pilli and W. Admiraal, 'A Taxonomy of Massive Open Online Courses', *Contemp. Educ. Technol.*, vol. 7, no. 3, Sep. 2016, doi: 10.30935/cedtech/6174.
[33] 'Survey of Personalized Learning Software Systems: A Taxonomy of Environments, Learning Content, and User Models'. Accessed: May 31, 2025. [Online]. Available: https://www.mdpi.com/2227-7102/13/7/741

[34] 'Apprentissage évolutionnaire d'une chaîne de Markov cachée pour la reconnaissance automatique de la parole audiovisuelle | ASJP'. Accessed: Jul. 07, 2026. [Online]. Available: https://asjp.cerist.dz/en/article/257320
[35] B. Abdelmdjid, B. Tahar, M. Amina, Z. Meriem, and M. Majda, 'Cooperative Approach navigation for autonomous Mobile Robots (Ap-plication on a building Site)', *المجلة الجزائرية للعلوم*, vol. 1, no. 2, pp. 41–45, 2018, Accessed: Jul. 07, 2026. [Online]. Available: https://asjp.cerist.dz/en/article/257317
[36] B. Abdelmdjid, B. Tahar, M. Amina, and M. Majda, 'Un modèle de planification multi robots base sur les algorithmes géné-tiques pour la navigation optimale de robots autonomes', *المجلة الجزائرية للعلوم*, vol. 5, no. 10, pp. 47–50, 2020, Accessed: Jul. 07, 2026. [Online]. Available: https://asjp.cerist.dz/en/article/259816
[37] K. Spriggs, M. C. Lau, and K. Passi, 'Personalizing Education through an Adaptive LMS with Integrated LLMs', Jan. 24, 2025, *arXiv*: arXiv:2502.08655. doi: 10.48550/arXiv.2502.08655.
[38] J. Jabbour, K. Kleinbard, O. Miller, R. Haussman, and V. J. Reddi, 'SocratiQ: A Generative AI-Powered Learning Companion for Personalized Education and Broader Accessibility', Feb. 01, 2025, *arXiv*: arXiv:2502.00341. doi: 10.48550/arXiv.2502.00341.
[39] J. Yu *et al.*, 'From MOOC to MAIC: Reshaping Online Teaching and Learning through LLM-driven Agents', Sep. 05, 2024, *arXiv*: arXiv:2409.03512. doi: 10.48550/arXiv.2409.03512.
[40] A. Goel, P. Thajchayapong, V. Nandan, H. Sikka, and S. Rugaber, 'A4L: An Architecture for AI-Augmented Learning', May 08, 2025, *arXiv*: arXiv:2505.06314. doi: 10.48550/arXiv.2505.06314.
[41] B. Sedraoui, A. Benmachiche, and A. Makhlouf, *Developing and Evaluating Lightweight Cryptographic Algorithms for Secure Embedded Systems in IoT Devices*. 2026. doi: 10.48550/arXiv.2601.02981.
[42] T. R. Liyanagunawardena, K. Lundqvist, R. Mitchell, S. Warburton, and S. A. Williams, 'A MOOC Taxonomy Based on Classification Schemes of MOOCs', *Eur. J. Open Distance E-Learn.*, vol. 22, no. 1, pp. 85–103, Jul. 2019, doi: 10.2478/eurodl-2019-0006.
[43] E. P. Ododo, A. I. Anwan, I. E. Clement, E. E. Epoke, and M. Constant, 'ASSESSING SOCIAL STUDIES TEACHERS'PREPAREDNESS TO INTEGRATE AI AND E-LEARNING PLATFORMS FOR ENHANCED TEACHING AND LEARNING EXPERIENCES', *Am. J. Soc. Sci. Educ. Innov.*, vol. 6, no. 09, pp. 115–127, 2024, Accessed: Jun. 03, 2025. [Online]. Available: https://inlibrary.uz/index.php/tajssei/article/view/43554
[44] I. Karadag, O. Z. Güzelci, and S. Alaçam, 'EDU-AI: A twofold machine learning model to support classroom layout generation', *Constr. Innov.*, vol. 23, no. 4, pp. 898–914, 2023, Accessed: Jun. 03, 2025. [Online]. Available: https://www.emerald.com/insight/content/doi/10.1108/ci-02-2022-0034/full/html
[45] F. Altoe and H. S. Pinto, 'Towards a Personalized Online Fake News Taxonomy', in *Proceedings of the 31st ACM Conference on User Modeling, Adaptation and Personalization*, Limassol Cyprus: ACM, Jun. 2023, pp. 96–105. doi: 10.1145/3565472.3592963.
[46] A. T. Mujallid, 'Digital Active Learning Strategies in Blended Environments to Develop Students' Social and Emotional Learning Skills and Engagement in Higher Education', *Eur. J. Educ.*, vol. 59, no. 4, p. e12748, Dec. 2024, doi: 10.1111/ejed.12748.
[47] O. Erstad, M. Černochová, G. Knezek, T. Furuta, K. Takami, and C. Liang, 'Social and Emotional Modes of Learning Within Digital Ecosystems: Emerging Research Agendas', *Technol. Knowl. Learn.*, vol. 29, no. 4, pp. 1751–1766, Dec. 2024, doi: 10.1007/s10758-024-09775-w.
[48] N. Najjar, M. Rouphael, T. Bitar, and W. Hleihel, 'The rise and drop of online learning: adaptability and future prospects', in *Frontiers in Education*, Frontiers Media SA, 2025, p. 1522905. Accessed: Jun. 03, 2025. [Online]. Available: https://www.frontiersin.org/journals/education/articles/10.3389/feduc.2025.1522905/full
[49] A. Siedahmed *et al.*, 'E-TRIALS: Empowering Data-Driven Decisions to Enhance Computer-Based Learning Platforms', Apr. 16, 2025, *arXiv*: arXiv:2502.10545. doi: 10.48550/arXiv.2502.10545.

[50] L. J. Johnston, J. E. Griffin, I. Manolopoulou, and T. Jendoubi, 'Uncovering Student Engagement Patterns in Moodle with Interpretable Machine Learning', Dec. 16, 2024, *arXiv*: arXiv:2412.11826. doi: 10.48550/arXiv.2412.11826.
[51] 'Moodle Plugins directory: Analytics for Moodle with IntelliBoard'. Accessed: Jun. 03, 2025. [Online]. Available: https://moodle.org/plugins/local_intelliboard/versions
[52] 'Learning Analytics for Moodle - Moodle - LMS Analytics', Moodle. Accessed: Jun. 04, 2025. [Online]. Available: https://moodle.com/functionality-with-moodle/learning-analytics-for-moodle/
[53] 'xAPI and Moodle', Moodle US. Accessed: Jun. 04, 2025. [Online]. Available: https://supportus.moodle.com/support/solutions/articles/80001086547-xapi-and-moodle
[54] 'Moodle Plugins directory: Learning Analytics'. Accessed: Jun. 04, 2025. [Online]. Available: https://moodle.org/plugins/local_learning_analytics
[55] 'Home - Open edX'. Accessed: Jun. 04, 2025. [Online]. Available: https://openedx.org/
[56] 'List of Features', Open edX. Accessed: Jun. 04, 2025. [Online]. Available: https://openedx.org/the-platform/features/
[57] 'Proctortrack's easy and seamless integration with Open edx'. Accessed: Jun. 04, 2025. [Online]. Available: https://proctortrack.com/proctortrack-for-open-edx
[58] 'Exploring Open edX Quince: A Comprehensive Guide - Appsembler'. Accessed: Jun. 04, 2025. [Online]. Available: https://appsembler.com/blog/exploring-open-edx-quince-a-comprehensive-guide/
[59] 'AI Grading in Peer Reviews: Enhancing Coursera's Learning Experience with Faster, High-Quality Feedback - Coursera Blog'. Accessed: Jun. 04, 2025. [Online]. Available: https://blog.coursera.org/ai-grading-in-peer-reviews-enhancing-courseras-learning-experience-with-faster-high-quality-feedback/
[60] 'Coursera announces new AI content and innovations to help HR and learning leaders drive organizational agility amid relentless disruption - Coursera Blog'. Accessed: Jun. 04, 2025. [Online]. Available: https://blog.coursera.org/trusted-content-and-ai-innovations-to-drive-organizational-agility-for-learning-leaders/
[61] 'GenAI for Performance Reviews: Streamline Feedback Processes | Coursera'. Accessed: Jun. 04, 2025. [Online]. Available: https://www.coursera.org/learn/genai-for-performance-reviews-streamline-feedback-processes
[62] 'AI in Workforce Management', Coursera. Accessed: Jun. 04, 2025. [Online]. Available: https://www.coursera.org/learn/ai-in-workforce-management
[63] R. Kop and A. Hill, 'Connectivism: Learning theory of the future or vestige of the past?', *Int. Rev. Res. Open Distrib. Learn.*, vol. 9, no. 3, Oct. 2008, doi: 10.19173/irrodl.v9i3.523.
[64] A. Margaryan, M. Bianco, and A. Littlejohn, 'Instructional quality of Massive Open Online Courses (MOOCs)', *Comput. Educ.*, vol. 80, pp. 77–83, Jan. 2015, doi: 10.1016/j.compedu.2014.08.005.
[65] 'The Rise of MOOCs'. Accessed: Jun. 11, 2025. [Online]. Available: https://www.downes.ca/post/57911
[66] A. Fox, 'Viewpoint: From MOOCs to SPOCs', *Commun. ACM*, Jan. 2013.
[67] W. Wang, H. Yu, and C. Miao, 'Deep Model for Dropout Prediction in MOOCs', in *Proceedings of the 2nd International Conference on Crowd Science and Engineering*, in ICCSE'17. New York, NY, USA: Association for Computing Machinery, Jul. 2017, pp. 26–32. doi: 10.1145/3126973.3126990.
[68] 'Breslow: Studying learning in the worldwide classroom... - Google Scholar'. Accessed: Jun. 12, 2025. [Online]. Available: https://scholar.google.com/scholar_lookup?title=Studying+learning+in+the+worldwide+classroom:+Research+into+edX%E2%80%99s+first+MOOC&author=Breslow,+L&publication_year=2013
[69] Neha and E. Kim, 'Designing effective discussion forum in MOOCs: insights from learner perspectives', *Front. Educ.*, vol. 8, Dec. 2023, doi: 10.3389/feduc.2023.1223409.
[70] C. Piech, J. Huang, Z. Chen, C. Do, A. Ng, and D. Koller, 'Tuned Models of Peer Assessment in MOOCs', arXiv.org. Accessed: Jun. 12, 2025. [Online]. Available: https://arxiv.org/abs/1307.2579v1

[71] 'Khalil: Learning analytics: principles and constraints - Google Scholar'. Accessed: Jun. 12, 2025. [Online]. Available: https://scholar.google.com/scholar_lookup?title=Learning+Analytics:+Principles+and+Challenges&author=Khalil,+M&publication_year=2015
[72] B. Sedraoui, A. Benmachiche, A. Makhlouf, and C. Chemam, *Exploring the Integration of Differential Privacy in Cybersecurity Analytics: Balancing Data Utility and Privacy in Threat Intelligence*. 2026. doi: 10.48550/arXiv.2601.00385.
[73] D. Abbas, A. Benmachiche, M. Derdour, and B. K. Sedraoui, 'Privacy and Security in Decentralized Cyber-Physical Systems: A Survey'.
[74] S. O. Boufaida, A. Benmachiche, M. Maatallah, and C. Chemam, 'Hybrid Multi-Factor Authentication (MFA) Using Biometrics and Behavioral Analysis'.
[75] B. Oumaima, A. Benmachiche, M. Maatallah, and C. Chemam, *Hybrid Secure Routing in Mobile Ad-hoc Networks (MANETSs)*. 2026. doi: 10.48550/arXiv.2602.13204.
[76] B. Sedraoui, A. Benmachiche, A. Makhlouf, D. Abbas, and M. Derdour, *Cybersecurity in E-Learning: A Literature Review on Phishing Detection Using ML and DL Techniques*. 2025, p. 10. doi: 10.1109/ICNAS68168.2025.11298114.
[77] N. Gillani and R. Eynon, 'Communication patterns in massively open online courses', *Internet High. Educ.*, vol. 23, pp. 18–26, Oct. 2014, doi: 10.1016/j.iheduc.2014.05.004.
[78] C. Milligan, A. Littlejohn, and A. Margaryan, 'PATTERNS OF ENGAGEMENT IN CONNECTIVIST MASSIVE OPEN ONLINE COURSES', 2013.
[79] A. G. Spatioti, I. Kazanidis, and J. Pange, 'A Comparative Study of the ADDIE Instructional Design Model in Distance Education', *Information*, vol. 13, no. 9, Art. no. 9, Sep. 2022, doi: 10.3390/info13090402.
[80] M. Keramida (M.Ed.), 'The Pedagogy Behind MOOCs: What eLearning Professionals Should Know', eLearning Industry. Accessed: Jun. 26, 2025. [Online]. Available: https://elearningindustry.com/the-pedagogy-behind-moocs-what-elearning-professionals-should-know
[81] T. Storme, N. Vansieleghem, S. Devleminck, J. Masschelein, and M. Simons, 'The emerging pedagogy of MOOCs, the educational design of technology and practices of study', *J. Comput. Educ.*, vol. 3, no. 3, pp. 309–328, Sep. 2016, doi: 10.1007/s40692-016-0070-5.
[82] O. Vainas *et al.*, 'E-Gotsky: Sequencing Content using the Zone of Proximal Development', Apr. 28, 2019, *arXiv*: arXiv:1904.12268. doi: 10.48550/arXiv.1904.12268.
[83] M. D. Merrill, L. Drake, M. J. Lacy, and J. A. Pratt, 'Reclaiming Instructional Design'.
[84] S. Chimalakonda and K. V. Nori, 'A Patterns Based Approach for Design of Educational Technologies', Feb. 07, 2018, *arXiv*: arXiv:1802.02663. doi: 10.48550/arXiv.1802.02663.
[85] S. Li, J. Du, and J. Sun, 'Unfolding the learning behaviour patterns of MOOC learners with different levels of achievement', *Int. J. Educ. Technol. High. Educ.*, vol. 19, no. 1, p. 22, Dec. 2022, doi: 10.1186/s41239-022-00328-8.
[86] A. Anderson, D. Huttenlocher, J. Kleinberg, and J. Leskovec, 'Engaging with Massive Online Courses', Mar. 14, 2014, *arXiv*: arXiv:1403.3100. doi: 10.48550/arXiv.1403.3100.
[87] Z. Liu, R. Brown, and C. F. Lynch, 'MOOC Learner Behaviors by Country and Culture; an Exploratory Analysis'.
[88] R. L. Moore and J. and Shin, 'Using learner engagement patterns for predictive modeling in a MicroMasters program', *Distance Educ.*, vol. 0, no. 0, pp. 1–21, doi: 10.1080/01587919.2024.2441252.
[89] B. Poellhuber, N. Roy, and I. Bouchoucha, 'Understanding Participant's Behaviour in Massively Open Online Courses', *Int. Rev. Res. Open Distrib. Learn.*, vol. 20, no. 1, Feb. 2019, doi: 10.19173/irrodl.v20i1.3709.
[90] L. Shi, A. I. Cristea, A. M. Toda, and W. Oliveira, 'Revealing the Hidden Patterns: A Comparative Study on Profiling Subpopulations of MOOC Students', Aug. 12, 2020, *arXiv*: arXiv:2008.05850. doi: 10.48550/arXiv.2008.05850.
[91] G. Siemens and P. Long, 'Penetrating the Fog: Analytics in Learning and Education', *Educ. Rev.*, vol. 46, no. 5, p. 30, 2011.
[92] B. Sedraoui, A. Benmachiche, A. Makhlouf, and C. Chemam, *Applications of Secure Multi-Party Computation in Financial Services*. 2026. doi: 10.48550/arXiv.2601.00334.

[93] 'dgreen,+Journal+manager,+JEDMVol1Issue1_BakerYacef.pdf'.
[94] R. F. Kizilcec, C. Piech, and E. Schneider, 'Deconstructing disengagement: analyzing learner subpopulations in massive open online courses', in *Proceedings of the Third International Conference on Learning Analytics and Knowledge*, Leuven Belgium: ACM, Apr. 2013, pp. 170–179. doi: 10.1145/2460296.2460330.
[95] Z. Papamitsiou and A. A. Economides, 'Learning Analytics and Educational Data Mining in Practice: A Systematic Literature Review of Empirical Evidence', *J. Educ. Technol. Soc.*, vol. 17, no. 4, pp. 49–64, 2014, Accessed: Jun. 12, 2025. [Online]. Available: https://www.jstor.org/stable/jeductechsoci.17.4.49
[96] P. J. Guo, J. Kim, and R. Rubin, 'How video production affects student engagement: an empirical study of MOOC videos', in *Proceedings of the first ACM conference on Learning @ scale conference*, Atlanta Georgia USA: ACM, Mar. 2014, pp. 41–50. doi: 10.1145/2556325.2566239.
[97] M. Fei and D.-Y. Yeung, 'Temporal Models for Predicting Student Dropout in Massive Open Online Courses', in *2015 IEEE International Conference on Data Mining Workshop (ICDMW)*, Atlantic City, NJ, USA: IEEE, Nov. 2015, pp. 256–263. doi: 10.1109/ICDMW.2015.174.
[98] M. Wen, D. Yang, and C. Rosé, *Linguistic Reflections of Student Engagement in Massive Open Online Courses*. 2014.
[99] J. Kim, P. J. Guo, D. T. Seaton, P. Mitros, K. Z. Gajos, and R. C. Miller, 'Understanding in-video dropouts and interaction peaks inonline lecture videos', in *Proceedings of the first ACM conference on Learning @ scale conference*, Atlanta Georgia USA: ACM, Mar. 2014, pp. 31–40. doi: 10.1145/2556325.2566237.
[100] J. Reich and J. A. Ruipérez-Valiente, 'The MOOC pivot', *Science*, vol. 363, no. 6423, pp. 130–131, Jan. 2019, doi: 10.1126/science.aav7958.
[101] Y. Zheng, Z. Shao, M. Deng, Z. Gao, and Q. Fu, 'MOOC dropout prediction using a fusion deep model based on behaviour features', *Comput. Electr. Eng.*, vol. 104, p. 108409, Dec. 2022, doi: 10.1016/j.compeleceng.2022.108409.
[102] C. Geigle and C. Zhai, 'Modeling MOOC Student Behavior With Two-Layer Hidden Markov Models', in *Proceedings of the Fourth (2017) ACM Conference on Learning @ Scale*, Cambridge Massachusetts USA: ACM, Apr. 2017, pp. 205–208. doi: 10.1145/3051457.3053986.
[103] P. Oleksandra and D. Shane, 'Untangling MOOC learner networks', in *Proceedings of the Sixth International Conference on Learning Analytics & Knowledge*, in LAK '16. New York, NY, USA: Association for Computing Machinery, Apr. 2016, pp. 208–212. doi: 10.1145/2883851.2883919.
[104] 'Kizilcec et al. - 2013 - Deconstructing disengagement analyzing learner su.pdf'. Accessed: Jun. 12, 2025. [Online]. Available: https://rene.kizilcec.com/wp-content/uploads/2013/09/Kizilcec-Piech-Schneider-2013-Deconstructing-Disengagement-Analyzing-Learner-Subpopulations-in-Massive-Open-Online-Courses.pdf
[105] T. V. Semenova and L. M. Rudakova, 'Barriers to Taking Massive Open Online Courses (MOOCs)', *Russ. Educ. Soc.*, vol. 58, no. 3, pp. 228–245, Mar. 2016, doi: 10.1080/10609393.2016.1242992.
[106] T. Anderson and J. Dron, 'Three generations of distance education pedagogy', *Int. Rev. Res. Open Distrib. Learn.*, vol. 12, no. 3, p. 80, Mar. 2011, doi: 10.19173/irrodl.v12i3.890.
[107] K. Jordan, 'Initial trends in enrolment and completion of massive open online courses', *Int. Rev. Res. Open Distrib. Learn.*, vol. 15, no. 1, Jan. 2014, doi: 10.19173/irrodl.v15i1.1651.
[108] A. Kaplan and M. Haenlein, 'Higher education and the digital revolution: About MOOCs, SPOCs, social media, and the Cookie Monster', *Bus. Horiz.*, vol. 59, May 2016, doi: 10.1016/j.bushor.2016.03.008.
[109] T. R. Liyanagunawardena, A. A. Adams, and S. A. Williams, 'MOOCs: A systematic study of the published literature 2008-2012', *Int. Rev. Res. Open Distrib. Learn.*, vol. 14, no. 3, p. 202, Jul. 2013, doi: 10.19173/irrodl.v14i3.1455.
[110] M. Zhu, N. Sabir, C. J. Bonk, A. Sari, S. Xu, and M. Kim, 'ADDRESSING LEARNER CULTURAL DIVERSITY IN MOOC DESIGN AND DELIVERY: STRATEGIES AND

PRACTICES OF EXPERTS', *Turk. Online J. Distance Educ.*, pp. 1–25, Mar. 2021, doi: 10.17718/tojde.906468.
[111] K.-Z. Chen and B. Oakley, 'Redeveloping a global MOOC to be more locally relevant: design-based research', *Int. J. Educ. Technol. High. Educ.*, vol. 17, no. 1, p. 9, Feb. 2020, doi: 10.1186/s41239-020-0178-6.
[112] R. Kerr, R. D. Rosa, and I. Merciai, 'MULTILINGUAL LEARNING THROUGH MOOCS', *EDULEARN16 Proc.*, pp. 5081–5088, 2016, doi: 10.21125/edulearn.2016.2203.
[113] S. Rizvi, B. Rienties, J. Rogaten, and R. F. Kizilcec, 'Are MOOC learning designs culturally inclusive (enough)?', *J. Comput. Assist. Learn.*, vol. 40, no. 6, pp. 2496–2512, 2024, doi: 10.1111/jcal.12883.
[114] L. Rai, C. Deng, S. Lin, and L. Fan, 'Massive Open Online Courses and intercultural competence: analysis of courses fostering soft skills through language learning', *Front. Psychol.*, vol. 14, Aug. 2023, doi: 10.3389/fpsyg.2023.1219478.
[115] R. F. Kizilcec, C. Piech, and E. Schneider, 'Deconstructing disengagement: analyzing learner subpopulations in massive open online courses', in *Proceedings of the Third International Conference on Learning Analytics and Knowledge*, Leuven Belgium: ACM, Apr. 2013, pp. 170–179. doi: 10.1145/2460296.2460330.
[116] S. Gasmi, T. Bouhadada, and A. Benmachiche, 'Survey on Recommendation Systems', in *Proceedings of the 10th International Conference on Information Systems and Technologies*, in ICIST '20. New York, NY, USA: Association for Computing Machinery, Mar. 2021, pp. 1–7. doi: 10.1145/3447568.3448518.
[117] D. Yang, M. Wen, I. Howley, R. Kraut, and C. Rose, 'Exploring the Effect of Confusion in Discussion Forums of Massive Open Online Courses', in *Proceedings of the Second (2015) ACM Conference on Learning @ Scale*, in L@S '15. New York, NY, USA: Association for Computing Machinery, Mar. 2015, pp. 121–130. doi: 10.1145/2724660.2724677.
[118] P. Singh, 'Smart Learning with AI Driven Personalization in MOOCs', vol. 12, no. 5, 2025.
[119] Z. Zhao, 'AI-Driven Learning-Style Detection for Personalized MOOC Content Delivery in Lifelong Learning', *Int. J. Educ. Soc. Dev.*, vol. 3, no. 1, Art. no. 1, May 2025, doi: 10.54097/7d530x88.
[120] S. Dutta, S. Ranjan, S. Mishra, V. Sharma, P. Hewage, and C. Iwendi, 'Enhancing Educational Adaptability: A Review and Analysis of AI-Driven Adaptive Learning Platforms', in *2024 4th International Conference on Innovative Practices in Technology and Management (ICIPTM)*, Noida, India: IEEE, Feb. 2024, pp. 1–5. doi: 10.1109/ICIPTM59628.2024.10563448.
[121] D. Gomes, 'Intelligent Tutoring System A Comprehensive Study of Advancements in Intelligent Tutoring Systems through Artificial Intelligence Education Platform', 2024. doi: 10.4018/979-8-3693-6170-2.ch008.
[122] Y. Dan *et al.*, 'EduChat: A Large-Scale Language Model-based Chatbot System for Intelligent Education', Aug. 05, 2023, *arXiv*: arXiv:2308.02773. doi: 10.48550/arXiv.2308.02773.
[123] R. Cobos, 'The Learning Analytics System that Improves the Teaching-Learning Experience of MOOC Instructors and Students', in *Learning Technologies and Systems: 21st International Conference on Web-Based Learning, ICWL 2022, and 7th International Symposium on Emerging Technologies for Education, SETE 2022, Tenerife, Spain, November 21–23, 2022, Revised Selected Papers*, Berlin, Heidelberg: Springer-Verlag, Nov. 2022, pp. 29–40. doi: 10.1007/978-3-031-33023-0_3.
[124] P. M. Moreno-Marcos, C. Alario-Hoyos, P. J. Munoz-Merino, and C. D. Kloos, 'Prediction in MOOCs: A Review and Future Research Directions', *IEEE Trans. Learn. Technol.*, vol. 12, no. 3, pp. 384–401, Jul. 2019, doi: 10.1109/TLT.2018.2856808.
[125] Á. Becerra, R. Daza, R. Cobos, A. Morales, M. Cukurova, and J. Fierrez, 'M2LADS: A System for Generating MultiModal Learning Analytics Dashboards', in *2023 IEEE 47th Annual Computers, Software, and Applications Conference (COMPSAC)*, Torino, Italy: IEEE, Jun. 2023, pp. 1564–1569. doi: 10.1109/COMPSAC57700.2023.00241.
[126] S. Farokhi, A. Yaramala, J. Huang, M. F. A. Khan, X. Qi, and H. Karimi, 'Enhancing the Performance of Automated Grade Prediction in MOOC using Graph Representation Learning', Oct. 18, 2023, *arXiv*: arXiv:2310.12281. doi: 10.48550/arXiv.2310.12281.

[127] Y. Xue, F. Khalid, and A. Karim, ‘Emerging Trends in Self-Regulated Learning: A Bibliometric Analysis of MOOCs and AI-Enhanced Online Learning (2014–2024)’, *Int. J. Learn. Teach. Educ. Res.*, vol. 24, pp. 420–442, Jan. 2025, doi: 10.26803/ijlter.24.1.21.
[128] B. Ross, A.-M. Chase, D. Robbie, G. Oates, and Y. Absalom, ‘Adaptive quizzes to increase motivation, engagement and learning outcomes in a first year accounting unit’, *Int. J. Educ. Technol. High. Educ.*, vol. 15, Aug. 2018, doi: 10.1186/s41239-018-0113-2.
[129] S.-H. Jin, K. Im, M. Yoo, I. Roll, and K. Seo, ‘Supporting students’ self-regulated learning in online learning using artificial intelligence applications’, *Int. J. Educ. Technol. High. Educ.*, vol. 20, no. 1, p. 37, Jun. 2023, doi: 10.1186/s41239-023-00406-5.
[130] I. Gligorea, M. Cioca, R. Oancea, A.-T. Gorski, H. Gorski, and P. Tudorache, ‘Adaptive Learning Using Artificial Intelligence in e-Learning: A Literature Review’, *Educ. Sci.*, vol. 13, no. 12, Art. no. 12, Dec. 2023, doi: 10.3390/educsci13121216.
[131] D. Parsons, H. Sparks, D. Vo, and A. Singh, ‘MOOCS and Micro-Credentials’, 2023, pp. 1–13.
[132] L. Y. Tan, S. Hu, D. J. Yeo, and K. H. Cheong, ‘Artificial intelligence-enabled adaptive learning platforms: A review’, *Comput. Educ. Artif. Intell.*, vol. 9, p. 100429, Dec. 2025, doi: 10.1016/j.caeai.2025.100429.
[133] C. Merino-Campos, ‘The Impact of Artificial Intelligence on Personalized Learning in Higher Education: A Systematic Review’, *Trends High. Educ.*, vol. 4, no. 2, Art. no. 2, Jun. 2025, doi: 10.3390/higheredu4020017.
[134] R. Abadia, J. Fritsch, S. Abdelaal, and T. Jayawickrama, ‘Opportunities overcome challenges in adopting immersive virtual reality in online learning’, *Comput. Educ. Open*, vol. 7, p. 100208, Dec. 2024, doi: 10.1016/j.caeo.2024.100208.
[135] H. Ip *et al.*, ‘Design and Evaluate Immersive Learning Experience for Massive Open Online Courses (MOOCs)’, *IEEE Trans. Learn. Technol.*, vol. PP, pp. 1–1, Oct. 2018, doi: 10.1109/TLT.2018.2878700.
[136] T. Xie, X. Wang, J. Cifuentes-Faura, and Y. Xing, ‘Integrating immersive experience into hybrid education: a case study in fintech experimental education’, *Sci. Rep.*, vol. 13, no. 1, p. 22762, Dec. 2023, doi: 10.1038/s41598-023-50259-1.
[137] K. Bayly-Castaneda, M.-S. Ramirez-Montoya, and A. Morita-Alexander, ‘Crafting personalized learning paths with AI for lifelong learning: a systematic literature review’, *Front. Educ.*, vol. 9, p. 1424386, Aug. 2024, doi: 10.3389/feduc.2024.1424386.
[138] Z. Pan, L. Biegley, A. Taylor, and H. Zheng, ‘A Systematic Review of Learning Analytics: Incorporated Instructional Interventions on Learning Management Systems’, *J. Learn. Anal.*, vol. 11, no. 2, Art. no. 2, May 2024, doi: 10.18608/jla.2023.8093.
[139] A. Palanci, R. M. Yılmaz, and Z. Turan, ‘Learning analytics in distance education: A systematic review study’, *Educ. Inf. Technol.*, vol. 29, no. 17, pp. 22629–22650, Dec. 2024, doi: 10.1007/s10639-024-12737-5.
[140] Z. Chen, T. Zhang, and M. Taub, ‘How Does a Data-Informed Deliberate Change in Learning Design Impact Students’ Self-Regulated Learning Tactics?’, *J. Learn. Anal.*, vol. 11, no. 2, Art. no. 2, Jul. 2024, doi: 10.18608/jla.2024.8083.
[141] ‘A systematic review of the opportunities and challenges of micro-credentials for multiple stakeholders: learners, employers, higher education institutions and government | International Journal of Educational Technology in Higher Education’. Accessed: Jun. 27, 2025. [Online]. Available: https://link.springer.com/article/10.1186/s41239-023-00381-x
[142] M. Alenezi, M. Akour, and L. Alfawzan, ‘Evolving Microcredential Strategies for Enhancing Employability: Employer and Student Perspectives’, *Educ. Sci.*, vol. 14, no. 12, Art. no. 12, Dec. 2024, doi: 10.3390/educsci14121307.
[143] M. Brown and J. M. Duart, ‘Exploring gaps in the quality assurance of micro-credentials: a global mapping review of current practices’, *J. Open Distance Digit. Educ.*, vol. 1, no. 1, pp. 1–16, Aug. 2024, doi: 10.25619/2BWHVW68.
[144] M. Khalil, ‘Learning Analytics in Massive Open Online Courses’, Feb. 17, 2018, *arXiv*: arXiv:1802.09344. doi: 10.48550/arXiv.1802.09344.
[145] R. Rodrigues Major and M. and Mira da Silva, ‘Evaluating a gamified MOOC’, *Cogent Educ.*, vol. 12, no. 1, p. 2479400, Dec. 2025, doi: 10.1080/2331186X.2025.2479400.

[146] R. R. Major and M. Mira Da Silva, 'Gamification in MOOCs: A systematic literature review', *Cogent Educ.*, vol. 10, no. 2, p. 2275820, Dec. 2023, doi: 10.1080/2331186X.2023.2275820.
[147] J. Slamet, 'Promoting Autonomous Learning in ESP Courses through a Gamified MOOC Platform: A Self-Directed Learning Framework', *J. Educ. Online*, vol. 22, no. 2, Mar. 2025, doi: 10.9743/JEO.2025.22.2.7.
[148] A. S. Luccioni, G. Pistilli, R. Sefala, and N. Moorosi, 'Bridging the Gap: Integrating Ethics and Environmental Sustainability in AI Research and Practice', Apr. 01, 2025, *arXiv*: arXiv:2504.00797. doi: 10.48550/arXiv.2504.00797.
[149] D. Cumming, K. Saurabh, N. Rani, and P. Upadhyay, 'Towards AI ethics-led sustainability frameworks and toolkits: Review and research agenda', *J. Sustain. Finance Account.*, vol. 1, p. 100003, Mar. 2024, doi: 10.1016/j.josfa.2024.100003.
[150] M. Boutassetta, A. Makhlouf, N. Messaoudi, A. Benmachiche, and I. Boutabia, 'Hybrid IDS Using Signature-Based and Anomaly-Based Detection', Jan. 17, 2026, *arXiv*: arXiv:2601.11998. doi: 10.48550/arXiv.2601.11998.
[151] B. Oumaima, A. Benmachiche, M. Maatallah, and C. Chemam, *An Extensive Examination of Varied Approaches in E-Learning and MOOC Research: A Thorough Overview*. 2024, p. 8. doi: 10.1109/PAIS62114.2024.10541129.
[152] B. S. Oumaima, 'A HYBRID SYSTEM FOR RECOMMENDING BOOKS', université chadli ben djedid eltarf, 2021. Accessed: Jul. 13, 2026. [Online]. Available: http://depotucbet.univ-eltarf.dz:4000/handle/123456789/2154
[153] A. Nazyrova, M. Milosz, G. Bekmanova, A. Omarbekova, A. Mukanova, and G. Aimicheva, 'Analysis of the Consistency of Prerequisites and Learning Outcomes of Educational Programme Courses by Using the Ontological Approach', *Appl. Sci.*, vol. 13, no. 4, p. 2661, Feb. 2023, doi: 10.3390/app13042661.
[154] N. Evain, E. Exposito, M. L. Gueye, and P. Arnould, 'Ontology-Driven Approach for Competency-Oriented and Student-Centered Engineering Education', in *2024 IEEE Global Engineering Education Conference (EDUCON)*, Kos Island, Greece: IEEE, May 2024, pp. 1–10. doi: 10.1109/EDUCON60312.2024.10578793.
[155] S. Amin, M. I. Uddin, W. K. Mashwani, A. A. Alarood, A. Alzahrani, and A. O. Alzahrani, 'Developing a Personalized E-Learning and MOOC Recommender System in IoT-Enabled Smart Education', *IEEE Access*, vol. 11, pp. 136437–136455, 2023, doi: 10.1109/ACCESS.2023.3336676.
[156] K. Ouassif and B. Ziani, 'Predicting university major selection and academic performance through the combination of Apriori algorithm and deep neural network', *Educ. Inf. Technol.*, vol. 30, no. 1, pp. 333–346, Jan. 2025, doi: 10.1007/s10639-024-13022-1.
[157] A. David, D. Mihai, M.-E. Mihailescu, M. Carabas, and N. Tapus, 'Scalability through Distributed Deployment for Moodle Learning Management System', *Procedia Comput. Sci.*, vol. 214, pp. 34–41, 2022, doi: 10.1016/j.procs.2022.11.145.
[158] Q. Zhang and L. Sun, 'CNN-Bi-LSTM Model for MOOC Forum Post Classification', *Int. J. Emerg. Technol. Learn. IJET*, vol. 18, no. 21, pp. 89–101, Nov. 2023, doi: 10.3991/ijet.v18i21.37843.
[159] A. Bouamrane, M. Derdour, M. S. Kahil, A. Benmachiche, M. Deriche, and M. Nasor, 'A Novel Sequential Mobility-Based Federated Learning Framework with Agentic Explainable AI for Histopathology-Based Lung Diagnosis', in *2026 IEEE 23rd International Multi-Conference on Systems, Signals & Devices (SSD)*, IEEE, 2026, pp. 855–864. Accessed: Jul. 07, 2026. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/11558924/
[160] X. Wei, H. Lin, L. Yang, and Y. Yu, 'A Convolution-LSTM-Based Deep Neural Network for Cross-Domain MOOC Forum Post Classification', *Information*, vol. 8, no. 3, Art. no. 3, Sep. 2017, doi: 10.3390/info8030092.
[161] C. Yan, J. Liu, W. Liu, and X. Liu, 'Sentiment Analysis and Topic Mining Using a Novel Deep Attention-Based Parallel Dual-Channel Model for Online Course Reviews', *Cogn. Comput.*, vol. 15, no. 1, pp. 304–322, Jan. 2023, doi: 10.1007/s12559-022-10083-7.
[162] K. Talebi, Z. Torabi, and N. Daneshpour, 'Ensemble models based on CNN and LSTM for dropout prediction in MOOC', *Expert Syst. Appl.*, vol. 235, p. 121187, 2024, Accessed: Feb.

06, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417423016895
[163] 'Comparative Analysis of Attention Mechanisms in Pix2Pix for Multimodal MRI Fusion | Journal of Electronics, Electromedical Engineering, and Medical Informatics'. Accessed: Jul. 06, 2026. [Online]. Available: https://www.jeeemi.org/index.php/jeeemi/article/view/1720
[164] A. Bouamrane, M. Derdour, A. Bennour, A. Benmachiche, and M. Gasmi, 'Machine Learning for Medical Image Analysis', in *AI for Medical Image Analysis*, N. Ben Aoun, S. Ahmad, and M. Hammad, Eds, Cham: Springer Nature Switzerland, 2026, pp. 97–125. doi: 10.1007/978-3-032-02963-8_4.
[165] H. Sebbaq and N. El Faddouli, 'Fine-tuned BERT Model for Large Scale and Cognitive Classification of MOOCs', *Int. Rev. Res. Open Distrib. Learn.*, vol. 23, no. 2, pp. 170–190, May 2022, doi: 10.19173/irrodl.v23i2.6023.
[166] X. Chen, D. Zou, H. Xie, G. Cheng, Z. Li, and F. L. Wang, 'Automatic Classification of Online Learner Reviews Via Fine-Tuned BERTs', *Automatic Classification of Online Learner Reviews Via Fine-Tuned BERTs*, pp. 57-79., 2025.
[167] A. Alkhalil, 'A Neural Network Framework for Classifying Educational Content Aligned with Learning Outcomes', *Procedia Comput. Sci.*, vol. 257, pp. 529–536, 2025, doi: 10.1016/j.procs.2025.03.068.
[168] M. A. El-Rashidy, A. Farouk, N. A. El-Fishawy, H. K. Aslan, and N. A. Khodeir, 'New weighted BERT features and multi-CNN models to enhance the performance of MOOC posts classification', *Neural Comput. Appl.*, vol. 35, no. 24, pp. 18019–18033, Aug. 2023, doi: 10.1007/s00521-023-08673-z.
[169] N. Khodeir and F. Elghannam, 'Efficient topic identification for urgent MOOC Forum posts using BERTopic and traditional topic modeling techniques', *Educ. Inf. Technol.*, Sep. 2024, doi: 10.1007/s10639-024-13003-4.
[170] B. Sedraoui, A. Benmachiche, A. Makhlouf, and C. Chemam, *Intrusion Detection with deep learning: A literature review*. 2024, p. 8. doi: 10.1109/PAIS62114.2024.10541191.
[171] J. Yu *et al.*, 'MOOCCubeX: A Large Knowledge-centered Repository for Adaptive Learning in MOOCs', in *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, Virtual Event Queensland Australia: ACM, Oct. 2021, pp. 4643–4652. doi: 10.1145/3459637.3482010.
[172] Z. Liang, L. Mu, J. Chen, and Q. Xie, 'Graph path fusion and reinforcement reasoning for recommendation in MOOCs', *Educ. Inf. Technol.*, vol. 28, no. 1, pp. 525–545, Jan. 2023, doi: 10.1007/s10639-022-11178-2.
[173] N. Tyagi, S. Sarkar, and M. Gaur, 'Leveraging Knowledge and Reinforcement Learning for Enhanced Reliability of Language Models', Aug. 25, 2023, *arXiv*: arXiv:2308.13467. doi: 10.48550/arXiv.2308.13467.
[174] K. Naji and A. Ibriz, 'Approach for Eliciting Learners' Preferences in Moocs Through Collaborative Filtering', *Int. J. Emerg. Technol. Learn. IJET*, vol. 17, no. 14, pp. 235–245, Jul. 2022, doi: 10.3991/ijet.v17i14.29887.
[175] M. Loukili and F. Messaoudi, 'Enhancing Cold-Start Recommendations with Innovative Co-SVD: A Sparsity Reduction Approach', *Stat. Optim. Inf. Comput.*, vol. 13, no. 1, pp. 396–408, Aug. 2024, doi: 10.19139/soic-2310-5070-2048.
[176] S. Assami, D. Najima, and R. Ajhoun, 'Implementation of a Machine Learning-Based MOOC Recommender System Using Learner Motivation Prediction', *Int. J. Eng. Pedagogy IJEP*, vol. 12, pp. 68–85, Nov. 2022, doi: 10.3991/ijep.v12i5.30523.
[177] I. El Mourabit, S. Jai Andaloussi, O. Ouchetto, and M. Miyara, 'Exploring Collaborative Filtering Algorithms in MOOCs Recommender Systems: A Comprehensive Overview', in *Proceedings of the 6th International Conference on Networking, Intelligent Systems & Security*, Larache Morocco: ACM, May 2023, pp. 1–5. doi: 10.1145/3607720.3607742.
[178] F. Safarov, A. Kutlimuratov, A. B. Abdusalomov, R. Nasimov, and Y.-I. Cho, 'Deep Learning Recommendations of E-Education Based on Clustering and Sequence', *Electronics*, vol. 12, no. 4, p. 809, Feb. 2023, doi: 10.3390/electronics12040809.

[179] Y. Li *et al.*, '*MHRR* : MOOCs Recommender Service With Meta Hierarchical Reinforced Ranking', *IEEE Trans. Serv. Comput.*, vol. 16, no. 6, pp. 4467–4480, Nov. 2023, doi: 10.1109/TSC.2023.3325302.

[180] A. M. Alfurjani and I. S. Mansour, 'Enhanced E-learning through Neural Network Based Cloud Data Extraction', pp. 281-292., Enhanced e-learning through neural network based cloud data extraction.

[181] X. Xia and W. Qi, 'Driving STEM learning effectiveness: dropout prediction and intervention in MOOCs based on one novel behavioral data analysis approach', *Humanit. Soc. Sci. Commun.*, vol. 11, no. 1, p. 430, Mar. 2024, doi: 10.1057/s41599-024-02882-0.

[182] A. Benmachiche, A. Sahia, S. O. Boufaida, K. Rais, M. Derdour, and F. Maazouzi, 'Enhancing learning recommendations in mooc search engines through named entity recognition', *Educ. Inf. Technol.*, Jan. 2025, doi: 10.1007/s10639-024-13308-4.

[183] I. Boutabia, A. Benmachiche, A. A. Betouil, C. Chemam, and K. Rais, 'Advanced Text Prediction System Integrated Within the Search Engine for the Open Classroom Approach Based on Particle Swarm Optimization and Long Short-Term Memory Models', *Arab. J. Sci. Eng.*, Mar. 2026, doi: 10.1007/s13369-026-11247-5.

[184] H. Liu *et al.*, 'GeAR: Generation Augmented Retrieval', Jan. 06, 2025, *arXiv*: arXiv:2501.02772. doi: 10.48550/arXiv.2501.02772.

[185] Y. Jia, 'BTCBMA Online Education Course Recommendation Algorithm Based on Learners' Learning Quality:', *Int. J. Inf. Technol. Syst. Approach*, vol. 16, no. 1, pp. 1–17, Jun. 2023, doi: 10.4018/IJITSA.324101.

[186] A. Agarwal, D. S. Mishra, and S. V. Kolekar, 'Knowledge-based recommendation system using semantic web rules based on Learning styles for MOOCs', *Cogent Eng.*, vol. 9, no. 1, p. 2022568, Dec. 2022, doi: 10.1080/23311916.2021.2022568.

[187] H. Slimani, O. Hamal, N. El Faddouli, S. Bennani, and N. Amrous, 'The hybrid recommendation of digital educational resources in a distance learning environment: the case of MOOC', in *Proceedings of the 13th International Conference on Intelligent Systems: Theories and Applications*, Rabat Morocco: ACM, Sep. 2020, pp. 1–9. doi: 10.1145/3419604.3419621.

[188] V. Pleskach, K. Tkachenko, O. Tkachenko, and O. Tkachenko, 'Using Ontologies and Knowledge Graphs to Individualize in E- Learning System', 2023.

[189] Y. Gomaa, C. Lahoud, M.-H. Abel, and S. Moussa, 'Managing Data Heterogeneity for Ontology-Driven Models: Application to Gamified E-Learning Contexts':, in *Proceedings of the 17th International Conference on Computer Supported Education*, Porto, Portugal: SCITEPRESS - Science and Technology Publications, 2025, pp. 726–736. doi: 10.5220/0013496700003932.

[190] S. Bousalem, F. Benchikha, and N. Marir, 'Personalized Learning Through MBTI Prediction: A Deep Learning Approach Integrated With Learner Profile Ontology', *IEEE Access*, vol. 13, pp. 30861–30873, 2025, doi: 10.1109/ACCESS.2025.3542701.

[191] J. Honeder, A. Nussbaumer, and C. Guetl, 'A Geo-Contextualized Multi-Genre Scientific Search Engine'.

[192] K. M. Alzahrani and M. Meccawy, 'MOOCs One-Stop Shop: A Realization of a Unified MOOCs Search Engine', *IEEE Access*, vol. 9, pp. 160175–160185, 2021, doi: 10.1109/ACCESS.2021.3130841.

[193] J. R. Kumar, B. Dheeraj, G. Gowtham, D. V. Babu, B. A. Kiran, and M. Srikanth, 'Image Search Engine with Recognition', in *2025 4th International Conference on Sentiment Analysis and Deep Learning (ICSADL)*, Bhimdatta, Nepal: IEEE, Feb. 2025, pp. 1217–1223. doi: 10.1109/ICSADL65848.2025.10933116.

[194] S. O. Boufaida, A. Benmachiche, and M. Maatallah, 'Real-Time Image Processing Algorithms for Embedded Systems', Jan. 09, 2026, *arXiv*: arXiv:2601.06243. doi: 10.48550/arXiv.2601.06243.

[195] R. Deng, 'Effect of video styles on learner engagement in MOOCs', *Technol. Pedagogy Educ.*, vol. 33, no. 1, pp. 1–21, Jan. 2024, doi: 10.1080/1475939X.2023.2246981.

[196] 'CNN-OOA-Based Cyber Threat Detection: Protecting E-Learning from Phishing | Arabian Journal for Science and Engineering | Springer Nature Link'. Accessed: Jul. 06, 2026. [Online]. Available: https://link.springer.com/article/10.1007/s13369-026-11122-3

[197] Akanbi M. B, Okike B, O. Owolabi, and M. B. Hamawa, ‘Developing Explainable Artificial Intelligence to Enhance Transparency and Personalization in E-Learning Recommendation Systems’, Jan. 2025, doi: 10.5281/ZENODO.14775809.
[198] S. Wu *et al.*, ‘A Comprehensive Exploration of Personalized Learning in Smart Education: From Student Modeling to Personalized Recommendations’, Jan. 15, 2024, *arXiv*: arXiv:2402.01666. doi: 10.48550/arXiv.2402.01666.
[199] N. S. K. M. K. Tirumanadham, S. Thaiyalnayaki, and V. Ganesan, ‘Towards Smarter E-Learning: Real-Time Analytics and Machine Learning for Personalized Education’, *Int. J. Comput. Exp. Sci. Eng.*, vol. 11, no. 1, Jan. 2025, doi: 10.22399/ijcesen.786.
[200] ‘Enhancing learning recommendations in mooc search engines through named entity recognition | Education and Information Technologies’. Accessed: Jan. 02, 2026. [Online]. Available: https://link.springer.com/article/10.1007/s10639-024-13308-4
[201] S. Amin, M. I. Uddin, A. A. Alarood, W. K. Mashwani, A. Alzahrani, and A. O. Alzahrani, ‘Smart E-Learning Framework for Personalized Adaptive Learning and Sequential Path Recommendations Using Reinforcement Learning’, *IEEE Access*, vol. 11, pp. 89769–89790, 2023, doi: 10.1109/ACCESS.2023.3305584.
[202] A. Z. Azhar, A. Segal, and K. Gal, ‘Optimizing Representations and Policies for Question Sequencing using Reinforcement Learning’, Jul. 2022, doi: 10.5281/ZENODO.6853123.
[203] Y. Zhang, ‘Developing an Adaptive Learning Recommendation Algorithm and System for MOOCs’, *J. Mach. Comput.*, pp. 962–970, Oct. 2024, doi: 10.53759/7669/jmc202404089.
[204] M. Abrar, W. Aboraya, R. Khaliq, K. Subramanian, Y. Al Husaini, and M. Abdulla, ‘AI-Powered Learning Pathways: Personalized Learning and Dynamic Assessments’, *Int. J. Adv. Comput. Sci. Appl.*, vol. 16, Jan. 2025, doi: 10.14569/IJACSA.2025.0160145.
[205] H. Zhang, T. Huang, Z. Lv, S. Liu, and Z. Zhou, ‘MCRS: A course recommendation system for MOOCs’, *Multimed. Tools Appl.*, vol. 77, no. 6, pp. 7051–7069, Mar. 2018, doi: 10.1007/s11042-017-4620-2.
[206] K. Rabahallah, L. Mahdaoui, and F. Azouaou, ‘MOOCs Recommender System using Ontology and Memory-based Collaborative Filtering.’, in *ICEIS (1)*, 2018, pp. 635–641. Accessed: Jun. 14, 2025. [Online]. Available: https://www.academia.edu/download/80191363/67860.pdf
[207] S. Yin, K. Yang, and H. Wang, ‘A MOOC Courses Recommendation System Based on Learning Behaviours’, in *Proceedings of the ACM Turing Celebration Conference - China*, Hefei China: ACM, May 2020, pp. 133–137. doi: 10.1145/3393527.3393550.
[208] J. Tan, L. Chang, T. Liu, and X. Zhao, ‘Attentional autoencoder for course recommendation in mooc with course relevance’, in *2020 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC)*, IEEE, 2020, pp. 190–196. Accessed: Jun. 14, 2025. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9329380/
[209] G. Anand, S. Kumari, and R. Pulle, ‘Fractional-Iterative BiLSTM Classifier : A Novel Approach to Predicting Student Attrition in Digital Academia’, *Int. J. Comput. Sci. Eng.*, vol. 10, no. 5, pp. 1–9, May 2023, doi: 10.14445/23488387/IJCSE-V10I5P101.
[210] Q. Chen, X. Yu, N. Liu, X. Yuan, and Z. Wang, ‘Personalized Course Recommendation Based on Eye-Tracking Technology and Deep Learning’, in *2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA)*, sydney, Australia: IEEE, Oct. 2020, pp. 692–968. doi: 10.1109/DSAA49011.2020.00079.
[211] M. Homola, Z. Kubincová, R. Urbánek, and J. Kľuka, ‘Tracking the Adaptive Learning Process with Topics Ontology’, in *Advances in Web-Based Learning – ICWL 2023*, vol. 14409, H. Xie, C.-L. Lai, W. Chen, G. Xu, and E. Popescu, Eds, in Lecture Notes in Computer Science, vol. 14409. , Singapore: Springer Nature Singapore, 2023, pp. 155–165. doi: 10.1007/978-981-99-8385-8_13.
[212] Y. He *et al.*, ‘Online At-Risk Student Identification Using RNN-GRU Joint Neural Networks’, 2020.
[213] Y.-F. Zheng, Z.-H. Gao, Y.-H. Wang, and Q. Fu, ‘MOOC Dropout Prediction Using FWTS-CNN Model Based on Fused Feature Weighting and Time Series’, *IEEE Access*, 2020.
[214] H. Waheed, S.-U. Hassan, R. Nawaz, N. R. Aljohani, G. Chen, and D. Gasevic, ‘Early prediction of learners at risk in self-paced education: A neural network approach’, *Expert Syst. Appl.*, vol. 213, p. 118868, Mar. 2023, doi: 10.1016/j.eswa.2022.118868.

[215] E. Er, 'An Explainable Machine Learning Approach to Predicting and Understanding Dropouts in MOOCs', *Kastamonu Eğitim Derg.*, vol. 31, no. 1, pp. 143–154, Jan. 2023, doi: 10.24106/kefdergi.1246458.
[216] S. Rizwan, C. K. Nee, and S. Garfan, 'Identifying the Factors Affecting Student Academic Performance and Engagement Prediction in MOOC Using Deep Learning: A Systematic Literature Review', *IEEE Access*, vol. 13, pp. 18952–18982, 2025, doi: 10.1109/ACCESS.2025.3533915.
[217] A. M. Shitaya, M. E. S. Wahed, M. Y. Shams, and A. A. Salama, 'Predicting Student Behavior Using a Neutrosophic Deep Learning Model', *A*, vol. 76, 2025.
[218] Z. Shou, P. Chen, H. Wen, J. Liu, and H. Zhang, 'MOOC Dropout Prediction Based on Multidimensional Time-Series Data', *Math. Probl. Eng.*, vol. 2022, pp. 1–12, Apr. 2022, doi: 10.1155/2022/2213292.
[219] M. J. S. Barón, J. S. G. Sanabria, and J. E. E. Díaz, *Neural Network Applied to the Analysis of.pdf*. n.d, 2022.
[220] D. Abbas, A. Benmachiche, M. Derdour, and B. K. Sedraoui, 'Toward Trustworthy IoT Systems: Integrating Blockchain and Artificial Intelligence'.
[221] N. S. Rahayu, M. H. Dhiaullah, and A. Marsha, 'Utilizing e-learning and user loyalty with user satisfaction as mediating variable in public sector context', *Int. J. Data Netw. Sci.*, vol. 7, no. 3, pp. 1341–1348, 2023, doi: 10.5267/j.ijdns.2023.4.004.
[222] 'Paper+59+(2025.7.4)+The+Transformative+Potential…+JCSTS.pdf'.
[223] W. G. Alghabban and R. Hendley, 'Adaptive E-Learning and Dyslexia: an Empirical Evaluation and Recommendations for Future Work', *Interact. Comput.*, p. iwad036, Jun. 2023, doi: 10.1093/iwc/iwad036.
[224] '8+Personalized+Learning+in+Education+through+AIoT_+Adaptive+Systems+for+Student+Engagement+and+Performa.pdf'.
[225] A. Benmachiche, B. Hadjar, I. Boutabia, A. A. Betouil, M. Maatallah, and A. Makhlouf, 'Development of a biometric authentication platform using voice recognition', in *2022 4th International Conference on Pattern Analysis and Intelligent Systems (PAIS)*, IEEE, 2022, pp. 1–7. Accessed: Jul. 07, 2026. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9946890/
[226] W. Alarefah, S. K. Jarraya, and N. Abuzinadah, 'Transformer-Based Student Engagement Recognition Using Few-Shot Learning', *Computers*, vol. 14, no. 3, p. 109, Mar. 2025, doi: 10.3390/computers14030109.
[227] P. Bhardwaj, P. K. Gupta, H. Panwar, M. K. Siddiqui, R. Morales-Menendez, and A. Bhaik, 'Application of Deep Learning on Student Engagement in e-learning environments', *Comput. Electr. Eng.*, vol. 93, p. 107277, Jul. 2021, doi: 10.1016/j.compeleceng.2021.107277.
[228] S. Kim, S. Cho, J. Y. Kim, and D.-J. Kim, 'Statistical Assessment on Student Engagement in Asynchronous Online Learning Using the k-Means Clustering Algorithm', *Sustainability*, vol. 15, no. 3, p. 2049, Jan. 2023, doi: 10.3390/su15032049.
[229] B. Su and J. Peng, 'Sentiment Analysis of Comment Texts on Online Courses Based on Hierarchical Attention Mechanism', *Appl. Sci.*, vol. 13, no. 7, p. 4204, Mar. 2023, doi: 10.3390/app13074204.
[230] J. Chonraksuk and S. Boonlue, 'Development of an AI predictive model to categorize and predict online learning behaviors of students in Thailand', *Heliyon*, vol. 10, no. 11, p. e32591, Jun. 2024, doi: 10.1016/j.heliyon.2024.e32591.
[231] K. Benabbes, K. Housni, B. Hmedna, A. Zellou, and A. E. Mezouary, 'A New Hybrid Approach to Detect and Track Learner's Engagement in e-Learning', *IEEE Access*, vol. 11, pp. 70912–70929, 2023, doi: 10.1109/ACCESS.2023.3293827.
[232] B. Sedraoui, A. Benmachiche, A. Bennour, A. Makhlouf, M. Derdour, and D. Ghabban, 'LSTM-SWAP: A Hybrid Deep Learning Model for Cheating Detection', *SN Comput. Sci.*, vol. 6, Sep. 2025, doi: 10.1007/s42979-025-04334-1.
[233] M. Boutassetta, A. Makhlouf, N. Messaoudi, A. Benmachiche, I. Boutabia, and M. Derdour, *Cyberattack Detection in Smart Cities Using AI: A literature review*. 2025, p. 9. doi: 10.1109/ICNAS68168.2025.11298103.

[234] Department of Computing, Muscat College, Sultanate of Oman and A. N. Doss, ‘Learning Analytics Model for Predictive Analysis of Learners Behavior for an Indigenous MOOC Platform (Tadakhul System) in Oman’, *Int. J. Inf. Educ. Technol.*, vol. 14, no. 7, pp. 961–967, 2024, doi: 10.18178/ijiet.2024.14.7.2123.
[235] G. Feng, M. Fan, and Y. Chen, ‘Analysis and Prediction of Students’ Academic Performance Based on Educational Data Mining’, *IEEE Access*, vol. 10, pp. 19558–19571, 2022, doi: 10.1109/ACCESS.2022.3151652.
[236] P. Mukala, ‘Unveiling the Synergy of Process Mining, Explainable AI, and Learning Analytics in Advancing Educational Data Interpretability: Paving the Way for a New Era in Educational Analytics’, Jan. 31, 2025. doi: 10.36227/techrxiv.173835396.67982180/v1.
[237] B. Kong, E. Hemberg, A. Bell, and U.-M. O’Reilly, ‘Investigating Student’s Problem-solving Approaches in MOOCs using Natural Language Processing’, in *LAK23: 13th International Learning Analytics and Knowledge Conference*, Arlington TX USA: ACM, Mar. 2023, pp. 262–272. doi: 10.1145/3576050.3576091.
[238] H. Sebbeq and N. El Faddouli, ‘Towards Quality Assurance in MOOCs: A Comprehensive Review and Micro-Level Framework’, *Int. Rev. Res. Open Distrib. Learn.*, vol. 25, no. 1, pp. 1–23, Mar. 2024, doi: 10.19173/irrodl.v25i1.7544.
[239] O. Agatova and L. Latipova, ‘Exploring AI-powered personalized learning in universities’, *Int. J. Eval. Res. Educ. IJERE*, vol. 14, no. 2, p. 837, Apr. 2025, doi: 10.11591/ijere.v14i2.29899.
[240] A. Chanaa and N. El Faddouli, ‘Sentiment Analysis on Massive Open Online Courses (MOOCs): Multi-Factor Analysis, and Machine Learning Approach’, *Int. J. Inf. Commun. Technol. Educ.*, vol. 18, no. 1, pp. 1–22, Sep. 2022, doi: 10.4018/IJICTE.310004.
[241] J. Pecuchova and M. Drlik, ‘Enhancing the Early Student Dropout Prediction Model Through Clustering Analysis of Students’ Digital Traces’, *IEEE Access*, vol. 12, pp. 159336–159367, 2024, doi: 10.1109/ACCESS.2024.3486762.
[242] A. Ismail, A. Ismail, A. Alazar, M. Saman, A. Abu-Elqomboz, and F. K. Sharaf, ‘E-Learning Medical Education in Gaza During COVID-19: Students’ Experiences and Policy Recommendations’, *J. Med. Educ. Curric. Dev.*, vol. 10, p. 23821205231164228, Jan. 2023, doi: 10.1177/23821205231164228.
[243] Y. Hu, Z. Jiang, and K. Zhu, ‘An Optimized CNN Model for Engagement Recognition in an E-Learning Environment’, *Appl. Sci.*, vol. 12, no. 16, p. 8007, Aug. 2022, doi: 10.3390/app12168007.
[244] ‘RNP Deep Neural Network Applied to the Analysis of.pdf’.
[245] F. Alfiana *et al.*, ‘Apply the Search Engine Optimization (SEO) Method to determine Website Ranking on Search Engines’, *Int. J. Cyber IT Serv. Manag.*, vol. 3, no. 1, pp. 65–73, Mar. 2023, doi: 10.34306/ijcitsm.v3i1.126.
[246] X. Li, J. Lipp, A. Shakir, R. Huang, and J. Li, ‘BMX: Entropy-weighted Similarity and Semantic-enhanced Lexical Search’, Aug. 14, 2024, *arXiv*: arXiv:2408.06643. doi: 10.48550/arXiv.2408.06643.
[247] M. Redjimi, M. Majda, A. Benmachiche, B. Oumaima, and M. Derdour, *A Comprehensive Survey on Blockchain, Federated Learning, and AI for Securing the Internet of Medical Things*. 2025, p. 8. doi: 10.1109/ICNAS68168.2025.11298047.
[248] M. Maatallah, A. Benmachiche, K. Rais, and S. Touam, ‘Intelligent Fusion of Multi-Modal Medical Imaging: A Comprehensive Review of Methods, Challenges, and Clinical Integration’, *J. Electron. Electromed. Eng. Med. Inform.*, vol. 8, no. 3, pp. 897–934, May 2026, doi: 10.35882/jeeemi.v8i3.1527.
[249] S. Kim, N. Fiorini, W. J. Wilbur, and Z. Lu, ‘Bridging the Gap: Incorporating a Semantic Similarity Measure for Effectively Mapping PubMed Queries to Documents’, *J. Biomed. Inform.*, vol. 75, pp. 122–127, Nov. 2017, doi: 10.1016/j.jbi.2017.09.014.
[250] J. Guo *et al.*, ‘A Deep Look into Neural Ranking Models for Information Retrieval’, Jun. 27, 2019, *arXiv*: arXiv:1903.06902. Accessed: Aug. 02, 2024. [Online]. Available: http://arxiv.org/abs/1903.06902
[251] ‘Dense Retrieval for Efficient Paper Retrieval in Academic Question Answering’, 2024.
[252] Sengar, B. S., ‘Search Engine Optimization with Artificial Intelligence a Paradigm for Targeted Marketing.’, *Int. J. Innov. Sci. Eng. Manag.*, pp. 134–139.

[253] ‘enhancing-search-engine-optimization-through-artificial-intelligence.pdf’.
[254] J. Gong *et al.*, ‘Attentional Graph Convolutional Networks for Knowledge Concept Recommendation in MOOCs in a Heterogeneous View’, in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, Virtual Event China: ACM, Jul. 2020, pp. 79–88. doi: 10.1145/3397271.3401057.
[255] W. M. Elhaj-Suliman, ‘Design and Implementation of a Proposed Model for a Semantic-Based Search Engine’, 2025.
[256] A. Sharma, ‘The Impact of AI-Powered Search on SEO: The Emergence of Answer Engine Optimization’.
[257] R. Mounira, M. Maatallah, A. Benmachiche, and O. Boufaida, ‘Enhancing Malware Detection Robustness through Hybrid Adversarial Machine Learning’.
[258] F. Nestaas, E. Debenedetti, and F. Tramèr, ‘Adversarial Search Engine Optimization for Large Language Models’, Jul. 02, 2024, *arXiv*: arXiv:2406.18382. doi: 10.48550/arXiv.2406.18382.
[259] J. Kuai, C. Brantner, M. Karlsson, E. Van Couvering, and S. Romano, ‘AI chatbot accountability in the age of algorithmic gatekeeping: Comparing generative search engine political information retrieval across five languages’, *New Media Soc.*, p. 14614448251321162, Feb. 2025, doi: 10.1177/14614448251321162.
[260] Z. Hailemariam, ‘Amharic Language Audio Data Search Engine’.
[261] A. Benmachiche and A. Makhlouf, ‘Optimization of Hidden Markov Model with Gaussian Mixture Densities for Arabic Speech Recognition’, *WSEAS Trans. Circuits Syst.*, vol. 15, pp. 85–95, Apr. 2019.
[262] A. Benmachiche, A. Makhlouf, and T. Bouhadada, ‘Evolutionary learning of HMM with Gaussian mixture densities for Automatic speech recognition’, in *Proceedings of the 9th International Conference on Information Systems and Technologies*, Cairo Egypt: ACM, Mar. 2019, pp. 1–6. doi: 10.1145/3361570.3361591.
[263] A. Benmachiche, A. Makhlouf, and T. Bouhadada, ‘Optimization learning of hidden Markov model using the bacterial foraging optimization algorithm for speech recognition’, *Int. J. Knowl.-Based Intell. Eng. Syst.*, vol. 24, no. 3, pp. 171–181, Sep. 2020, doi: 10.3233/KES-200039.
[264] L. J. Droscha, S. Chung, Z. Li-Khan, A. Scott, and E. Rubenstein, ‘Perception of four intellectual and developmental disabilities based on search engine and news portrayal’, *PLOS ONE*, vol. 20, no. 2, p. e0316928, Feb. 2025, doi: 10.1371/journal.pone.0316928.
[265] N. Makrydakis, ‘SEO mix 6 O’s model and categorization of search engine marketing factors for websites ranking on search engine result pages’, *Int. J. Res. Mark. Manag. Sales*, vol. 6, no. 1, pp. 18–32, Jan. 2024, doi: 10.33545/26633329.2024.v6.i1a.146.
[266] M. Wu and S. Cao, ‘LLM-Augmented Retrieval: Enhancing Retrieval Models Through Language Models and Doc-Level Embedding’, Apr. 08, 2024, *arXiv*: arXiv:2404.05825. doi: 10.48550/arXiv.2404.05825.
[267] B. Oumaima, A. Benmachiche, M. Derdour, M. Maatallah, M. Kahil, and M. Ghanem, ‘TSA-GRU: A Novel Hybrid Deep Learning Module for Learner Behavior Analytics in MOOCs’, *Future Internet*, vol. 17, Aug. 2025, doi: 10.3390/fi17080355.
[268] B. Oumaima, A. Benmachiche, A. Bennour, M. Maatallah, M. Derdour, and D. Ghabban, ‘Enhancing MOOC Course Classification with Convolutional Neural Networks via Lion Algorithm-Based Hyperparameter Tuning’, *SN Comput. Sci.*, vol. 6, Jul. 2025, doi: 10.1007/s42979-025-04179-8.
[269] ‘Recommendation approach of ed ucational resources in a fre-quent subnet | ASJP’. Accessed: Jul. 07, 2026. [Online]. Available: https://asjp.cerist.dz/en/article/259528
[270] ‘ASSISTmentsData - 2009-2010 ASSISTment Data’. Accessed: Mar. 13, 2025. [Online]. Available: https://sites.google.com/site/assistmentsdata/home/2009-2010-assistment-data