Computational thinking with computer vision:Developing AI Competency In An Introductory Computer Science Course (e-learning)

Sagi Akansha
Department Of Computer Science and Engineering
Talla Padmavathi College Of Engineering (Autonomus),
Somidi, Kazipet, Telangana.
Email: akansharaosagi14@gmail.com

Mr. E. Ajith Kumar
Assistant Professor, Department of CSE
Talla Padmavathi College Of Engineering(Autonomus),
Somidi, Kazipet, Telangana.
Email: ajithkumartpce@gmail.com

Abstract-

The widespread adoption of digital education platforms has dramatically increased the availability of learning materials, including online courses, instructional videos, assessments, tutorials, and interactive resources. While this abundance of content offers greater learning opportunities, it also creates challenges for learners who must identify resources that align with their interests, skill levels, and educational objectives. To address this issue, this paper proposes an Artificial Intelligence-driven recommender system designed to deliver personalized learning experiences through collaborative filtering techniques. The proposed framework analyzes various learner interactions, such as course ratings, enrollment history, completion records, search behavior, and user feedback, to understand individual preferences and learning patterns. These interactions are transformed into a structured user-resource relationship model that serves as the foundation for generating personalized recommendations. By leveraging both learner similarity and resource similarity analysis, the system identifies relevant educational content and produces ranked suggestions tailored to each user's needs. The platform is implemented as a web-based application using Python, Django, and a relational database system, providing an intuitive interface for learners to explore and engage with recommended content. The proposed architecture aims to reduce information overload, enhance learner satisfaction, and support adaptive learning by presenting resources that are more relevant to individual educational journeys. Functional evaluation demonstrates the system's ability to collect interaction data, continuously update recommendation results, and provide alternative suggestions when limited user history is available. The study also examines practical challenges commonly associated with recommendation systems, including sparse interaction data, new-user initialization problems, privacy concerns, and scalability requirements. Furthermore, potential future enhancements involving hybrid recommendation strategies, deep learning models, and intelligent learning path generation are discussed to improve recommendation quality and long-term system performance. The results indicate that personalized recommendation technologies can play a significant role in improving learner engagement, content discovery, and overall effectiveness within modern e-learning environments.

Keywords- E-Learning, Personalized Learning, Recommender Systems, Collaborative Filtering, Artificial Intelligence, Machine Learning, User-Item Interaction Matrix, Django Framework, Educational Technology, Learning Analytics.

I. INTRODUCTION

The rapid advancement of internet technologies and digital education platforms has transformed the way individuals acquire knowledge and develop new skills. E-learning environments provide learners with convenient access to a vast collection of educational resources, including online courses, video lectures, quizzes, assignments, tutorials, and interactive learning materials. These platforms enable flexible and self-paced learning, allowing students, professionals, and lifelong learners to access educational content anytime and from virtually any location. As a result, e-learning has become an essential component of modern education, supporting both academic and professional development across diverse fields of study. Despite the significant benefits offered by online learning platforms, the enormous volume of available content often creates a new challenge known as information overload. Learners are frequently presented with thousands of courses and learning resources, making it difficult to identify materials that align with their individual interests, learning objectives, skill levels, and career aspirations. New users may struggle to determine where to begin their learning journey, while experienced learners may receive generic recommendations that fail to reflect their evolving preferences and educational progress. Consequently, learners may spend excessive time searching for appropriate content rather than engaging in meaningful learning activities. This issue can reduce user satisfaction, decrease platform engagement, and negatively affect learning outcomes.

To address these challenges, intelligent recommendation systems have become a critical component of modern digital learning environments. Recommendation systems analyze user behavior and historical interactions to predict which resources are most likely to be relevant to a particular learner. By providing personalized suggestions, these systems help users discover useful content more efficiently, reduce search effort, and improve overall learning experiences. Among the various recommendation techniques, Collaborative Filtering has emerged as one of the most effective and widely adopted approaches. Instead of relying solely on predefined categories or manually assigned metadata, collaborative filtering identifies hidden relationships among users and learning resources by examining patterns of interaction within the platform.

In educational environments, collaborative filtering can utilize multiple forms of learner activity, including course enrollments, ratings, watch history, quiz performance, completion status, search behavior, and feedback submissions. By analyzing these interactions, the system can identify learners with similar interests and recommend resources that have been beneficial to users with comparable learning patterns. Likewise, it can discover relationships among educational resources and suggest content that complements previously completed courses. Such personalized recommendations support adaptive learning experiences and encourage continuous learner engagement.

The increasing availability of educational data and advances in Artificial Intelligence (AI) and Machine Learning (ML) have further enhanced the capabilities of recommendation systems. AI-driven recommendation engines can process large volumes of interaction data, identify complex behavioral patterns, and generate more accurate suggestions than traditional search-based approaches. These technologies play an important role in improving content discovery, promoting learner retention, and supporting individualized educational pathways. However, challenges such as sparse interaction data, cold-start users, privacy concerns, and scalability continue to affect the performance of recommendation systems and require ongoing research and innovation.

To address these challenges, this paper proposes an AI-based recommender system for personalized e-learning platforms using collaborative filtering techniques. The proposed framework integrates a user-friendly Django-based web interface, a recommendation engine, a user-item interaction matrix, and dedicated course and feedback databases to generate personalized learning recommendations. The system collects and analyzes learner interactions, computes similarities among users and educational resources, and delivers ranked course suggestions tailored to individual preferences. The primary objective is to improve content relevance, enhance learner engagement, and simplify the process of discovering suitable educational materials within large-scale e-learning environments.

The proposed architecture demonstrates how collaborative filtering can be effectively implemented within a practical web-based learning platform to support personalized education. By combining intelligent recommendation mechanisms with modern web technologies, the system contributes toward creating a more adaptive, efficient, and learner-centered educational experience. Furthermore, the framework provides a foundation for future enhancements involving hybrid recommendation models, deep learning techniques, explainable recommendations, and adaptive learning path generation, enabling the development of next-generation intelligent e-learning systems.

II. RELATED WORK

A. Recommender Systems

Adomavicius and Tuzhilin described the major classes of recommender systems, including collaborative filtering, content-based filtering, and hybrid methods, and identified personalization as a key direction for future systems [1]. Sarwar et al. proposed item-based collaborative filtering, which improved recommendation scalability by comparing items rather than repeatedly comparing every active user with all other users [2]. Herlocker et al. later showed that evaluation design choices strongly influence the measured quality of collaborative filtering systems [3].

B. Educational Recommendations

Recommender systems for learning differ from commercial recommenders because learning quality, learner progress, course difficulty, and educational goals must be considered. Manouselis et al. organized technology-enhanced learning recommendation scenarios such as recommending resources, peers, activities, and learning paths [5]. Bobadilla et al. surveyed accuracy, robustness, scalability, and user trust issues in recommender systems, which are directly relevant to large e-learning platforms [4].

C. Need for Personalization

Static course catalogs do not adapt to learner behavior. Adaptive educational systems use user models to personalize content, while collaborative filtering uses community behavior to discover hidden relationships between learners and resources. Matrix factorization and neighborhood-based methods have both been applied to sparse rating data, but practical e-learning systems must also handle cold-start users, changing interests, and privacy constraints [7], [8].

III. PROPOSED SYSTEM

A. Problem Statement

Many online learning platforms provide large content collections but do not guide each learner effectively. Traditional search and category-based browsing fail when students are unsure about the right course sequence. Without personalization, learners may miss useful materials, lose motivation, or discontinue courses. The proposed system addresses this gap by recommending resources based on similarities between learners and learning materials.

B. Architecture

The proposed architecture consists of a learner interface, authentication service, recommendation engine, user-item matrix, course database, feedback database, and recommendation display module. The learner interacts with the platform through a web interface. Ratings, enrollments, and activity logs are stored and converted into structured interaction data. The recommendation engine computes similarity and ranks candidate courses.

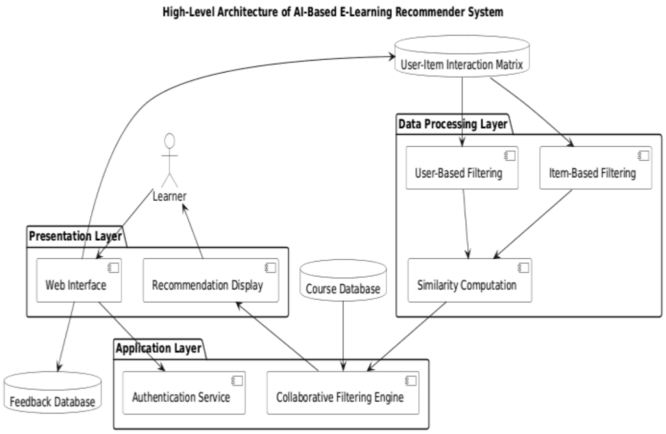

Fig. 1. High-level architecture of the proposed e-learning recommender system.

TABLE I. HARDWARE AND SOFTWARE REQUIREMENTS

Component	Specification / Tool
Processor	Intel i3 5th gen or equivalent
Memory	4 GB RAM minimum
Storage	500 GB available storage
Language	Python 3.10
Framework	Django web framework
Front end	HTML5, CSS3, JavaScript
Database	SQL / MySQL
Domain	Machine learning recommender system

C. Collaborative Filtering Model

The system supports both user-based and item-based collaborative filtering techniques to generate personalized learning recommendations. User-based collaborative filtering analyzes the interaction history of learners and identifies users with similar preferences, behaviors, and learning patterns. Recommendations are then generated by suggesting resources that have been positively received by learners with comparable interests. In contrast, item-based collaborative filtering focuses on the relationships between learning resources and recommends courses that are similar to those a learner has previously enrolled in, completed, or rated highly. Similarity measurements are calculated using techniques such as cosine similarity and Pearson correlation, which help determine the degree of association between users or learning materials. Based on these similarity values, the system predicts preference scores for resources that the learner has not yet accessed. The predicted scores are then ranked to create a prioritized list of recommendations. To further improve recommendation quality, the system continuously updates its interaction data as learners engage with new courses, submit feedback, complete assessments, or modify their learning preferences. This dynamic updating process enables the recommendation engine to adapt to changing interests and learning goals over time. Additionally, when sufficient historical data is unavailable, such as in the case of new learners, the system can provide alternative recommendations based on popular, highly rated, or recently added courses. By combining multiple collaborative filtering strategies, the proposed framework enhances recommendation accuracy, improves content discovery, and supports a more personalized and engaging e-learning experience.with each learner's goals and skill levels.

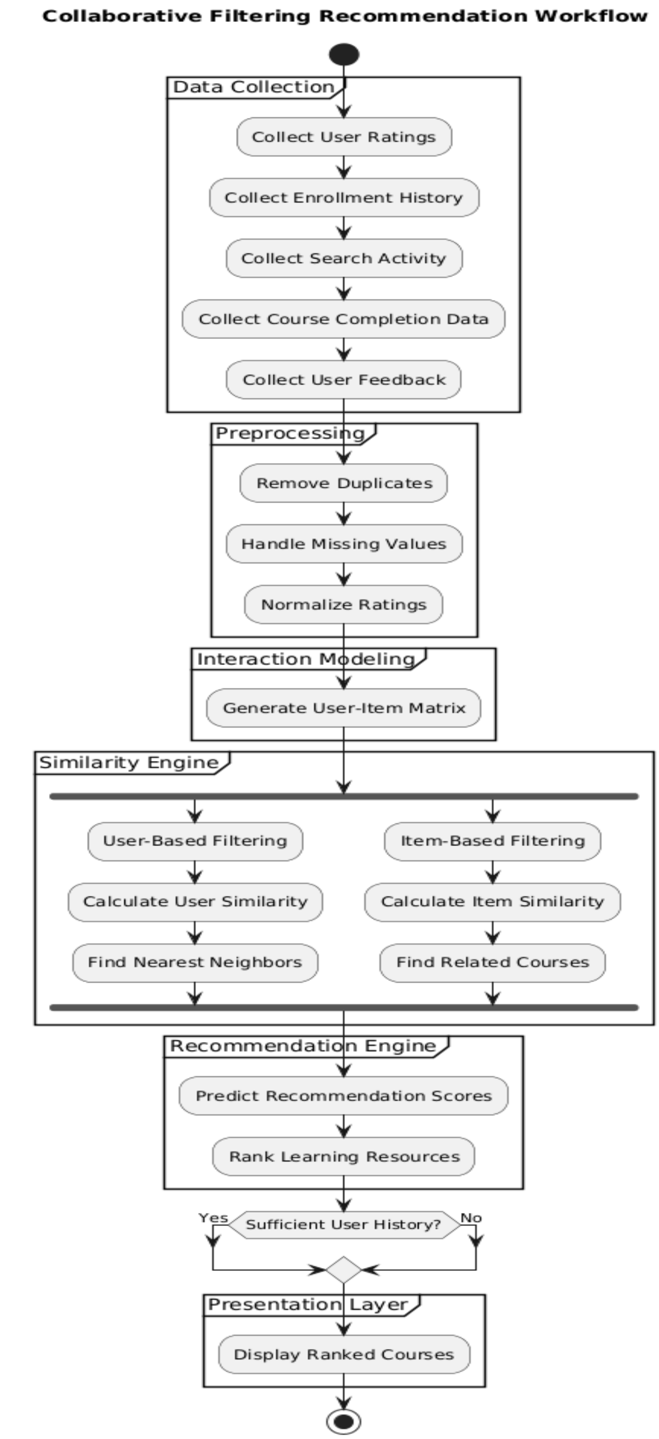

Fig. 2. Collaborative filtering workflow used in the proposed system.

IV. METHODOLOGY AND IMPLEMENTATION

A. Data Collection and Preprocessing

The first stage collects user interactions such as course ratings, enrollment history, search actions, course completion status, and feedback. Raw data is cleaned by removing duplicates, handling missing values, and normalizing rating values. The processed data is converted into a user-item matrix in which each row represents a learner and each column represents a learning resource.

B. Recommendation Generation

After preprocessing, similarity values are computed between users or between items. For a target learner, the system selects neighbors with high similarity and predicts scores for unseen courses. Items with higher predicted scores are recommended first. When a new learner does not have enough history, the system displays popular or recently added courses as a fallback until personal data becomes available.

C. Web Application

The application is implemented using Python with a Django-based backend, HTML5, CSS3, and JavaScript for the front end, and SQL/MySQL for persistent storage. The system provides login, search, course listing, rating, feedback, and recommendation display functions. The modular design allows the recommendation engine to be upgraded later with hybrid filtering or matrix factorization without rebuilding the entire platform.

D. Results

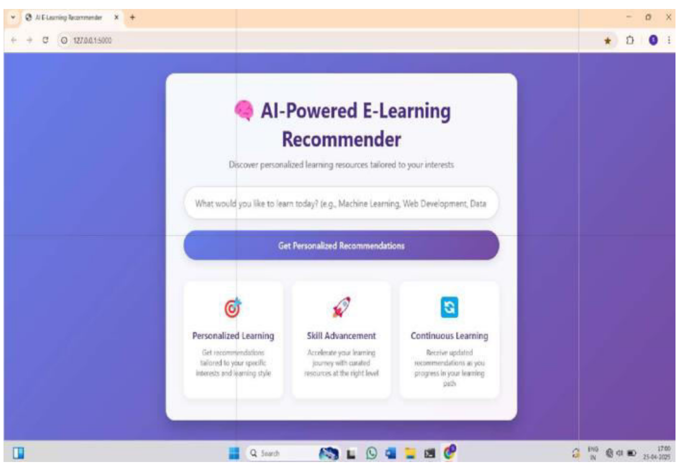

Fig. 3. Home Page of the E-Learning Recommender System

Description:
The figure illustrates the home page of the proposed AI-based e-learning recommender system. This interface serves as the primary access point for learners, providing navigation to course catalogs, search functionality, user profiles, and recommendation services. The dashboard is designed with a clean and user-friendly layout that enables learners to explore educational resources efficiently. Through this interface, users can access personalized learning features, view available courses, track learning progress, and interact with the recommendation engine. The home page acts as the central hub connecting learners with the platform's personalized learning environment.

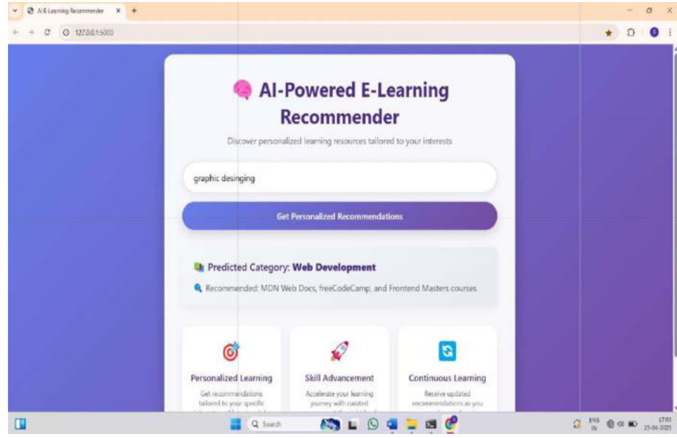

Fig. 4. Course Search and Recommendation Request Interface

Description:
The figure presents the search interface where learners can browse educational resources and request personalized recommendations. Users can search for courses based on topics, interests, skill levels, or learning objectives. The system records search activities, enrollment behavior, ratings, and interaction history to enhance recommendation quality. This interface serves as the data collection layer of the platform, enabling the collaborative filtering engine to analyze learner preferences and generate relevant course suggestions tailored to individual learning needs.

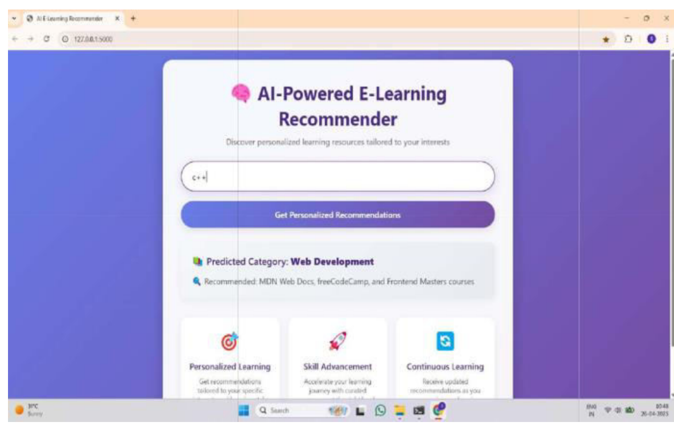

Fig. 5. Personalized Course Recommendation Results

Description:
The figure displays the recommendation results generated by the collaborative filtering engine. Based on user interactions, enrollment history, ratings, feedback, and similarity analysis, the system provides a ranked list of recommended courses that match the learner's interests and learning patterns. The results page presents recommended resources in an organized format, allowing users to easily compare and select suitable courses. The interface helps reduce information overload by prioritizing the most relevant educational content and supporting a more personalized and efficient learning experience.

V. TESTING AND DISCUSSION

The prototype was evaluated through functional testing. The tests verified whether the system can accept learner activity, store interactions, build the matrix, generate similar-user or similar-item recommendations, and display ranked resources. The testing does not claim educational outcome improvement; it verifies system behavior and technical feasibility.

TABLE II. REPRESENTATIVE FUNCTIONAL TEST CASES

ID	Scenario	Expected Output	Result
T1	New login	User dashboard loads	Pass
T2	Ratings entered	Matrix updates	Pass
T3	Similar users	Relevant courses ranked	Pass
T4	Course search	Matching courses listed	Pass
T5	Feedback update	Suggestions improve	Pass
T6	Empty history	Popular fallback shown	Pass

The major limitations are cold-start, sparse ratings, popularity bias, and privacy concerns. Cold-start occurs when a new user or new course has no interaction history. Sparse ratings reduce the reliability of similarity calculations. Privacy controls are necessary because learning behavior can reveal personal academic interests and weaknesses. These issues can be reduced through hybrid recommendation, secure data handling, explainable recommendations, and periodic model evaluation.

VI. CONCLUSION AND FUTURE WORK

This paper presented an AI-based recommender system for personalized e-learning platforms using collaborative filtering. The proposed design analyzes learner interactions and generates course suggestions using user-based and item-based similarity. The Django-based web architecture provides a practical student-level implementation for reducing content overload and improving learner engagement.

Future work can add hybrid filtering, deep learning embeddings, adaptive learning paths, multilingual course support, real-time analytics, explainable recommendations, and privacy-preserving learning analytics. With these improvements, the system can become a more robust learning assistant for digital education platforms.

REFERENCES

[1] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 6, pp. 734–749, 2005.

[2] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," in *Proceedings of the 10th International Conference on World Wide Web*, 2001, pp. 285–295.

[3] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl, "Evaluating collaborative filtering recommender systems," *ACM Transactions on Information Systems*, vol. 22, no. 1, pp. 5–53, 2004.

[4] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutierrez, "Recommender systems survey," *Knowledge-Based Systems*, vol. 46, pp. 109–132, 2013.

[5] N. Manouselis, H. Drachsler, K. Verbert, and E. Duval, *Recommender Systems for Learning*. Springer, 2013.

[6] P. Brusilovsky and E. Millan, "User models for adaptive hypermedia and adaptive educational systems," in *The Adaptive Web*, Springer, 2007.

[7] F. Ricci, L. Rokach, and B. Shapira, *Recommender Systems Handbook*, 3rd ed. Springer, 2022.

[8] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, 2009.

[9] J. S. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," in *Proceedings of UAI*, 1998.

[10] M. J. Pazzani and D. Billsus, "Content-based recommendation systems," in *The Adaptive Web*, Springer, 2007.

[11] D. Goldberg, D. Nichols, B. M. Oki, and D. Terry, "Using collaborative filtering to weave an information tapestry," *Communications of the ACM*, vol. 35, no. 12, pp. 61–70, 1992.

[12] X. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," *Advances in Artificial Intelligence*, vol. 2009.

[13] S. Rendle, "Factorization machines," in *IEEE International Conference on Data Mining*, 2010.

[14] C. Desrosiers and G. Karypis, "A comprehensive survey of neighborhood-based recommendation methods," in *Recommender Systems Handbook*, Springer, 2011.

[15] K. Verbert, N. Manouselis, H. Drachsler, and E. Duval, "Dataset-driven research for improving recommender systems in learning environments," *Educational Technology & Society*, vol. 15, no. 3, pp. 205–216, 2012.

[16] D. B. Leake and E. Plaza, *Case-Based Reasoning Research and Development*. Springer, 2003.

[17] J. Ben Schafer, D. Frankowski, J. Herlocker, and S. Sen, "Collaborative filtering recommender systems," in *The Adaptive Web*, Springer, 2007.

[18] C. C. Aggarwal, *Recommender Systems: The Textbook*. Springer, 2016.

[19] A. Gunawardana and G. Shani, "A survey of accuracy evaluation metrics of recommendation tasks," *Journal of Machine Learning Research*, vol. 10, pp. 2935–2962, 2009.

[20] M. D. Ekstrand, J. T. Riedl, and J. A. Konstan, *Collaborative Filtering Recommender Systems*. Foundations and Trends in Human-Computer Interaction, 2011.

[21] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," arXiv:1301.3781, 2013.

[22] J. Pennington, R. Socher, and C. Manning, "GloVe: Global vectors for word representation," in *EMNLP*, 2014.

[23] A. Vaswani et al., "Attention Is All You Need," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[24] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *NAACL-HLT*, 2019.

[25] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.
[26] F. Chollet, *Deep Learning with Python*, 2nd ed. Manning Publications, 2021.
[27] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 4th ed. Pearson, 2021.
[28] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. Cambridge University Press, 2008.
[29] K. Murphy, *Machine Learning: A Probabilistic Perspective*. MIT Press, 2012.
[30] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of Machine Learning*, 2nd ed. MIT Press, 2018.
[31] Django Software Foundation, *Django Documentation*. Available: https://docs.djangoproject.com/
[32] Oracle Corporation, *MySQL Reference Manual*. Available: https://dev.mysql.com/doc/
[33] T. Anderson, *The Theory and Practice of Online Learning*, 3rd ed. Athabasca University Press, 2020.
[34] G. Siemens, "Learning analytics: The emergence of a discipline," *American Behavioral Scientist*, vol. 57, no. 10, pp. 1380–1400, 2013.
[35] R. Ferguson, "Learning analytics: Drivers, developments and challenges," *International Journal of Technology Enhanced Learning*, vol. 4, no. 5, pp. 304–317, 2012.