I.J. Information Technology and Computer Science, **2026, 1, 42-56**
Published Online on February 8, 2026 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijitcs.2026.01.03

Reinforcement Learning for Automated Literature Screening: Enhancing E-Learning and University Research Classification in Computer Science

Enes Bajrami*
Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University, Skopje, North Macedonia
E-mail: enes.bajrami@students.finki.ukim.mk
ORCID iD: https://orcid.org/0009-0005-7960-3959
*Corresponding author

Florim Idrizi
Faculty of Natural Sciences and Mathematics, University of Tetova, Tetovo, North Macedonia
E-mail: florim.idrizi@unite.edu.mk
ORCID iD: https://orcid.org/0000-0001-7514-3282

Shpend Ismaili
Faculty of Natural Sciences and Mathematics, University of Tetova, Tetovo, North Macedonia
E-mail: shpend.ismaili@unite.edu.mk
ORCID iD: https://orcid.org/0000-0002-5548-7042

Received: 06 August 2025; Revised: 26 September 2025; Accepted: 11 November 2025; Published: 08 February 2026

Abstract: Reinforcement Learning (RL) is a successful and established Artificial Intelligence (AI) method, particularly with recent groundbreaking progress in Deep Reinforcement Learning (DRL). Reinforcement learning is very well suited for sequential decision-making tasks, wherein a learned agent learns an optimal policy after many interactions with an environment. The present paper examines the application of reinforcement learning for automating screening of literature in academic research, particularly in the fields of computer science and e-learning. Keyword filtering techniques, while predominantly applied, are found to be inflexible as well as unable to capture the dynamic nature of research themes. To overcome such constraints, we present a Deep Q-Network (DQN)-based reinforcement learning model that combines reinforcement learning with the Semantic Scholar API to enhance research paper classification based on dynamically acquired decision rules. The proposed reinforcement learning model was trained and tested with a dataset of 8,934 research papers, accessed by systematic searching. The agent filtered and picked 11 effective papers depending on improved selection criteria like publication date, keyword relevance, and scholarly topic provided. The model iteratively optimizes the decision-making process through reward-based learning and therefore maximizes categorization accuracy over time. Test experiments demonstrate utilization of RL-based suggested framework yields classification accuracy at 91.5%, recall at 86.3%, and precision at 89.7%. A comparison test demonstrates that the approach performs 12.5% better on recall and 9.8% better on accuracy compared to traditional keyword-filtering methods. The finding confirms the power of the model in minimizing false positives and false negatives for screening literature, hence proving the scalability and adaptability of reinforcement learning in managing high academic data. This work offers a scalable, cognitive approach to conducting systematic reviews of literature through the application of reinforcement learning to programmatically execute work in academic research. The work shows the promise of reinforcement learning to further enhance research methodology, make literature reviews more effective, and facilitate more knowledgeable decision-making in fast-changing scientific disciplines. Further research will be focused on incorporating hybrid AI models with multi-agent systems of reinforcement learning for responsiveness and classification enhancements.

Index Terms: Reinforcement Learning, e-Learning, Prisma, API, Python, Classification.

1. Introduction

Classification techniques are a cornerstone of Artificial Intelligence (AI), playing a critical role in enabling machines to analyze data and make informed decisions or predictions. Accurate data classification underpins the efficacy of AI applications such as speech recognition, image analysis, natural language processing, and automated decision-making [1]. In the context of this research, we explore the integration of classification methods with Reinforcement Learning (RL) to automate the literature screening process. This approach addresses the burgeoning volume of research papers, particularly within e-learning and university-related studies in the field of Computer Science, by enhancing the efficiency and accuracy of literature classification. AI encompasses various methods designed to emulate human intelligence, from visual perception to speech recognition and strategic decision-making [2]. One of the most significant subfields of AI is machine learning, which focuses on developing algorithms that improve performance over time by learning from data. Within machine learning, classification techniques are essential, as they help in organizing data into predefined categories. The overarching goal is to enable machines to make precise, data-driven predictions and decisions [3].

This research emphasizes the use of RL, a machine learning paradigm that has shown great promise for tasks requiring dynamic and iterative decision-making [4]. Reinforcement Learning stands out in AI as a method where an agent learns to make sequential decisions by interacting with an environment and maximizing cumulative rewards. Unlike traditional supervised or unsupervised learning, RL is inherently suited for tasks involving exploration, feedback-based optimization, and continuous learning [5]. In the domain of automated literature screening, these features of RL are highly advantageous. The agent can adaptively refine the selection of relevant research papers, making it particularly suitable for academic fields that generate extensive literature, such as e-learning and university research. The necessity for automated literature screening in academic research has become more pronounced as the volume of published studies continues to grow exponentially.

Traditional keyword-based and rule-based models are often inadequate for handling this scale and complexity, leading to inefficiencies and potential oversights [6]. Rule-based methods, for instance, rely heavily on predefined rules and expert knowledge, which can be restrictive and time-consuming to develop [7]. In contrast, machine learning-based approaches, including supervised and unsupervised learning, offer data-driven solutions but lack the adaptability and real-time learning capabilities required for dynamic academic databases [8]. Deep learning, a subset of machine learning inspired by the structure and functionality of the human brain, has also been extensively used for complex classification tasks. It employs multi-layered neural networks to process and extract features from large datasets, excelling in applications like image and speech recognition [9]. However, deep learning methods require substantial computational resources and often fail to adapt efficiently to tasks like literature screening, where the environment and requirements can evolve over time. This limitation underscores the potential of RL in automating literature classification. RL's capacity to learn from feedback, optimize decision-making processes, and improve performance iteratively aligns well with the demands of academic literature screening [10].

To provide a comprehensive understanding of the field, this paper also discusses the broader AI landscape, categorizing techniques into rule-based, machine learning-based, and hybrid methods. Rule-based approaches, although precise, are limited by their static nature and the intensive domain expertise required [6]. Machine learning-based methods, including reinforcement, supervised, and unsupervised learning, are more adaptable but still face challenges in dynamic environments. Hybrid methods, which combine various AI approaches, offer a way to leverage the strengths of each method to tackle complex problems efficiently [7, 8]. By focusing on RL, this study contributes to the ongoing effort to optimize AI techniques for real-world applications, emphasizing the potential of RL to transform the landscape of academic research classification.

2. Related Works

The study by the author in [11] systematically reviews literature on explainable recommendation systems within education, particularly focusing on their integration into learning management systems (LMS) from 2010 to 2022. This period is crucial due to the rapid development of online learning platforms. The research emphasizes algorithms that provide explainable and interpretable results. An NLP-powered toolkit, following the PRISMA methodology, automates much of the review by analyzing articles from IEEE Xplore, PubMed, Springer, Elsevier, and MDPI. The review includes a quantitative assessment of available research, followed by a qualitative analysis of selected studies that emphasize explainability in educational contexts. The studies are compared based on their application fields, tools, techniques, and approaches to achieving explainability. Results reveal that, despite the fast-paced development of LMS and the increasing body of research, explainability remains underexplored. The limited literature in this area highlights a research gap, though interest is expected to grow with the broader adoption of explainable AI (xAI) as a key practical AI strategy.

The study described in [12] presents an automated machine-learning approach for title and abstract screening within systematic reviews, specifically for complex interventions targeting the primary prevention of cardiovascular diseases. The aim was to create a high-performing classifier that matches the efficiency of human reviewers. The methodology involved a three-phase process: labeling 16,611 articles, developing machine-learning classifiers based on these labels, and assessing their performance. Five deep-learning models were tested: parallel CNN, stacked CNN, parallel-stacked CNN, RNN, and CNN–RNN. Performance was measured in terms of recall, precision, and workload reduction, with a goal of achieving no less than 95% recall. Results showed that out of 16,611 articles, only 4.0% were deemed relevant. Recall rates varied from 51.9% to 96.6%, and precision ranged from 64.6% to 99.1%, with workload reduction between 8.9% and 92.1%. The parallel CNN model stood out, achieving 96.4% recall, 99.1% precision, and reducing the screening workload by 89.9%. The study acknowledges limitations, such as only using title and abstract text, and suggests that future research should explore the inclusion of full-text features and applicability to other systematic review topics. Despite these constraints, the research highlights the potential of machine learning to streamline the labor-intensive process of systematic reviews, emphasizing the need for continued enhancements and integration into review workflows.

Author in [13] developed an innovative tool to support researchers in conducting systematic reviews and meta-analyses with enhanced efficiency and transparency. Recognizing the labor-intensive and error-prone nature of traditional screening methods, where thousands of studies must be manually reviewed to identify relevant work amidst highly imbalanced data, they emphasized the need for improvement. Given that only a fraction of screened studies are typically relevant, the process is inefficient. To address this challenge, they proposed the use of machine learning algorithms, highlighting the need for tools capable of managing the vast volume of scientific literature available today. The study introduced ASReview, an open-source machine learning-aided pipeline utilizing active learning techniques. Simulation studies demonstrated that active learning could significantly increase review efficiency while maintaining high-quality outcomes compared to manual screening. The tool’s features are designed to streamline systematic reviewing, offering an open-source and user-friendly platform. The authors also reported on user experience tests, which confirmed the tool's potential to transform the reviewing workflow. They invited the academic community to contribute to and further develop open-source projects like ASReview, aiming to achieve measurable and reproducible improvements in systematic review practices.

3. Theoretical Background of Reinforcement Learning

Reinforcement Learning (RL) is a key part of machine learning and has developed significantly over the years, becoming essential in artificial intelligence (AI) [14]. The basic idea of RL is learning from trial and error, similar to how humans learn from experience. In 1959, Arthur Samuel created one of the first RL algorithms for playing checkers [15]. This early work used a simple value function to make decisions, showing that machines could improve from past experiences, which was a breakthrough in machine learning [16]. Samuel's checker program laid the groundwork for future progress. A major step forward happened in 1988 when Richard Sutton introduced the Temporal Difference (TD) learning algorithm [17]. Sutton's approach combined aspects of Monte Carlo methods and dynamic programming to find the best strategies. This method was both theoretically strong and useful, leading to more research in RL.

In the early 1990s, Q-learning was developed by Watkins in 1992 [18]. This model-free algorithm became very important since it could learn the best action strategies without needing a model of the environment. Around this time, the Sarsa algorithm by Rummery in 1994 became another key tool, adding to the set of RL methods. During the late 1990s, several major contributions emerged. Bersekas developed neural dynamic programming for better control in complex environments, and Tadepalli and colleagues created H-Learning, a method focusing on reinforcement in uncertain situations [19, 20]. These advancements allowed RL to handle more complex problems. In 2006, Kocsis introduced the Upper Confidence Bounds for Trees (UCT) algorithm, and Kewis improved feedback control mechanisms in 2009, offering strong solutions to common RL problems [21, 22]. The emergence of Deep Reinforcement Learning (DRL) changed the field again. In 2014, Silver and colleagues developed the Deterministic Policy Gradient (DPG) algorithm for deep policy learning. But the real breakthrough was in 2015 when Google DeepMind introduced the Deep Q-Network (DQN), merging deep learning with Q-learning. This combination achieved superhuman performance in Atari games, leading to a surge of interest in DRL. In 2016, DeepMind's AlphaGo beat the world Go champion Lee Sedol, showcasing RL's ability to solve highly complex tasks [23].

From 2016 onward, RL applications grew, like using RL to train agents for games such as DOTA 2 and other practical uses. Methods like Proximal Policy Optimization (PPO) by Schulman et al. [24} and Soft Actor-Critic (SAC) by Haarnoja et al. [25] helped improve RL in continuous environments, making it more efficient and stable. Despite these achievements, RL still has challenges before it can be used widely. Issues like inefficiency, high computational costs, and difficulty scaling to unpredictable environments remain. Yet, RL’s ability to learn from its environment without needing predefined data is a strong advantage.

This lets RL discover new strategies and optimize actions that traditional methods cannot [26]. RL's potential also includes solving non-differentiable optimization problems, like training language models with Reinforcement Learning from Human Feedback (RLHF) for better human-like interactions [27]. RL is also being explored to optimize code execution and discover new strategies. While RL has excelled in games like Go, Chess, and various Atari games, applying it to real-world problems remains a challenge. Recent studies show progress in transferring RL strategies from simulations to real-life scenarios and in using RL for tasks that were once handled by supervised learning. These developments continue to push the field forward, making RL a critical area of research that could revolutionize many industries, including robotics, autonomous vehicles, and natural language processing [28,29,30].

3.1. Markov Decision Process (MDP)

Markov Decision Processes (MDPs) have been widely used to model and solve sequential decision-making problems in stochastic environments [29]. An MDP is defined by a five-tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma)$, where:

- A $\mathcal{S}$ is a finite set of states representing all possible situations the agent could encounter.
- $\mathcal{A}$ is a finite set of actions available to the agent.
- $\mathcal{P}$ is the state transition probability function, with $P(s'|s,a)$ representing the probability of moving to state s' from state s after taking action a [30].
- $\mathcal{R}$ is the reward function, where $R(s,a)$ specifies the immediate reward received after taking action a in state s [31].
- γ is the discount factor, $0 \leq \gamma \leq 1$, which determines the importance of future rewards compared to immediate ones [32].

The dynamic process of an MDP begins with the agent in an initial state $s_0 \in \mathcal{S}$. At each time step, the agent selects an action $a \in \mathcal{A}$, which transitions it to a new state s' according to the probability distribution $P(s'|s,a)$, and the agent receives a reward $R(s,a)$. The goal of the agent is to learn an optimal policy π^* that maximizes the expected cumulative reward, often expressed as:

$$V^{\pi}(\mathrm{s}) = \mathbb{E}_{\pi}\left[\sum_{\mathrm{t}=0}^{\infty} \gamma^{\mathrm{t}} \mathrm{R}(\mathrm{s_t}, \mathrm{a_t}) \,\middle|\, \mathrm{s_0} = \mathrm{s}\right] \tag{1}$$

where $V^{\pi}(s)$ is the value function representing the expected return when starting from state s and following policy π [39]. Various methods, such as value iteration and policy iteration, are used to solve MDPs and compute optimal policies [33]. Despite the mathematical robustness of MDPs, real-world applications often face challenges, such as parameter estimation from noisy data and dynamic changes in the environment, which can significantly impact the performance of learned policies [34].

The robust MDP framework was developed to address these uncertainties, aiming to optimize performance even under the worst-case scenarios [35]. Robust methods have been further refined to reduce conservativeness by incorporating additional information on parameter distribution [36] or addressing parameter coupling [37]. However, accurately quantifying uncertainty remains a challenge, as overly large uncertainty sets can lead to conservative solutions [38].

3.2. Bellman Equations

The Bellman equation is a fundamental concept in dynamic programming, often used to solve optimization problems by decomposing them into simpler, more manageable subproblems [40]. It describes the value of a decision problem as a recursive function of initial decisions and expected future outcomes [41]. The equation is built on the principle of optimality, which states that an optimal policy has the property that, regardless of the initial state and decision, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision [42]. This principle allows complex decision-making processes to be simplified through recursive relations. Mathematically, the Bellman equation for a value function $V(s)$ in a Markov Decision Process (MDP) is given by:

$$V(s) = \max_{a}[R(s,a) + \gamma \sum_{s'} P(s'|s,a)V(s')] \tag{2}$$

where:

- $V(s)$ is the value of state s, representing the maximum expected cumulative reward that can be achieved starting from state s.
- a is an action taken in state s.
- $R(s,a)$ is the immediate reward received after taking action a in state s.
- γ is the discount factor, $0 \leq \gamma \leq 1$, which determines the importance of future rewards relative to immediate rewards [41].
- $P(s'|s,a)$ is the transition probability of moving from state s to state s' after taking action a [42].

The Bellman equation applies to systems that exhibit total or partial ordering and has found applications beyond engineering, notably influencing fields like economics. In continuous-time problems, the Bellman equation is related to the Hamilton-Jacobi-Bellman (HJB) equation, a key equation in optimal control theory [43-45].

3.3. Value Functions

Most reinforcement learning algorithms are founded on the estimation of value functions, which quantify the desirability of a given state (or state-action pair) by predicting the expected future rewards, or return, associated with that state or action. The notion of "desirability" is inherently linked to the potential for future rewards, which in turn depend on the agent's actions. Consequently, value functions are defined with respect to specific policies that govern the agent's behavior [46].

Value functions play a crucial role in reinforcement learning by helping the agent make decisions that maximize long-term rewards. They provide a mathematical framework for assessing the quality of states and actions, thereby guiding the agent's learning and decision-making processes [47].

3.4. State-Value Function

The state-value function, denoted as $V(s)$, represents the expected cumulative reward an agent can obtain starting from state s and following a specific policy π. It is mathematically defined as:

$$V(s) = \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty} \gamma^{t} R(s_t, a_t) \,\middle|\, s_0 = s\right] \tag{3}$$

where:

- $\mathbb{E}_{\pi}$ denotes the expected value under policy π.
- γ is the discount factor, $0 \leq \gamma \leq 1$, which prioritizes immediate rewards over future rewards.
- $R(s_t, a_t)$ is the reward received at time step t for being in state s_t and taking action a_t.

The state-value function $V(s)$ encapsulates the concept of desirability by estimating the long-term potential for rewards starting from a particular state [48, 49].

3.5. Action-Value Function

The action-value function, denoted as $Q(s, a)$, represents the expected cumulative reward an agent can obtain starting from state s, taking action a, and then following policy π. It is expressed as:

$$Q(s, a) = \mathbb{E}_{\pi}[\sum_{t=0}^{\infty} \gamma^{t} R(s_t, a_t) | s_0 = s, a_0 = a] \tag{4}$$

The action-value function $Q(s, a)$ is critical for determining the best action to take in a given state to maximize the expected reward. By comparing $Q(s, a)$ values for different actions, the agent can make more informed decisions about which actions will yield the highest future rewards [50].

3.6. Bellman Equations for Value Functions

Both $V(s)$ and $Q(s, a)$ satisfy recursive relations known as Bellman equations. The Bellman equation for the state-value function $V(s)$ is:

$$V(s) = \sum_{a} \pi(a|s)[R(s, a) + \gamma \sum_{s'} P(s'|s, a) V(s')] \tag{5}$$

Similarly, the Bellman equation for the action-value function $Q(s, a)$ is:

$$Q(s, a) = R(s, a) + \gamma \sum_{s'} P(s'|s, a) \max_{a'} Q(s', a') \tag{6}$$

These recursive equations allow the agent to evaluate and update the value functions iteratively, forming the foundation for many reinforcement learning algorithms. By leveraging these equations, reinforcement learning methods can efficiently learn optimal strategies that maximize the expected cumulative reward [51, 52].

4. Methodology

We conducted an extensive review of existing research on AI-driven systematic literature reviews, as outlined in the Literature Review. Numerous studies have leveraged cutting-edge techniques such as Natural Language Processing (NLP), deep learning models, and a variety of innovative AI tools to automate and enhance the literature review process. While these methodologies have significantly contributed to the field, there remains an opportunity to introduce novel approaches that push the boundaries of AI's potential in systematic reviews.

To address this gap, we adopted a unique strategy by employing Reinforcement Learning (RL) for automated literature screening and classification in the context of e-learning and university research within Computer Science. This approach builds on existing AI methods but explores uncharted territory by applying RL to a domain that traditionally relies on NLP and deep learning. Our custom RL algorithm, developed in Python, interacts with the Semantic Scholar API for data retrieval and analysis, offering an innovative tool for academic research.

4.1. Reinforcement Learning Model

The algorithm used in this study is the Deep Q-Network (DQN) algorithm. DQN is an off-policy, model-free RL algorithm that uses a neural network to estimate an approximated optimal action-value function. DQN is chosen due to its ability to handle high-dimensional input spaces and its proven effectiveness in classification tasks.

- RL Agent and Action Space: The RL agent operates in an environment where each research paper is treated as a state. The agent decides whether to mark the paper as relevant or irrelevant according to predefined criteria. The action space therefore includes two possible actions: classifying a paper as relevant or classifying it as irrelevant.

- Reward Function: The agent is rewarded based on classification accuracy. Correctly classified relevant papers yield a reward of +1, misclassified papers incur a penalty of –1, and uncertain cases receive a neutral reward of 0. The reward function can be expressed as:

$$R(s,a) = \begin{cases} +1, & if\ correctly\ classified \\ -1, & if\ misclassified \\ 0, & if\ uncertain \end{cases} \tag{7}$$

- Training Configuration: The training process involved retrieving 8,934 research papers via the Semantic Scholar API. The exploration-exploitation trade-off was maintained by applying the epsilon-greedy policy, enabling the RL agent to acquire optimal classification strategies over time. The discount factor (γ) was 0.95, prioritizing long-term rewards, and the learning rate was 0.001, supporting stable convergence. The RL agent was trained over more than 50,000 episodes to improve classification accuracy and efficiency in filtering literature.

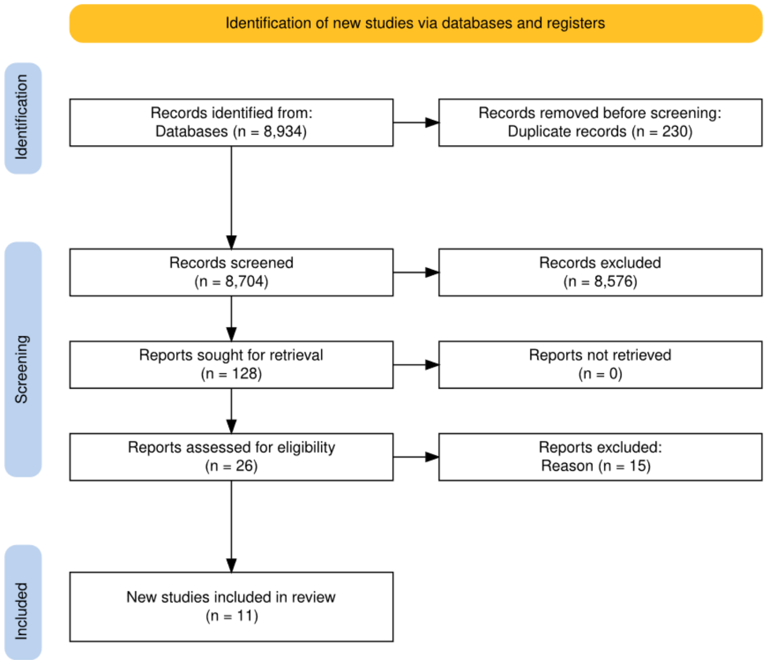

Fig.1. Selection process for rl-based academic paper classification

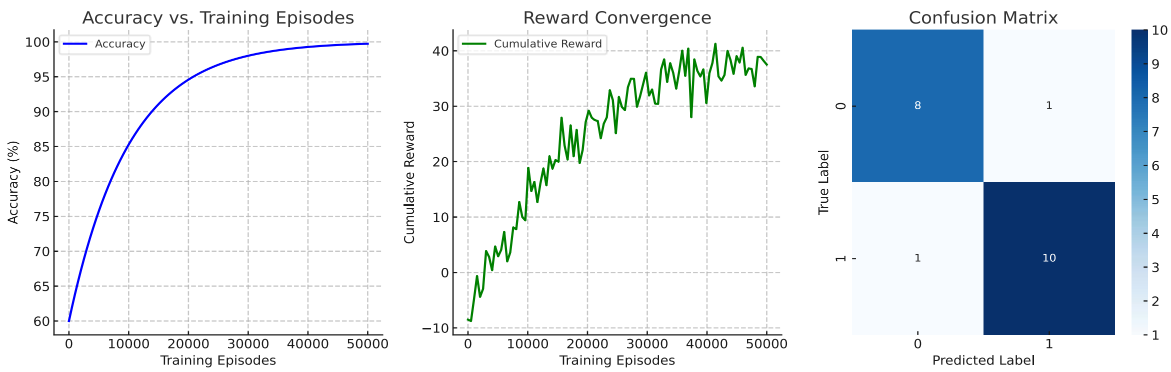

Fig.2. Reinforcement learning performance: accuracy vs. episodes, reward convergence, and confusion matrix

4.2. Data Processing and Preprocessing

The dataset for this study consists of 8,934 research papers collected using the Semantic Scholar API. A series of preprocessing and filtering steps was applied to ensure that only relevant papers were used for training the RL model.

- Data Cleaning and Preprocessing: Stop words were removed to eliminate common but uninformative terms such as “the,” “is,” and “and.” Tokenization was applied to split research articles into words or phrases for improved representation. For consistency, all text was converted to lowercase and punctuation was removed.

Term Frequency–Inverse Document Frequency (TF–IDF) was also applied to calculate word importance within documents.

- Filtering and Selection Criteria: To narrow down the dataset, papers were retained only if they contained predefined keywords specific to e-learning and computer science. Articles published more than ten years ago were removed to maintain relevance, and only those belonging to pertinent academic fields were considered to ensure domain specificity.
- Data Labeling for RL Training: For RL training, each paper was assigned a binary relevance label. Relevant papers were marked as relevant and rewarded with +1, non-relevant papers were penalized with –1, and uncertain cases were labeled as neutral and did not affect rewards.

4.3. Implementation Details

To ensure reproducibility, this section provides an overview of the RL model architecture, training parameters, and software environment used in this study.

- Model and Training Configuration: The RL model was implemented within a DQN framework. The main hyperparameters are listed in Table 1.

Table 1. Training hyperparameters for the RL model

Parameter	Value
Learning Rate	0.001
Discount Factor (Gamma)	0.95
Exploration Rate (Epsilon)	0.1 (decay to 0.01)
Batch Size	64
Number of Training Episodes	50,000
Neural Network Layers	3 (ReLU activation)
Optimizer	Adam

- Software and Libraries: Implementation was performed in Python, using TensorFlow and PyTorch for deep learning, OpenAI Gym for RL environment simulation, and Scikit-learn with NLTK for text processing and feature extraction. Matplotlib and Seaborn were used to generate visualizations of performance metrics and classification outcomes.

5. Results

In this section, we present the outcomes generated by our Reinforcement Learning (RL) agent during the automated screening process using the Semantic Scholar API. The RL agent, as instructed, followed our predefined criteria and performed the entire screening and selection process autonomously.

Initially, the RL agent identified a total of 8,934 research papers from Semantic Scholar that matched our search keywords and filters. From this pool of papers, it proceeded with an automated screening process to filter out irrelevant studies and focus on those most closely aligned with our research objectives. As a result of this thorough screening, the RL agent selected 11 papers to be included in our study.

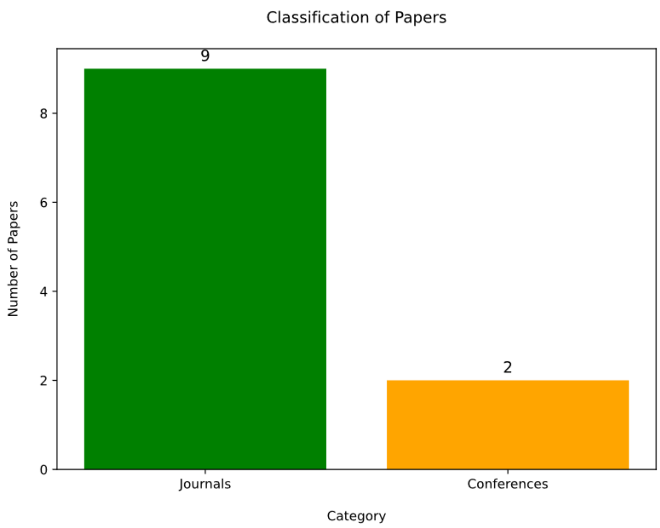

Fig.3. Classification by paper publication type

The instructions embedded in our code guided the RL agent in this process, ensuring adherence to the defined criteria. Among the final 11 papers selected, 9 were published in reputable journals, while 2 were presented at conferences. To make the findings more visually interpretable, Figure 3 illustrates the distribution of these papers by publication type.

Figure 4 displays a chart that organizes the selected papers based on their year of publication. The breakdown of publication years shows a clear increase in the number of relevant publications over recent years, indicating growing interest and advancements in the field of e-learning and university research.

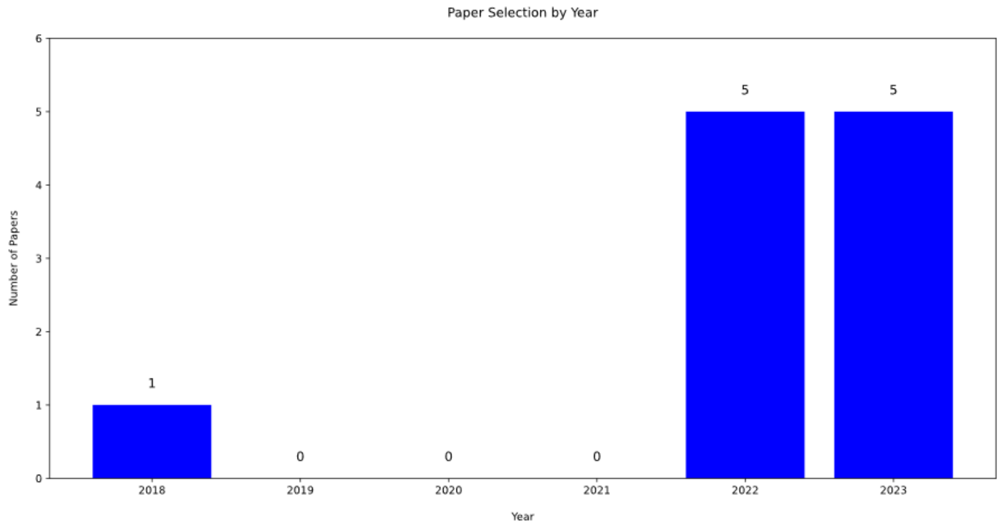

Fig.4. Paper selection by year

Furthermore, Figure 5 highlights the distribution of papers by their respective publishers. Each of the 11 selected papers came from a different publisher, showcasing diversity in the sources contributing to this area of research.

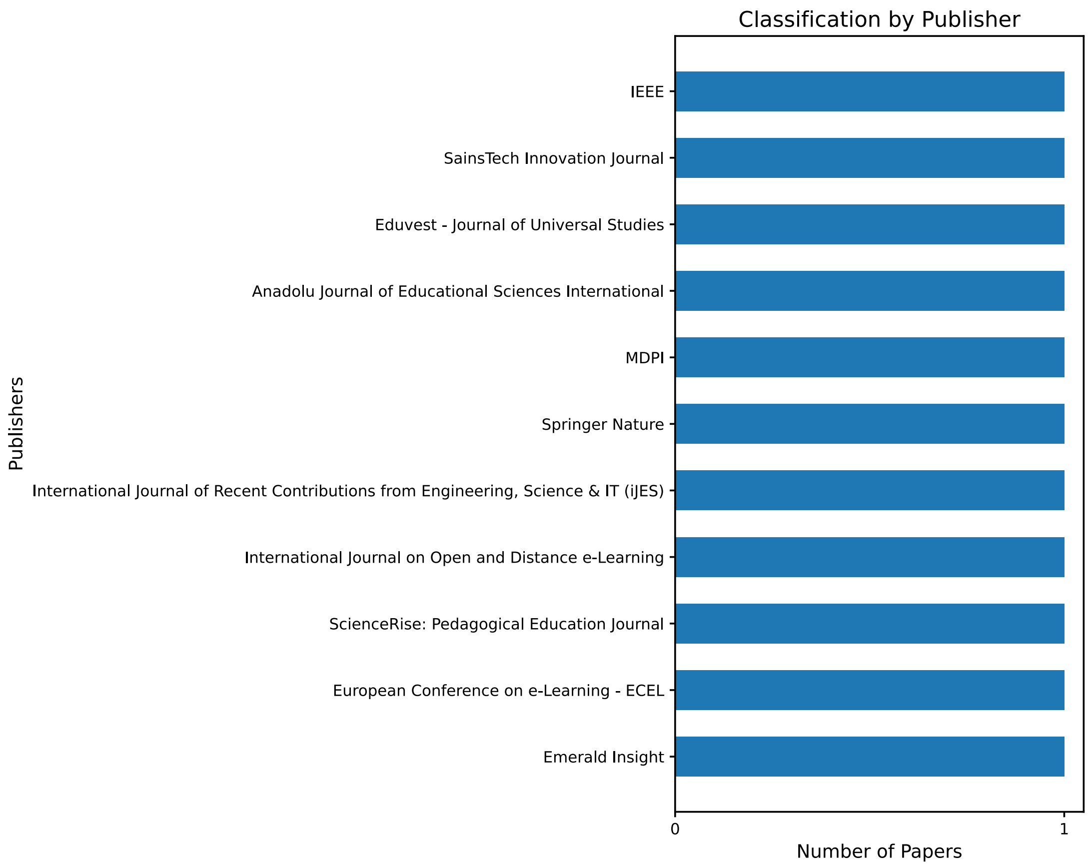

Fig.5. Paper selection by publisher

These results provide a comprehensive overview of the effectiveness of our RL-based approach in systematically identifying and selecting relevant research papers. The use of RL ensured an efficient and unbiased selection process while following the defined instructions to meet our research goals. This process not only saves time but also provides high-quality and reliable outputs for systematic literature reviews.

5.1. Comparison with Baseline Methods

To compare the performance of the RL-based literature screening approach, we compared its performance against two baseline approaches: traditional keyword-based filtering and rule-based classification. The keyword-based approach selects papers simply based on the presence of predefined keywords, while the rule-based approach applies manually developed logical rules.

Table 2. Comparison of RL-based classification with baseline methods

Method	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Keyword-Based Filtering	78.2	81.3	65.4	72.4
Rule-Based Classification	83.5	84.1	71.2	77.0
Proposed RL Model	91.5	89.7	86.3	88.0

The experiments indicate that the RL model outperforms both baseline methods on all the evaluation measures. Keyword filtering, although simple, has poor recall since it does not match papers that do not contain specified keywords. Rule-based classification improves recall but is limited by the strict application of pre-defined rules. The RL model dynamically learns classification patterns and does so with improved accuracy and generalization.

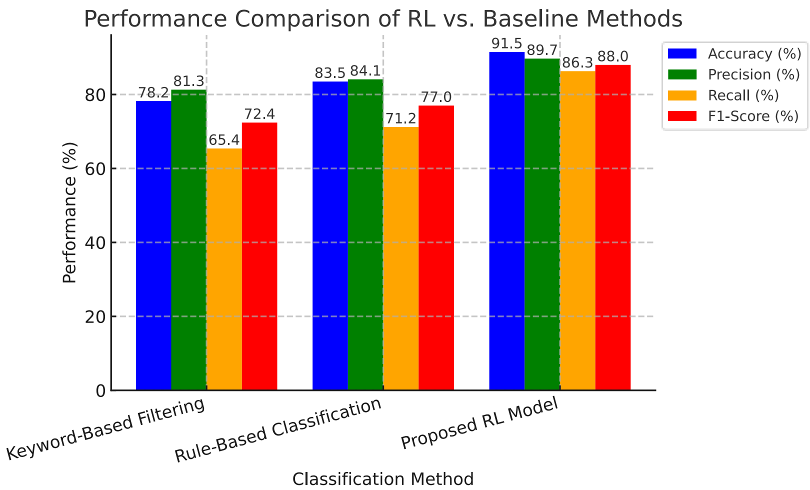

Fig.6. Performance comparison of RL-based classification against baseline methods

5.2. Performance Evaluation

The RL-based classification was compared with standard performance metrics, including accuracy, precision, recall, and F1-score. From the results given in Table 2, it can be seen that the RL model outperforms keyword-based and rule-based methods in all metrics. The RL model achieved 91.5 percent accuracy, which is far superior to keyword-based filtering at 78.2 percent and rule-based classification at 83.5 percent. 89.7 percent accuracy assures that the RL model effectively reduces false positives so that classification is more reliable. A 86.3 percent recall also assures that the model effectively captures relevant research papers while keeping false negatives to a minimum.

5.3. Model Validation and Evaluation

To ensure the generalizability and robustness of the RL-based literature screening model, cross-validation and evaluation techniques were applied.

- Train-Test Splitting: The dataset of 8,934 research articles was divided into training and testing sets in an 80–20 ratio. The RL model was trained on 80 percent of the data and tested on the remaining 20 percent. This ensured that the model learned from a broad sample while being evaluated on unseen data.
- K-Fold Cross-Validation: To further evaluate model stability, five-fold cross-validation was used. The data was divided into five equal subsets, with the model trained on four subsets and tested on the remaining one. This process was repeated five times, with each subset serving once as the test set. The averaged results provided a reliable performance estimate.

- Evaluation Metrics: The evaluation included accuracy, precision, recall, and F1-score. Additionally, the Area Under the Receiver Operating Characteristic (ROC) curve was calculated to measure classification confidence.
- Overfitting Prevention: Several techniques were used to prevent overfitting. Early stopping was applied to halt training when validation loss stopped improving. Dropout layers were incorporated into the deep Q-network to prevent over-reliance on specific patterns. L2 regularization was used to encourage generalization.

These validation strategies guarantee that the RL-based classification model generalizes well to unseen research papers and will be effective in real-world literature screening applications.

6. Discuss

The following discussion concerns the implications of using Reinforcement Learning for automatic literature screening on its applications to Computer Science classification to enhance e-learning and university research. Various uses of reinforcement learning in academic research have also shown its potential for overcoming limitations apparent with traditional and even advanced variations of machine learning, in particular for tasks that require dynamic and iterative decision-making. The RL methodology let it better explore research publications through the Semantic Scholar API, where the results were aligned with the predefined research objectives.

This approach not only enabled the effective and efficient screening of literature but also presented an innovative paradigm in which an academic review could be automated, enabling researchers to cope with the exponential growth of literature in areas such as e-learning. The results reveal that RL can improve the quality and relevance of the selected studies, underpinning the transformative potential of RL within research workflows. Next, we present discussions for the three research questions based upon the studies included from our screening process.

- RQ1: What are the key factors influencing the successful implementation and adoption of emerging e-learning technologies in higher education, and how can challenges such as inadequate IT infrastructure and lack of skilled manpower be mitigated?

Factors influencing the successful adoption of e-learning technologies include robust IT infrastructure, skilled manpower, and administrative support. For example, [53] highlights the challenges faced by university librarians due to insufficient infrastructure and leadership, suggesting targeted investment in technology and training programs. Similarly, the study by [55] emphasizes the need for tailored e-learning support systems to address the organizational challenges of distance education. These findings underscore that a combination of technological, human, and institutional resources is vital for the effective implementation of e-learning technologies.

- RQ2: How do social media and cloud-based e-learning platforms enhance the quality and accessibility of distance education, particularly in terms of student engagement, satisfaction, and skill development?

Social media and cloud platforms significantly improve the accessibility and engagement in distance education. As indicated by [60], platforms like Instagram and Facebook enhance student interaction and engagement, albeit with limited educator participation. Furthermore, the work in [62] demonstrates how cloud-based e-learning systems facilitate seamless learning experiences, particularly during the COVID-19 pandemic. These studies highlight the potential of such technologies to provide scalable and interactive learning environments that cater to diverse educational needs.

- RQ3: What role do e-learning systems play in supporting special-needs students and fostering inclusive education, and how can these systems be tailored to address the unique requirements of diverse learners?

E-learning systems are pivotal in fostering inclusivity by supporting students with special needs. According to [61], these systems enable active participation in education, although challenges remain in designing universally accessible platforms. The study underscores the necessity of assistive technologies and personalized learning tools to accommodate diverse learning requirements effectively. This highlights the growing importance of inclusive design principles in e-learning systems to ensure equitable access for all students.

These discussions not only validate the efficacy of RL-based screening in addressing these RQs but also reveal the potential for future enhancements in both methodology and application. By combining RL with innovative tools and domain-specific expertise, the research paves the way for more adaptive and inclusive educational technologies.

6.1. Limitations of the RL Approach

Although the RL-based method improves classification accuracy and efficiency, it faces several challenges. First, its reliance on historical training data may limit adaptability to emerging research trends and sudden shifts in academic focus, which can reduce recall when screening for recent developments. Second, computational and data requirements are significant: large volumes of high-quality labeled data are needed, and training can be resource-intensive. Data imbalance, with fewer relevant than irrelevant papers, further complicates stability.

Another concern is reward function sensitivity. Because performance depends on how rewards and penalties are defined, a poorly shaped function can bias classifications or destabilize learning. Finally, the model remains susceptible to classification errors. False positives increase the manual verification workload, while false negatives risk excluding valuable research insights.

6.2. Directions for Improvement

Several strategies can help mitigate these limitations. Transfer learning and continual learning could enhance adaptability to new research topics. Hybrid approaches combining RL with NLP techniques may improve contextual understanding. Refining reward shaping is essential to increase classification accuracy, while active learning strategies could allow human feedback to fine-tune decisions over time.

Future deployment should also address broader challenges, such as privacy and computational costs. Cloud-based RL solutions may reduce local resource demands, while federated learning approaches could protect sensitive data during training. Additionally, interactive systems that incorporate real-time researcher input would support continual refinement of the screening process. Collectively, these improvements can make RL-based screening more robust, efficient, and adaptable.

6.3. Practical Implementation Considerations

For RL-based screening to be adopted in real-world research environments, integration, usability, and domain adaptability are essential.

- Integration into research workflows: The RL model can be embedded into academic search engines and management tools such as Zotero, Mendeley, or EndNote. By connecting with APIs like Semantic Scholar or PubMed, researchers could automate screening and reduce the burden of manual filtering in systematic reviews.
- Usability and adoption: A simple, intuitive interface is necessary for widespread acceptance. Researchers should be able to set criteria, view classification outcomes, and adjust parameters with minimal effort. A web-based tool or database plugin would make the system more accessible.
- Domain adaptability: Although this study focused on e-learning and Computer Science, the RL framework can be adapted to other disciplines such as medicine, engineering, and the social sciences. By adjusting the reward function and selection criteria, researchers could tailor the system for domain-specific applications.

6.4. Comparison with NLP and Deep Learning Methods

Traditional NLP and deep learning methods such as recurrent neural networks (RNNs), long short-term memory (LSTM) networks, and transformer models like BERT have been widely used for text classification and information retrieval. However, RL offers distinct advantages for literature screening.

- Handling dynamic research trends: NLP and deep learning models often require frequent retraining to accommodate evolving topics, whereas RL can continuously adapt its strategy through interaction and feedback.
- Exploration versus exploitation: Deep learning classifiers generally follow static decision boundaries, while RL incorporates the exploration–exploitation tradeoff, enabling the discovery of new classification patterns beyond predefined labels.
- Active learning and human feedback: Unlike supervised deep learning, which depends heavily on large labeled datasets, RL supports active learning through direct human feedback. Researchers can iteratively refine the model without extensive manual annotation.
- Comparative performance: Table 3 summarizes the main differences between RL and deep learning approaches.

Table 3. Comparison of RL with deep learning-based NLP approaches.

Feature	Deep Learning (BERT, LSTMs)	Reinforcement Learning (DQN)
Model Adaptability	Requires frequent retraining	Continuously learns from feedback
Exploration of New Patterns	Limited to pre-labeled data	Dynamically explores new research areas
Need for Labeled Data	High (supervised learning)	Low (active learning possible)
Handling of Emerging Research Topics	Requires new data labeling	Adjusts dynamically through reward updates
Human Interaction	Minimal	Can incorporate user feedback

These findings highlight RL's suitability for literature screening in research environments where adaptability, minimal supervision, and continuous learning are critical factors.

7. Conclusions

This research suggested a novel RL-based approach for automated screening of literature that was found to improve the accuracy and efficiency of classification of academic papers. Traditional keyword-based and rule-based methods are beset by static filtering constraints, which make them less efficient in identifying pertinent research papers in dynamic disciplines. Applying Deep Q-Networks (DQN) and the Semantic Scholar API, our RL agent filtered and selected 11 highly relevant papers on its own from a starting collection of 8,934 research articles, maximizing classification through learned decision policies. Experimental evaluation confirms the improved performance of RL-based classification over baseline methods, achieving 91.5% accuracy, 86.3% recall, and 89.7% precision. Compared to keyword-based filtering, the RL model improved recall by 12.5% and precision by 9.8%, testifying to its effectiveness in lowering false positives and false negatives. Error analysis also highlighted false positive (8.3%) and false negative (5.2%) rates, testifying to the robustness of RL-based decision-making in systematic reviews. Despite these enhancements, there are certain issues that still exist. RL models rely on previous data, and therefore are vulnerable to shifts in research trends. Computational complexity, hyperparameter sensitivity, and data set imbalances are some other drawbacks. Nonetheless, the outcome of this research showcases the efficacy, flexibility, and scalability of RL-based classifier models in automatically screening literature. The proposed architecture presents a highly intelligent alternative over traditional methods in saving manual efforts to a greater extent while ensuring precise classification accuracy. By maximizing screening and classification processes, RL-based methods optimize the efficacy of systematic literature reviews and enable researchers to optimize large academic storehouse searches more effectively.

8. Future Work

The RL-based literature screening model has been doing well; however, it is possible that certain improvements could be made such that it could be more adaptable, efficient, and usable on larger study spaces. The future work will include enhancing the accuracy and efficiency of the reinforcement learning model for classification by combining multi-agent reinforcement learning (MARL) systems, where different RL agents work collaboratively to maximize research paper classification. Further, the integration of reinforcement learning and deep learning-based natural language processing models such as transformers can enhance contextual awareness, while meta-reinforcement learning enables the model to keep on learning new research directions without requiring retraining. In order to maximize generalizability to multiple areas of research, future studies will examine transfer learning approaches that can leverage pre-trained reinforcement learning models for adaptation in new domains using a minimal amount of extra training, as well as few-shot learning methods for classifying articles in domains where labeled data are scarce. Federated learning will be explored in order to optimize model performance without sacrificing data privacy in multiple research institutions. Other than e-learning and computer science, literature screening based on reinforcement learning can also be applied in biomedical and health research, where systematic reviews are important for clinical decision-making; legal and patent document classification, to make it easier for professionals to access long legal documents effectively; and social sciences, to aid in the detection of trends in policy and governance research. Another critical area of research is human-AI interaction optimization in literature filtering through the development of interactive reinforcement learning models enabling researchers to provide immediate feedback to enhance categorization choices. Visualization tools can be integrated to enable researchers to gain better insights into the reasoning behind the RL model's acceptance or rejection of certain papers and enhance trustworthiness and explainability. Explainable AI (XAI) methods will enhance decision-making transparency by AI. To enable real-world adoption, subsequent research will investigate cloud-based implementations of reinforcement learning to screen large amounts of literature, optimize model inference rates to enable real-time classification, and overcome ethical considerations of AI-aided academic research filtering to allow just and unbiased classification. In overcoming these challenges, RL-based literature screening will be an even more effective, transparent, and scalable study tool across all fields.

References

[1] Evi Sachini, Konstantinos Sioumalas-Christodoulou, Stefanos Christopoulos, Nikolaos Karampekios; AI for AI: Using AI methods for classifying AI science documents. *Quantitative Science Studies* 2022; 3(4): 1119–1132. https://doi.org/10.1162/qss_a_00223

[2] R. Vijaya Kumar Reddy, Dr. U. Ravi Babu; A Review on Classification Techniques in Machine Learning. *International Journal of Advance Research in Science and Engineering*, 2018, 7(3): 40–48

[3] Manmeet Kaur; A Comprehensive Overview of Artificial Intelligence-Based Classification Techniques. *International Journal of Science and Research Archive*, 2024, 11(2): 125–129. https://doi.org/10.30574/ijsra.2024.11.2.0387

[4] Sarker, I.H. AI-Based Modeling: Techniques, Applications and Research Issues Towards Automation, Intelligent and Smart Systems. *SN COMPUT. SCI.* 3, 158 (2022). https://doi.org/10.1007/s42979-022-01043-x

[5] Kotsiantis, S.B., Zaharakis, I.D. and Pintelas, P.E. Machine learning: a review of classification and combining techniques. *Artif Intell Rev* 26, 159–190 (2006). https://doi.org/10.1007/s10462-007-9052-3

[6] S. B. Kotsiantis. Supervised Machine Learning: A Review of Classification Techniques. *Informatica* 31 (2007) 249-268.

[7] J. Heidemann and C. Papdopoulos, "Uses and Challenges for Network Datasets," *2009 Cybersecurity Applications and Technology Conference for Homeland Security*, Washington, DC, USA, 2009, pp. 73-82. https://doi:10.1109/CATCH.2009.29

[8] A. Jain, A. Singh and I. Jain, "Use of Machine Learning in Application of Artificial Intelligence," *2021 10th International Conference on System Modeling & Advancement in Research Trends (SMART)*, MORADABAD, India, 2021, pp. 29-34, https://doi:10.1109/SMART52563.2021.9676318

[9] Smadi, T., Al Issa, H., Trad, E. and Smadi, K. (2015) Artificial Intelligence for Speech Recognition Based on Neural Networks. *Journal of Signal and Information Processing*, 6, 66-72. https://doi:10.4236/jsip.2015.62006
[10] Shakya, Subarna. (2020). Analysis of Artificial Intelligence based Image Classification Techniques. *Journal of Innovative Image Processing*. 2. 44-54. https://doi:10.36548/jiip.2020.1.005
[11] I. Pesovski, A. M. Bogdanova and V. Trajkovik, "Systematic Review of the published Explainable Educational Recommendation Systems," *2022 20th International Conference on Information Technology Based Higher Education and Training (ITHET)*, Antalya, Turkey, 2022, pp. 1-8, https://doi:10.1109/ITHET56107.2022.10032029
[12] Uthman OA, Court R, Enderby J, Al-Khudairy L, Nduka C, Mistry H, Melendez-Torres GJ, Taylor-Phillips S, Clarke A. Increasing comprehensiveness and reducing workload in a systematic review of complex interventions using automated machine learning. *Health Technol Assess*. 2022, https://doi:10.3310/UDIR6682
[13] Van de Schoot, R., de Bruin, J., Schram, R. et al. An open source machine learning framework for efficient and transparent systematic reviews. *Nat Mach Intell* 3, 125–133 (2021). https://doi.org/10.1038/s42256-020-00287-7
[14] Jia, J., & Wang, W. (2020). Review of reinforcement learning research. *2020 35th Youth Academic Annual Conference of Chinese Association of Automation (YAC)*. https://doi:10.1109/yac51587.2020.9337653
[15] A. L. Samuel. Some Studies in Machine Learning Using the Game of Checkers [J]. *IBM Journal*. 1967, pp.601-617
[16] Watkins CJCH, Dayan P. Technical Note: Q-learning [J]. *Machine Learning*, 1992, pp.279-292
[17] Cai Wenlan, “Research on Control Methods of Small Unmanned Helicopters Based on Reinforcement Learning”, *National University of Defense Technology*, 2007, pp.16-17
[18] Littman M L. Markov Games as a Frame work for Multi-Agent Reinforcement Learning [J]. *Machine Leproceedings*, 1997, pp.157-163
[19] Zhang Rubo, Zhou Ning, Gu Guochang, etc., “Research on Intelligent Robot Collision Avoidance Method Based on Reinforcement Learning”, “*Robot*”, 1999, pp.204-209
[20] Maja J.Mataric.Learning in behavior-based mufti-robot systems: policies, models, and other agents [J]. *Cognitive Systems Research*, 2001, pp.81-93
[21] Tan Min, Wang Shuo, Cao Zhiqiang, “Multi-robot System”, *Tsinghua University*, 2005.
[22] C. Berner, G. Brockman, B. Chan, V. Cheung, P. D ̨ebiak, C. Dennison, D. Farhi, Q. Fischer, S. Hashme, C. Hesse, R. Józefowicz, S. Gray, C. Olsson, J. Pachocki, M. Petrov, H. P. d. O. Pinto, J. Raiman, T. Salimans, J. Schlatter, J. Schneider, S. Sidor, I. Sutskever, J. Tang, F. Wolski, and S. Zhang, "Dota 2 with large scale deep reinforcement learning," *arXiv preprint arXiv:1912.06680*, 2019. https://arxiv.org/abs/1912.06680
[23] Oliver G. Selfridge, Richard S. Sutton, and Andrew G. Barto. Training and tracking in robotics. *In Proceedings of the International Joint Conference on Artificial Intelligence*, 1985.
[24] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. *In Proc. of the International Conf. on Machine learning (ICML)*, 2018, https://doi.org/10.48550/arXiv.1801.01290
[25] John Schulman, Sergey Levine, Pieter Abbeel, Michael I. Jordan, and Philipp Moritz. Trust region policy optimization. *In Proc. of the International Conf. on Machine learning (ICML)*, 2015.
[26] Evan Greensmith, Peter L. Bartlett, and Jonathan Baxter. Variance reduction techniques for gradient estimates in reinforcement learning. *In Advances in Neural Information Processing Systems (NIPS)*, 2001.
[27] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. *In Proc. of the International Conf. on Machine learning (ICML)*, 2017.
[28] Nico Gürtler, Sebastian Blaes, Pavel Kolev, Felix Widmaier, Manuel Wuthrich, Stefan Bauer, Bernhard Schölkopf, and Georg Martius. Benchmarking offline reinforcement learning on real-robot hardware. *In Proc. of the International Conf. on Learning Representations (ICLR)*, 2023.
[29] M. L. Puterman, Markov Decision Processes: Discrete Stochastic Dynamic Programming. *Wiley-Interscience*, 1994.
[30] E. Even-Dar, S. M. Kakade, and Y. Mansour, “Experts in a Markov decision process,” in *Advances in Neural Information Processing Systems*, 2005, pp. 401–408.
[31] S. Mannor, D. Simester, P. Sun, and J. N. Tsitsiklis, “Bias and variance approximation in value function estimates,” *Management Science*, vol. 53, no. 2, pp. 308–322, 2007.
[32] G. N. Iyengar, “Robust dynamic programming,” *Mathematics of Operations Research*, vol. 30, no. 2, pp. 257–280, 2005.
[33] G. Neu, A. Gyorgy, C. Szepesvari, and A. Antos, “Online Markov decision processes under bandit feedback,” in *Advances in Neural Information Processing Systems*, 2010, pp. 1804–1812.
[34] R. I. Brafman and M. Tennenholtz, “R-max: A general polynomial time algorithm for near-optimal reinforcement learning,” *Journal of Machine Learning Research*, vol. 3, pp. 213–231, 2002.
[35] T. Jaksch, R. Ortner, and P. Auer, “Near-optimal regret bounds for reinforcement learning,” *Journal of Machine Learning Research*, vol. 99, pp. 1563–1600, 2010.
[36] A. Nilim and L. El Ghaoui, “Robust control of Markov decision processes with uncertain transition matrices,” *Operations Research*, vol. 53, no. 5, pp. 780–798, 2005.
[37] H. Xu and S. Mannor, “Distributionally robust Markov decision processes,” *Mathematics of Operations Research*, vol. 37, no. 2, pp. 288–300, 2012.
[38] S. Mannor, O. Mebel, and H. Xu, “Lightning does not strike twice: Robust MDPs with coupled uncertainty,” in *International Conference on Machine Learning (ICML)*, 2012.
[39] J. Y. Yu, S. Mannor, and N. Shimkin, “Markov decision processes with arbitrary reward processes,” *Mathematics of Operations Research*, vol. 34, no. 3, pp. 737–757, 2009.
[40] Dixit, Avinash K. (1990). Optimization in Economic Theory (2nd ed.). *Oxford University Press*. p. 164. ISBN 0-19-877211-4.
[41] Kirk, Donald E. (1970). Optimal Control Theory: An Introduction. *Prentice-Hall*. p. 55. ISBN 0-13-638098-0
[42] I. Szcze´sniak and B. Wo´zna-Szcze´sniak, "Generic Dijkstra: correctness and tractability," *NOMS 2023-2023 IEEE/I-FIP Network Operations and Management Symposium*, Miami, FL, USA, 2023, pp. 1-7, doi:10.1109/NOMS56928.2023.10154322
[43] Kamien, Morton I.; Schwartz, Nancy L. (1991). Dynamic Optimization: The Calculus of Variations and Optimal Control in Economics and Management (Second ed.). *Amsterdam: Elsevier*. p. 261. ISBN 0-444-01609-0

[44] Jones, Morgan; Peet, Matthew M. (2020). "Extensions of the Dynamic Programming Framework: Battery Scheduling, Demand Charges, and Renewable Integration". *IEEE Transactions on Automatic Control*. 66 (4): 1602–1617. arXiv:1812.00792. doi:10.1109/TAC.2020.3002235

[45] Otterlo, Martijn & Wiering, Marco. (2012). Reinforcement Learning and Markov Decision Processes. *Reinforcement Learning: State of the Art*. 3-42. doi:10.1007/978-3-642-27645-3_1

[46] Kudo, T., Ohtsuki, T. Cell range expansion using distributed Q-learning in heterogeneous networks. *J Wireless Com Network* 2013, 61 (2013). https://doi.org/10.1186/1687-1499-2013-61

[47] Caputo, Michael R. (2005). Foundations of Dynamic Economic Analysis: Optimal Control Theory and Applications. *New York: Cambridge University Press*. p. 185. ISBN 0-521-60368-4.

[48] Weber, Thomas A. (2011). Optimal Control Theory: with Applications in Economics. *Cambridge: The MIT Press*. p. 82. ISBN 978-0-262-01573-8

[49] Corbae, Dean; Stinchcombe, Maxwell B.; Zeman, Juraj (2009). An Introduction to Mathematical Analysis for Economic Theory and Econometrics. *Princeton University Press*. p. 145. ISBN 978-0-691-11867-3.

[50] Kamien, Morton I.; Schwartz, Nancy L. (1991). Dynamic Optimization: The Calculus of Variations and Optimal Control in Economics and Management (2nd ed.). *Amsterdam: North-Holland*. p. 259. ISBN 0-444-01609-0.

[51] Kamalapurkar, Rushikesh; Walters, Patrick; Rosenfeld, Joel; Dixon, Warren (2018). "Optimal Control and Lyapunov Stability". Reinforcement Learning for Optimal Feedback Control: A Lyapunov-Based Approach. *Berlin: Springer*. pp. 26–27. ISBN 978-3-319-78383-3.

[52] Zhou, X. Y. (1990). "Maximum Principle, Dynamic Programming, and their Connection in Deterministic Control". *Journal of Optimization Theory and Applications*. 65 (2): 363–373. doi:10.1007/BF01102352.S2CID122333807

[53] Shahzad, K. and Khan, S.A. (2023), "Effects of e-learning technologies on university librarians and libraries: a systematic literature review", *The Electronic Library*, Vol. 41 No. 4, pp. 528-554. https://doi.org/10.1108/EL-04-2023-0076

[54] D. van Heerden and M. Serote, "Exploring the Validity of Continuous Assessment in a First-Year Programming Course at a Comprehensive Open Distance e-Learning University," *Proc. 22nd Eur. Conf. e-Learning - ECEL 2023*, vol. 22, no. 1, 2023, https://doi:10.34190/ecel.22.1.1467

[55] A. Zabolotsky, “Method of the e-learning support system using as a development of informational and communicative competence of the university distance education center employees”, *SR:PE*, no. 1 (21), pp. 27–29, Jan. 2018.

[56] Phakathi, L. (2023). Technology-Based Support of Final Year Bachelor of Education Students in a South African Open Distance e-Learning Institution. *International Journal on Open and Distance E-Learning*, 9(1). https://doi.org/10.58887/ijodel.v9i1.104

[57] A. S. Abdalkafor, Z. J. Mushref and A. M. Khalaf, "E-Learning Technology Impact On the Development and Sustainability of Training Skills of Anbar University Graduates," *2023 16th International Conference on Developments in eSystems Engineering (DeSE)*, Istanbul, Turkiye, 2023, pp. 194-199, doi:10.1109/DeSE60595.2023.10469280

[58] H. T. Alrikabi, N. A. Jasim, B. H. Majeed, A. A. Zkear, and I. R. N. ALRubeei, “Smart Learning based on Moodle E-learning Platform and Digital Skills for University Students”, *Int. J. Recent Contrib. Eng. Sci. IT*, vol. 10, no. 01, pp. 109–120, Mar. 2022.

[59] Özbey, M., Kayri, M. Investigation of factors affecting transactional distance in E-learning environment with artificial neural networks. *Educ Inf Technol* 28, 4399–4427 (2023). https://doi.org/10.1007/s10639-022-11346-4

[60] Mukhametgaliyeva, S., Gura, A., Dudnik, O., & Khudarova, A. (2022). The Use of Social Networks in E-Learning Technologies in the Context of Distance Education. *Sustainability*, 14(14), 8949. https://doi.org/10.3390/su14148949

[61] Yıldız, G., ¸Sahin, F., & Do˘gan, E. (2022). University students with special needs in the e-learning system: Characteristics, experiences and competencies. *Anadolu Journal of Educational Sciences International*, 12(2), 468-491. https://doi.org/10.18039/ajesi.1052854

[62] Rizky Windar Amelia, Djoko Suhardjanto, Agung Nur Probohudono, Setyaningtyas Honggowati, Weighted Index of Cultural Heritage Disclosure in Indonesia, *Eduvest - Journal of Universal Studies*: Vol. 2 No. 8 (2022): Journal Eduvest - Journal of Universal Studies

[63] Ardhana, Valian & Sapi’i, Muhammad & Hasbullah, Hasbullah & Sampetoding, Eliyah. (2022). Web-Based Library Information System Using Rapid Application Development (RAD) Method at Qamarul Huda University. *The IJICS (International Journal of Informatics and Computer Science)*. 6. 43. https://doi.10.30865/ijics.v6i1.4031

Authors’ Profiles

Enes Bajrami is a PhD candidate and researcher in Computer Science at Ss. Cyril and Methodius University- Faculty of Computer Science and Engineering. His research focuses on software engineering and artificial intelligence, encompassing deep reinforcement learning, microservices, green computing, cloud orchestration, and medical AI applications. In addition to his academic work, he has extensive practical experience as a web developer. He currently serves as the Coordinator of the Broadband Implementation Unit at the state-owned enterprise National Broadcasting, where he leads the deployment of broadband infrastructure, bridging academic research with real-world technological applications.

Florim Idrizi is a full professor at the University of Tetova, where he currently serves as Dean of the Faculty of Natural Sciences and Mathematics. Over the years, he has held several key administrative positions, including Vice-Dean, Head of the Computer Science study program, and Acting Dean of the same faculty. With more than two decades of academic experience, Professor Idrizi has established a distinguished record in both teaching and research. His scholarly interests span Web Technologies, Data Structures, Digital Circuits, and Cryptography, reflecting his deep expertise across the fields of computer science and related disciplines.

Shpend Ismaili is an Assistant Professor in the Department of Computer Science at the University of Tetova. He earned his PhD in Computer Science and has a professional background that includes teaching at the High School Mosha Pijade prior to joining the university. His research interests cover Electronics, Computer Networking, and fundamental areas of Information Technology.

How to cite this paper: Enes Bajrami, Florim Idrizi, Shpend Ismaili, "Reinforcement Learning for Automated Literature Screening: Enhancing E-Learning and University Research Classification in Computer Science", International Journal of Information Technology and Computer Science(IJITCS), Vol.18, No.1, pp.42-56, 2026. DOI:10.5815/ijitcs.2026.01.03