Artificial intelligence for human learning: A review of machine learning techniques used in education research and a suggestion of a learning design model

American Journal of Education and Learning
Vol. 9, No. 1, 1-21, 2024
e-ISSN:2518-6647

Donggil Song

Engineering Technology & Industrial Distribution, College of Engineering, Texas A&M University, College Station, TX 77843, USA.
Email: creative@tamu.edu

ABSTRACT

The goal of this research is to (1) identify the status and development of AI and ML-based learning support systems and their impact on human learning, with a specific focus on techniques employed in previous research, and (2) demonstrate the process of designing a learning support system using AI. Artificial intelligence (AI) and machine learning (ML) technologies have received attention in education. The existing research on AI in education is examined, considering the implications of its application in research. Noteworthy ML techniques from the literature are explained, followed by a discussion on leveraging AI and ML technologies to enhance learning support. Additionally, with consideration of both front-end and back-end approaches,a framework for incorporating AI into education is proposed. Subsequently, a learning design model, Self-regulated Learning with AI Assistants (SLAA), is suggested for addressing the objectives of AI-based learning support system design. The categorization of AI and ML techniques in education research reveals nine types, including supervised learning, mining approaches, and Bayesian techniques. The exploration illustrates how these techniques can be employed in designing a learning support system. This paper provides an empirical overview of AI in education, addresses technological and pedagogical considerations for developing personalized and adaptive learning environments, and outlines the challenges and potential future research directions.

Keywords: *Artificial intelligence, Learning design, Learning system design, Machine learning, Personalized learning, Self-regulated learning.*

DOI: 10.55284/ajel.v9i1.1024
Citation | Song, D. (2024). Artificial intelligence for human learning: A review of machine learning techniques used in education research and a suggestion of a learning design model. *American Journal of Education and Learning*, *9*(1), 1–21.

Funding: This study received no specific financial support.
Institutional Review Board Statement: Not applicable.
Transparency: The author confirms that the manuscript is an honest, accurate, and transparent account of the study; that no vital features of the study have been omitted; and that any discrepancies from the study as planned have been explained. This study followed all ethical practices during writing.
Competing Interests: The author declares that there are no conflicts of interests regarding the publication of this paper.
History: Received: 22 November 2023/ Revised: 3 January 2024/ Accepted: 15 January 2024/ Published: 12 February 2024
Publisher: Online Science Publishing

Highlights of this paper

- The aim of this research is to assist researchers, educators, and practitioners in understanding AI and machine learning techniques, enabling their full utilization in various educational contexts.
- This paper focuses on adopting artificial intelligence in developing learning support systems.
- The authors propose a framework for incorporating AI in education, review specific AI techniques used in previous research, and suggest a design model.

1. INTRODUCTION

Artificial intelligence (AI) technology has garnered attention in education due to the abundance of big data collected from education-related systems and the capabilities of powerful chatbots. Since the late 1990s, intelligent tutoring systems have addressed student needs by representing instructional decisions through interactions with the learner (Beck, Stern, & Haugsjaa, 1996). Other approaches have been explored, including automated essay-scoring systems (Shermis & Burstein, 2003) and adaptive simulation-based military training (Department of the Army, 2011). Different procedural interpretations of big data give the system unique meanings, knowledge, and intelligence, often determined dynamically by the learning context. Recently, generative AI has introduced new possibilities and simultaneously raised serious concerns in education. As such, a wide variety of AI types, approaches, and methods can be readily found in education. The characteristics of AI, such as customization and adaptiveness, offer a potential solution for enhancing current learning designs to better support learners' diverse needs.

While numerous research reports on the use of AI in education exist, it can be challenging to discern which aspects of a learning environment involve AI and how to integrate AI into the development of learning support systems. Importantly, few studies have explained or suggested specific techniques or a design model that education researchers can apply. To address these issues, we aim to (1) illustrate specific machine learning (ML) techniques adopted in previous research, (2) clarify the use of AI in education, and (3) propose a design model for a learning support system using AI. First, we explain the techniques reported in the literature and suggest a framework for incorporating AI into education. Second, we present a learning design model to address the objectives of AI-based learning support system design. Lastly, we discuss the challenges of AI in education and outline future research agendas.

2. RELATED WORKS

2.1. Machine Learning Techniques for Learning Design

Machine learning (ML) refers to a study or field of techniques or algorithms in which computer programs learn something (e.g., data patterns) from data to perform a task independently. ML systems can learn primarily from big data without being explicitly programmed for a specific task. We have reviewed various ML techniques relevant to educational purposes and summarized them in Table 1.

2.1.1. Supervised ML Techniques

Supervised techniques rely on labeled input data to learn a function that predicts an output when given new unlabelled data. The process involves *training*, where the system learns, and *testing*, where predictions are made. The most common supervised task is classification, where a system observes a training dataset with input (predictor features) and output (classes or labels). Subsequently, the system learns a function that maps from input to output. The preparation steps typically include (1) data collection, (2) data pre-processing, and (3) algorithm selection.

One popular method for collecting learner data is through the use of learning management systems (LMS). If the data contains noise (i.e., meaningless, unstructured, and/or corrupted data that the machine cannot understand) and missing feature values, data preprocessing is necessary. During this phase, irrelevant and redundant features are removed. To execute the ML process successfully, specific and appropriate ML techniques must be chosen. Below, we briefly introduce examples of supervised ML techniques.

Naïve Bayes. This is built on Bayes' theorem for probabilistic classification and functions as a supervised learning approach. Recognized for its simplicity, it stands out as one of the fastest classification algorithms in terms of running time; it is particularly suitable for real-time tasks. If a learning designer is dealing with a large dataset containing a small number of variables, Naïve Bayes proves to be a worthwhile technique.

Decision Tree. As an intuitive technique, Decision Tree visualizes results in a structure reminiscent of a tree. This tree-shaped representation helps in understanding how the results were calculated. For instance, in a dataset containing information such as students' age, gender, GPA, and the successful completion of an exam, each feature (e.g., age, gender, GPA) can be considered a branch of a tree. The label of the data (the successful completion of the exam) becomes the conclusion, analogous to a leaf in the tree.

Random Forest. This technique utilizes multiple Decision Trees. Each tree's vote for class prediction, along with the prediction results, is used to determine the class that received the most votes. This can generate *ensemble* predictions that outperform the individual Decision Tree's prediction when there is a low correlation among the trees. In cases where Decision Trees lead to overly tall structures, called *deep trees*, irregular patterns and overfitting issues may arise.

Support Vector Machine (SVM). SVM serves both classification and regression analysis purposes. It identifies *decision boundaries* (separating hyperplanes) in an n-dimensional space to classify data points. For instance, consider a 2-dimensional space with math scores on the x-axis and reading scores on the y-axis, each point representing a student's scores in both subjects. If two students' data points are not in the same spot, SVM determines the hyperplane that separates them. Through training, SVM seeks the optimal line with the maximum distance between data points. In the case of three input features, the hyperplane becomes a two-dimensional plane, and so forth.

K-Nearest Neighbor. This technique assumes that similar data points are located in their neighborhood (proximity). It calculates the distance between a test sample and specified training samples on a graph to determine if the test sample is close to a specific class within the training samples. The parameter *K* denotes the number of nearest neighbors. If *K* is set to 1, a data point is assigned to the class of its *single* nearest neighbor. The results and accuracy may vary depending on the chosen value of *K*.

2.1.2. Unsupervised ML Techniques

The most common unsupervised learning task is clustering, which encompasses a specific group of ML techniques for creating homogeneous groups based on data features. A cluster refers to a collection of observations (i.e., data points) aggregated together due to certain similarities identified by the ML technique. In unsupervised ML, the system learns dataset patterns even in the absence of labels or classes. For example, a clustering model for groups of students' performance might gradually develop the concept of high-and low-performing students without the data of their final grades.

K-Means Clustering. This popular technique identifies a specific number (*K*) of centroids and assigns every observation to the nearest cluster. A centroid denotes the geometric center, which is the arithmetic mean position of all data points on the graph. Consider a 2-dimensional graph with the x-axis representing math scores and the y-axis representing science scores. If we aim to divide the student data into 2 groups (*K*=2), the iterative process categorizes the dataset into 2 clusters based on the similarity of their math and science scores. This technique ensures that the sum of the squared distance between the data points and the cluster's centroid is minimized.

Hierarchical Clustering. This is an alternative approach to conventional clustering for identifying groups in the dataset by building a hierarchy of clusters. Initially, each data point is treated as a separate cluster. The algorithm then identifies two close clusters and merges them. The distance between two clusters is calculated based on the length of a straight line between two observations (clusters). This process continues until all clusters are merged, resulting in a hierarchical relationship between the clusters represented as a dendrogram, a tree-shaped diagram.

Ripley's K function. This function assesses the structure of underlying patterns in a dataset by analyzing the spatial point pattern of data on event locations. It determines whether a dataset is dispersed, clustered, or randomly distributed (i.e., detects deviations from spatial homogeneity). This technique has been employed to describe a set of data point locations, examine research hypotheses about data point patterns, and estimate parameters in a spatial point process model.

Table 1. The use of ML techniques for human learning in the literature.

Categories	Techniques	Characteristics	Examples
Supervised ML techniques	Naïve bayes	One of the fastest classification algorithms	Aguiar, Ambrose, Chawla, Goodrich, and Brockman (2014) and Sabourin, Shores, Mott, and Lester (2013)
	Decision tree	An intuitive technique	Kai, Almeda, Baker, Heffernan, and Heffernan (2018);Sabourin et al. (2013) and Taherkhani and Malmi (2013)
	Random forest	Utilizes a multitude of decision trees	Spoon et al. (2016)
	Support vector machine	Finds decision boundaries	Yoo and Kim (2014) Cetintas et al. (2010)
	K-nearest neighbor	Calculates the distance between test and training samples	Gray, McGuinness, Owende, and Hofmann (2016)
Unsupervised ML techniques	K-means clustering	Categorizes the dataset into K clusters	Ferguson and Clow (2015); Lee and Tan (2017) and Vaessen, Prins, and Jeuring (2014)
	Hierarchical clustering	Follows a stepwise method that merges two observations at a time	Boroujeni and Dillenbourg (2019);Hao, Shu, and von Davier (2015) and Mirriahi, Liaqat, Dawson, and Gašević (2016)
	Ripley's K function	Conducts multi-distance spatial cluster analysis	Mallavarapu et al. (2015)
Other ML	Deep learning	Utilizes neural networks with hidden layers	Mao (2018)
	Reinforcement learning	Takes suitable action to maximize reward	Dorça, Lima, Fernandes, and Lopes (2013)
Advanced statistical techniques	Latent class analysis	A subset of structural equation modeling	Xu and Recker (2011) Pelaez, Levine, Fan, Guarcello, and Laumakis (2019)
	Hierarchical linear modeling	Handles multiple levels (Data containing measurement of different levels)	Miller, Soh, Samal, Kupzyk, and Nugent (2015)
	Singular value decomposition	Analyzes a factorization of a matrix	Morsy and Karypis (2019)
	Rasch model	Identifies latent variables and reveals the probability of an individual	Waters, Studer, and Baraniuk (2014)
	Expectation-maximization	Identifies maximum likelihood estimators in latent variable models	Harley, Bouchet, Trevors, and Azevedo (2013)
	Jaccard similarity	The size of the intersection divided by the size of the union of two sets	Dascalu et al. (2015)
Natural	Automated text	Evaluates semantic content, syntactic	Ezen-Can and Boyer

Categories	Techniques	Characteristics	Examples
language processing	evaluation, coh-metrix	structure, and rhetorical structure; quantifies the cohesion and coherence of the text	(2015);Ezen-Can, Grafsgaard, Lester, and Boyer (2015);Knight, Buckingham Shum, Ryan, Sándor, and Wang (2018) andSerban, Lowe, Henderson, Charlin, and Pineau (2018)
	N-gram model, cosine similarity	Predicts the next item in a sequence of textual data; calculates the cosine of the angle between two vectors	Schneider and Pea (2015)
Temporal and sequential data processing	Markov modeling	A system linked with a random probability	Althoff, Clark, and Leskovec (2016);Andrade, Danish, and Maltese (2017); Geigle and Zhai (2017);Shen, Mostafavi, Barnes, and Chi (2018) and Vaessen et al. (2014) Kowalski, Zhang, and Gordon (2014)
	Stochastic gradient descent	Starts from a random data point at each iteration to move down its slope in steps	Kassak, Kompan, and Bielikova (2016)
	Genetic algorithm	A search heuristic and a random-based classical evolutionary algorithm	Chen (2008)
Bayesian techniques	Bayesian modeling	A mathematical procedure that applies probabilities to conventional statistical problems	Gardner and Brooks (2018)
	Bayesian network	Models a set of variables and their conditional dependencies	Desmarais and Baker (2012) Lester et al. (2013)
	Bayesian knowledge tracing	Models each learner's mastery of the knowledge being instructed	Cui, Chu, and Chen (2019)
Mining techniques	Sequence mining	Discovers useful, novel, and unexpected rules in sequences	Taub and Azevedo (2018)
	Process mining	Simplifies the process model by identifying trends and patterns	Sonnenberg and Bannert (2016)
	Data stream mining	Extracts structures from continuous and rapid data points	Mahzoon, Maher, Eltayeby, Dou, and Grace (2018)
Network analysis	Social network analysis	Uses networks and graph theory to investigate relationships between network entities	Fincham, Gašević, and Pardo (2018) and Gruzd, Paulin, and Haythornthwaite (2016)
	Epistemic network analysis	Identifies connections in coded data and represents them in dynamic network models	Siebert-Evenstone et al. (2017)
	Clique percolation method	Identifies overlapping community structure within networks	Hecking, Ziebarth, and Hoppe (2014)

2.1.3. Other ML Techniques

Some techniques may not neatly fit into the categories of supervised or unsupervised ML.

Deep Learning. A Deep Learning model can be trained to perform classification and clustering tasks using images, text, or sound data. Inspired by the structure and function of the human brain, it employs neural networks with hidden layers, big data, and computational resources. The fundamental unit of a neural network is a node, and connections between nodes are modeled, akin to the connections between neurons in biological brains, which are trained and developed over time.

Reinforcement Learning. This involves helping a system take appropriate actions to maximize rewards in a given situation. Through learning from a series of feedback, reinforcements, rewards, or punishments, this technique accumulates training examples through trial and error to optimize long-term rewards. It is employed in creating agents (e.g., ChatGPT [Generative Pre-Trained Transformer]) that perform actions in an environment, receiving rewards based on the agent's status when it acts.

2.1.4. Advanced Statistical Techniques

The following techniques are commonly addressed in the AI-in-education field, although they may or may not be considered ML techniques.

Latent Class Analysis. As a subset of *structural equation modeling*, this is a model-based clustering approach that derives latent classes using a probabilistic model from data distribution without requiring data to be uncorrelated. This technique identifies relationships between observed multiple variables and latent variables, specifically finding groups (classes) of data points in categorical multivariate data. Unlike unsupervised ML algorithms, which identify clusters with autonomously chosen measures, Latent Class Analysis uses a model and calculates probabilities for each class.

Hierarchical Linear Modeling. This modeling technique identifies relationships within and between hierarchically structured (or multiple levels of) data, such as individual learner, course, school, and district levels (also called nested data). It overcomes an assumption violation of nested data (e.g., an individual learner's performance closely relates to their course or school's performance) when using other statistical methods.

Singular Value Decomposition (SVD). SVD analyzes a factorization of a matrix, decomposing a vector into its components along the x and y axes. It transforms vector data into orthogonal axes, reducing data dimensions like factor analysis. SVD learns representations of data by transforming features and determining an optimal representation for data by discovering the representations needed for feature detection.

Rasch Model. As a type of Generalized Linear Model, Rasch Model is similar to Item Response Theory (IRT, which analyzes students' answers to a test to find a relationship between the students' performance on a test item and their overall performance levels to improve the accuracy and reliability of measurement). However, Rasch Model is created from the data, while the IRT fits a model to data. This technique identifies latent variables and reveals the probability of an individual giving a correct response to a test item.

Expectation-Maximization (EM). Similar to a clustering algorithm, EM identifies maximum likelihood estimators in latent variable statistical models. It has two iterative processes: (1) using the observed data to estimate incomplete (missing, hidden, or latent) variables in the dataset (Expectation), and (2) calculating parameters to maximize the parameters of the model (Maximization).

Jaccard Similarity. This is a statistic employed to comprehend the similarities between sample sets using the Jaccard Similarity Coefficient, which refers to the size of the intersection divided by the size of the union of two sets. The resulting coefficient represents the percentage of similarity between the two sample sets, calculated as the number common to both sets divided by the number in either set.

2.1.5. Natural Language Processing (NLP)

NLP finds applications in various areas, such as automatic phone answering machines and smart speakers. In education, it has been utilized for assessing student writing, known as automated essay scoring. These systems evaluate the semantic content, syntactic structure, and rhetorical structure of students' writing.

N-gram Model and Cosine Similarity.In most research, N-gram Model and Cosine Similarity are frequently employed. N-gram refers to the sequence of words; for example, a 2-gram is a two-word sequence (e.g., "this is," "artificial intelligence"). The N-gram Model predicts the next item in a sequence of textual data by calculating the likelihood of the next item. This method is particularly useful for text similarity analysis, often using Cosine Similarity. In this case, N-gram can be used for feature extraction, and Cosine Similarity calculates the cosine of the angle between the two vectors (e.g., two documents).

2.1.6. Temporal and Sequential Data Processing

The following techniques address problems typically involving the numerical values of a system that dynamically changes over time.

Markov Modeling. Markov Modeling refers to a stochastic model that represents temporal or sequential data in a changing system. Stochastic implies *randomness*, and a stochastic system is linked to a random probability. The model comprises states, transition schemes between states, and emissions of outputs. Markov Model operates in a finite state, where transitions between states are assigned probabilities. The probability of transitioning to a new state depends solely on the previous state. Depending on output characteristics, there are two types: Discrete Markov Model, which has parameters as transition probabilities between states, and Hidden Markov Model, where the states are concealed.

Stochastic Gradient Descent. Gradient Descent locates the lowest point of a function through an iterative process. However, this method might be inefficient with large datasets due to computing derivatives concerning numerous features. To overcome the limitation, Stochastic Gradient Descent starts from a random data point at each iteration, moving down the slope in steps, reducing computational load.

Genetic Algorithm. Adopting concepts from natural selection and evolution in biology, Genetic Algorithm assumes that the fittest data points are selected for reproduction to produce *offspring* for the next generation. It involves phases such as the initial population, fitness function, selection, crossover, and mutation. This technique tackles optimization problems and search problems with solutions possessing unique properties that can be mutated and altered through random changes.

2.1.7. Bayesian Techniques

In contrast to frequentist statistics modeling, Bayesian Modeling is a mathematical procedure that applies probabilities to conventional statistical problems, representing a form of mathematical formulation of data points.Bayesian Network. This network models a set of variables and their conditional dependencies, representing multivariate probability distributions. Bayesian network use Bayesian inference for probability computations. It is a probabilistic graphical model where each edge (or connection) signifies a conditional dependency or causation, and each node represents a variable. A specific form, Dynamic Bayesian Network, models time series or sequences by relating variables across time steps.

Bayesian Knowledge Tracing. As a learner modeling tool, this technique models each learner's mastery of the instructed knowledge as a latent variable. Frequently used in designing intelligent agents to monitor learners' mastery levels and sequence learning problems, it estimates the probability that a learner masters a particular skill, represented as a dynamic Bayesian network.

2.1.8. Mining Techniques

Sequence Mining. This technique uncovers unexpected rules in sequences. Sequential Pattern Mining identifies common sequences within events, behaviors, or activities. Sequential Rule Mining, an alternative to pattern mining, focuses on the probability that a pattern (a subsequence in several sequences of data) will be followed.

Process Mining. This mining technique simplifies process models by identifying trends and patterns within datasets. It analyzes series of repeated actions in a dataset, comparing data with process models to monitor compliance, detect deviations, and predict discrepancies.

Data Stream Mining. This method extracts data structures from continuous data points and handles a large amount of real-time data (data changing over time). The term stream refers to continuous, massive sequences of data items. This technique is useful when the complete dataset is unknown and/or the data is not stored.

2.1.9. Network Analysis

Social Network Analysis. This utilizes networks and graph theory to investigate relationships among individuals, groups, organizations, and other network entities. It provides a visual analysis of learner relationships in computer-supported collaborative learning environments or social media networks through nodes (points) and ties (lines).

Epistemic Network Analysis. This technique identifies connections between elements in coded data, representing them in dynamic network models. Utilizing coded discourse data, it investigates the strength of elements' associations in an epistemic network. It can be employed to assess epistemic entities, including skills, knowledge, values, and decision-making.

Clique Percolation Method. This method identifies overlapping *communitystructures* (frequent occurrence of groups of nodes) within networks. A clique is a complete graph, and a *k*-clique represents a complete graph with *k* vertices or nodes. In a clique visualization the internal edges of the community and inter-community edges are represented as cliques.

2.1.10. More Thoughts on ML Techniques

Given that algorithms can outperform others under certain circumstances and vice versa, researchers aim to find functions mapping datasets to algorithm performance (Kotsiantis, Zaharakis, & Pintelas, 2006). Thus, investigating how to evaluate each algorithm's performance and compare them is crucial. Russell and Norvig (2010) propose four ways to evaluate an algorithm's performance: (1) completeness ("Is the algorithm guaranteed to find a solution when there is one?"), (2) optimality ("Does the strategy find the optimal solution"), (3) time complexity ("How long does it take to find a solution?"), and (4) space complexity ("How much memory is needed to perform the search?"). Researchers may use various methods to evaluate each ML model's performance, such as cross-validation (e.g., K-Fold Cross-Validation, assessing a prediction model's performance when generalizing to items outside the training set).

2.2. The Front/Back-End Framework of AI in Education

From the review of AI and ML techniques used in education, it becomes apparent that AI has been applied to support human learning in two distinct ways:*front-end* and *back-end*. This framework serves to elucidate the adoption of AI in education.

The *front-end* denotes the segment of a computer system directly interacting with the user, shaping the user experience (client side). Conversely, the *back-end*pertains to the portion not in direct interaction with the user; rather, it typically manages data and system processes (server side). Similarly, we propose two approaches in projects aiming to design learning support systems using AI.

2.2.1. Front-End AI for Human Learning

AI directly interacts with the learner to support their learning, constituting the front-end approach. Examples encompass information retrieval systems and personal assistants, such as Internet of Things interactive tools that communicate directly with the user. The examples given by Oh, Song, and Hong (2020) and Song, Rice, and Oh (2019), are typical ones for conversational agents. The anticipation is for AI systems to undertake tasks currently performed by human instructors or teachers, tasks that are both time-consuming and costly.

2.2.2. Back-end AI for Human Learning

AI supports the learning process through indirect means, specifically through data analysis. Examples can be found in the research fields of Learning Analytics and Educational Data Mining. AI or ML algorithms are applied to analyze learners' learning processes, behaviors, and performances. These fields share the common goal of enhancing education using educational big data through computational techniques, defined as "the use of computational techniques for analyzing data collected from learning environments" (Song, 2018).

3. A LEARNING DESIGN MODEL: SLAA (SELF-REGULATED LEARNING WITH AI ASSISTANTS)

Within the context of the front-back-end framework, we present a practical learning design model. Today's learners navigate a wealth of diverse learning materials, resources, and supportive tools, necessitating them to become inevitably self-directed and self-regulated learners. Recognizing the significance of self-regulated learning (SRL) (Wolters & Taylor, 2012), a concept well-theorized in education and psychology, is crucial. Self-regulated learners "plan, set goals, organize, self-monitor, and self-evaluate at various points during the process of acquisition" (Zimmerman, 1990).

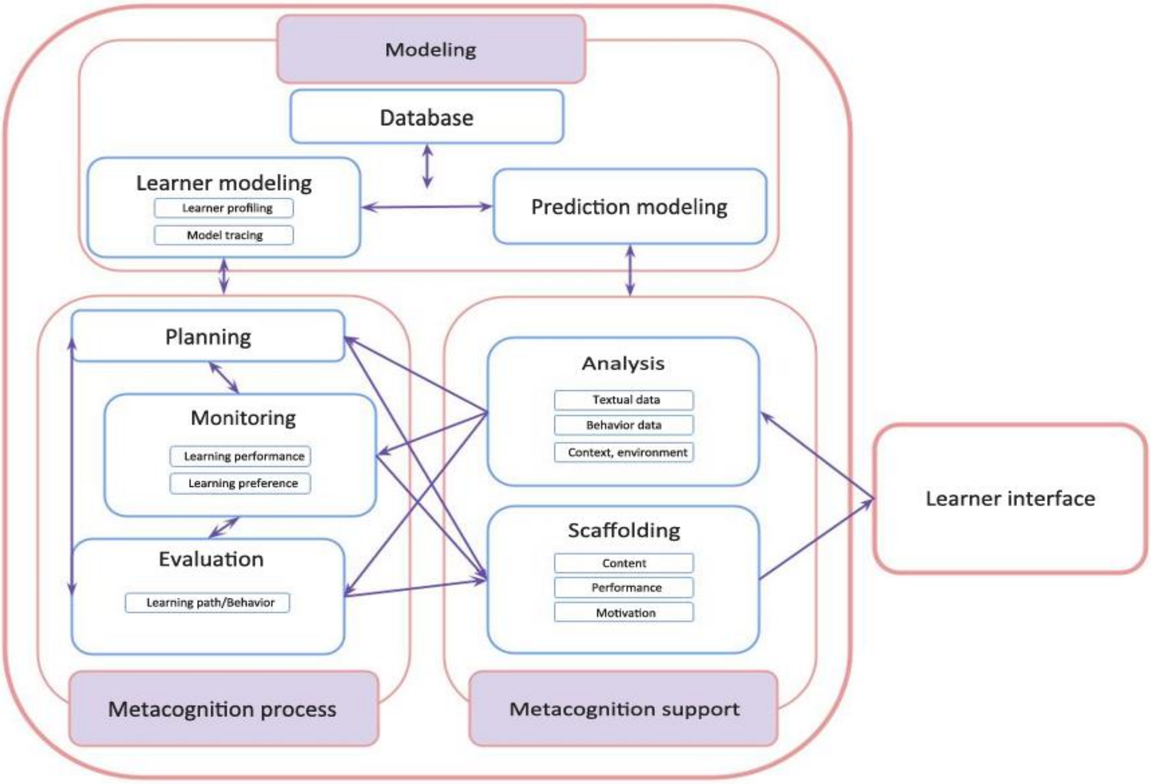

Figure 1. A learning design model: Self-regulated Learning with AIAssistants (SLAA).

As illustrated in Figure 1, we present a learning design model that elucidates the adoption of AI techniques within the front-end and back-end frameworks. The primary functions are proposed to be twofold: personalization from a learning analysis viewpoint (the back-end approach) and SRL scaffolding (specifically, metacognition support) from the interaction design perspective (i.e., the front-end approach).

3.1. Metacognition in SLAA

To support SRL, learning environments must assist learners in developing self-regulation aptitude and skills, enabling them to master the strategies for regulating both cognition and metacognition. This includes managing learning resources to control their learning (Pintrich, 1999). We propose that AI-infused learning environments can be designed to specifically support learners' metacognition. Research indicates that learners engaging in more metacognitive processes tend to exhibit higher learning performance than their counterparts (Muis, Psaradellis, Chevrier, Di Leo, & Lajoie, 2016).

Metacognitive learning encompasses: (1) setting a specific and proximal learning goal, (2) planning how to approach and strategize for a learning task, (3) adopting and implementing a learning strategy to achieve the goal, and (4) monitoring and evaluating the learning progress (Muis et al., 2016; Zimmerman, 2002). These aspects align with the components incorporated in SLAA.

3.1.1. Part 1. The Metacognition Process

As illustrated in Figure 1, the planning, monitoring, and evaluation modules are interconnected in the Metacognition Process within SLAA. These three submodules receive information from the Analysis module in the Metacognition Support module.

Planning. The learner's metacognition process commences with planning. When the learner establishes a learning goal, the AI system provides feasible suggestions generated from the successful outcomes of previous learners with similar profiles (e.g., comparable background information and prior knowledge levels). The Metacognition Process module communicates with the Learner Modeling module, housing learner models in the database. The system relies on past data demonstrating effective matches between learning strategies and appropriate learning goals and tasks.

Monitoring. When learners actively monitor their learning process, their satisfaction and efficacy levels tend to increase (Zimmerman, 2002). However, learners often overestimate their learning performances (Spoelstra, Van Rosmalen, & Sloep, 2014). The Monitoring module in SLAA supports learners' self-monitoring process for accurate evaluation. As learners implement their learning strategies, the system assists in monitoring the learning situation and context. Through scaffolding, the system proactively guides learners in searching for what they need and informs them of what they should learn. The Monitoring module oversees the learner's behaviors through the Analysis module in Metacognition Support via the Learner Interface. Monitoring decisions are made by analyzing the prediction model and learner model to determine if there are significant changes in the models. The Monitoring module also recognizes a learner's learning pace and provides corresponding feedback.

Learning Performance. Specific techniques are employed to investigate the learning performance monitoring process. One example is *knowledge tracing*. In a study by Kowalski et al. (2014) aimed at enhancing a language-teaching system, the researchers analyzed system data where students transcribed Chinese words to Roman characters. Different models were created to estimate the probability of correctness for students' answers to a syllable, considering the initial, final, and tone components. *Knowledge tracing* through the Hidden Markov Model (HMM) was employed, illustrating the probability of a state transition in a time series. The HMM technique demonstrated efficient processing time for knowledge tracing, crucial in designing personalized learning support systems. Knowledge modeling, along with tracing, could be essential for the monitoring process. Lester et al. (2013) proposed the use of the Dynamic Bayesian Network for knowledge modeling and tracing during interactive narrative experiences in an educational game. This model helped dynamically update probabilistic beliefs about a learner's understanding and knowledge of the learning content, estimating probability values representing students' knowledge, which were used to predict actual performance on posttests.

Learning Preference. Learning preferences can serve as predictors of learning performance. Dorça et al. (2013) suggested the use of reinforcement learning for automatically detecting learners' preferences. Assuming that the four types of students' learning preferences (i.e., reflective, intuitive, verbal, and sequential) change over time, their learning support system collects learners' performance values and calculates the reinforcement value. For instance, higher performance updates the reinforcement value, and vice versa. Stronger reinforcement increases the probability of the current learning preference, while lower reinforcement reduces the probability of the current preference. The approach successfully detected, monitored, and adjusted students' learning preferences to support their learning.

Evaluation. In addition to performance and preference monitoring, learning paths and behaviors require evaluation.

Learning Paths and Behavior. Chen (2008) delved into personalized learning paths in an e-learning system, where each course comprised elements like introductions, pretests, topics, summary modules, and posttests. Learners' paths (e.g., topic sequence, test types, the difficulty of topics) varied depending on the course sequence. They represented the course sequence as a *chromosome* in the Genetic Algorithm, generating personalized learning paths through reproduction, crossover, and mutation processes. Based on pretest results, the system suggested appropriate difficulty levels until the student successfully completed their learning. Similarly, mining techniques can assess learning behavior. Taub and Azevedo (2018) analyzed students' metacognitive behavior in a game-based learning environment. Grouping participants based on emotions and game performance, the study identified four groups: less efficient–low emotions, less efficient–high emotions, more efficient-low emotions, and more efficient-high emotions. Learning patterns were extracted from log files, indicating whether students' game performance was relevant, partially relevant, or irrelevant to their learning. Each student's sequence of these relevancies within the gameplay, such as partially relevant-irrelevant to irrelevant-relevant, served as input data for Sequential Pattern Mining analysis. After identifying noticeable patterns, Differential Sequence Mining was employed to check for significant sequence differences between groups. It was clear from these methods that the four students' groups' hypothesis-testing behavior patterns were not exactly the same. Thiswasn't clear from their first attempt at multivariate analysis of variance.

3.1.2. Part 2. Metacognition Support

The purpose of the previous part was to provide timely scaffolds for learners, leading to the Metacognition Support module, which consists of two submodules: the Analysis and Scaffolding modules.

Analysis.The Analysis module can be designed with data preprocessing functions for textual, behavioral, and contextual/environmental data. Learners' behavioral data undergoes cleaning (pre-processing) and analysis in this module, with the information then transmitted to appropriate submodules in the Metacognitive Process module.

Textual Data. Natural language could serve as a primary source of learner data. Ezen-Can and Boyer (2015) proposed a clustering approach to unsupervised NLP, collecting data on learner-tutor dialogs during online collaboration to solve computer programming problems. Following NLP steps (i.e., tokenization, parts of speech tagging, stemming, and presenting special entities), they applied various modeling techniques for clustering.

Schneider and Pea (2015) formulated their N-gram Model by calculating unigram, bigram, and trigram probabilities, revealing frequently used words in participants' discussions. The researchers assessed discussion coherence using Cosine Similarity, identifying how much a student group discussed a topic by building on their conversation partner's ideas and arguments. They successfully identified differences in coherence scores between student groups. Similar approaches can be found in research on online discussions (e.g., (Albatayneh, Ghauth, & Chua, 2018; Sullivan & Keith, 2019)).

Behavior Data. Harley et al. (2013) explored the existence of different student clusters and the relationship between learner behavior in an intelligent system and their performance to strengthen adaptive support. Data, including system-related (e.g., mouse clicks, keyboard entries, facial expressions, diagrams drawn, eye-tracking information) and survey data, was collected from a system teaching the human circulatory system. Using Expectation-Maximization, the researchers identified three distinct clusters based on twelve selected features. This technique was also employed to identify learners' voices, as seen in learner authentication systems (Kamaraj, Nidhyananthan, & Sundaram, 2019).

Context and Environment. Regardless of the learner's behavior and performance, the learning context might undergo changes. The dynamic interactions between this change and the learner should be analyzed; however, few studies have been conducted. Further research is required.

Scaffolding. Scaffolding can be offered to learners with the consideration of the learning content, performance, and motivation.

Learning Content. Morsy and Karypis (2019) devised a course recommendation framework using students' grades. They defined a good course (grade equal to or higher than GPA) and a bad course. Each student's course information was converted to a previous-subsequent co-occurrence frequency matrix for Singular Value Decomposition (SVD). The system recommends good courses, aiming to help students maintain or improve their overall grades. SVD demonstrated reasonable performance, particularly in predicting good courses. If learning resources are well-categorized as a form of formal instructional courses, resource recommendation systems could be beneficial in supporting learners.

Cetintas et al. (2010) proposed a classification model to estimate the difficulty level of math problems. They used an SVM classifier for the initial classification, considering individual features of math problem sentences. The individual features comprised a bag-of-words representation; for instance, "Deep learning is machine learning" can be represented as {"deep": 1, "learning": 2, "is.": 1, "machine": 1}. The bag-of-words was used to train the SVM model to find the best hyperplane classifying problem difficulty levels, a fundamental aspect of learning content scaffolding.

Dascalu et al. (2015) developed a learning material recommender system based on learning style matching algorithms. The system analyzes the user profile (e.g., learner interest, education, nationality, and the result of learning style questionnaires) and provides recommendations for learning material shortcuts and learning tools depending on learners' profiles and preferences. Jaccard Similarity was employed to calculate the similarity between recommendation items. For example, when recommending a learning tool to a learner, the system computes the similarities (Jaccard index or coefficient) between tools liked by other learners. Learning content recommendation systems have also been explored in online learning systems contexts, such as LMS integration (De Medio, Limongelli, Sciarrone, & Temperini, 2020) and MOOCs (Massive Open Online Courses; Xiao, Wang, Jiang, and Li (2018)).

Performance. Scaffolding is necessary when the learner's underperformance or misunderstanding is detected. Pelaez et al. (2019) applied data mining to identify at-risk students using Latent Class Analysis. The researchers collected data that included students' demographics, admission (e.g., SAT [the Scholastic Assessment Test] scores, high school GPA [grade point average]), and academic records (e.g., final grades). They applied Latent Class Analysis with the Random Forest technique and identified three clusters of at-risk students, which can be used for performance-based scaffolding.

Motivation. Motivation scaffolding supports learners' self-efficacy, self-directed aspiration, and perceptions of the value of learning activities (Belland, Kim, & Hannafin, 2013). Since motivation scaffolding contributes to learner retention (Ludwig-Hardman & Dunlap, 2003), the system needs to identify the features impacting learner retention. Besides, learners' motivation does not solely stem from learning tasks but also from their self-regulation processes (Zimmerman, 2002). Still, few studies have explored how to scaffold learners' motivation through AI techniques.

3.2. Modeling in SLAA

SLAA processes incoming learner data from the Learner Interface and triggers the Metacognition Process and Metacognition Support with the help of the Learner Modeling and Prediction Modeling modules.

3.2.1. Learner Modeling

With the information that learning support systems collect, a learner model can be built to effectively support learners. Learner modeling is the process of developing a conceptual understanding of the learner and building an internal representation of the learner efficiently to support their learning accordingly. The Learner Modeling module establishes a learner model through learner profiling, identifying what should be measured and how latent variables are related to the learner's SRL. The learner profiling process includes the measurement of the learner's knowledge, performance, cognitive and metacognitive levels, learning strategies, habits, motivation levels, affective aspects, and more. The learner data is stored in a database component, which is utilized to enable the system to build an initial learner model and update the model if needed. The learner model is adaptive depending on the learner's progress. To implement adaptability, the system needs to constantly update the learner model. Because knowledge representation can be a focus area of learner modeling, the result of a learner model can be an approximate qualitative representation of a learner's knowledge and skill level.

Learning Performance Profiling. *K-Means Clustering.* Ferguson and Clow (2015) investigated the patterns of learner engagement within MOOCs. They assigned students' activities to be *on track*, *behind*, *auditing*, and *out* based on their assessment submission time each week. Using K-Means Clustering, the researchers found distinct clusters, such as learners who completed the most tasks and those who explored videos only. Similarly, Lee and Tan (2017) and Vaessen et al. (2014) also utilized K-Means Clustering in their learning analysis research, focusing on how learners' ideas in discussion affected learning behaviors. These examples show how a clustering approach could be adopted for learner modeling with performance profiling.

HLM. Miller et al. (2015) used HLM (Hierarchical Linear Modelling) as their data consisted of learners' scores from different learning objects, which are nested (i.e., hierarchical data) within individual learners. From an LMS and student surveys, they collected eighty-one independent variables and used the final assessment as the dependent variable. Their approach identified demographic variables (e.g., college major and placement test) and self-regulation variables (e.g., control of learning beliefs, intrinsic goal orientation) as salient when predicting learner performance.

Learning Path and Pattern Profiling. Performance profiling shows a series of learners' performance statuses but does not reveal how their performances were achieved. Learning paths and patterns can explain it.

Hierarchical Clustering. Using a computational method to identify learners' behavioral changes, Boroujeni and

Dillenbourg (2019) analyzed learners' interaction log data (e.g., lecture video access, assessment submission) of a MOOC in undergraduate engineering. Focusing on the learners' behavior sequences, their Hierarchical Clustering analysis yielded distinct clusters. Hao et al. (2015) also used Hierarchical Clustering to identify learning patterns in a game-based learning environment.

Rasch Model. Waters et al. (2014) investigated learner collaboration patterns using the data of students' right-and-wrong responses. In their Rasch Model, a learner is characterized by a learner ability variable (i.e., single latent ability parameter), and the difficulty of questions is modeled. The model indicated the probability of a learner's correct answers. The performance on a large number of consecutive simple tasks not only demonstrated learners' achievements but also generated learning patterns characterized by a probabilistic model.

Process Mining. Sonnenberg and Bannert (2016) investigated the sequence of students' learning activities measured by a think-aloud method. Undergraduate students participated in online learning, which provided metacognitive prompts to support students' SRL. They identified both effective and non-effective metacognitive prompts. Using Process Mining, they found the sequential order of learning activities that provides detailed information about learning behaviors. Since the learning path follows a sequential order, Process Mining would be effective for constructing a learner model that incorporates learning path factors.

Network Analysis. Fincham et al. (2018) investigated the student discussion data collected from online courses. They employed the concept of social ties (i.e., depending on students' concept definitions, it could involve co-participation or direct replies in discussion) in their Social Network Analysis. Based on the identified social ties, they examined discussion nodes (e.g., degree centrality, closeness, betweenness) and network-level structures (e.g., density, diameter, path length). The results revealed that different definitions of social ties produced distinctions in the structural aspects of discussion networks. Similarly, Gruzd et al. (2016) used Social Network Analysis to explore learner discussions in social media, revealing interaction patterns that could be utilized for learning pattern profiling.

Learners' chatter information can reveal their learning paths and patterns. Siebert-Evenstone et al. (2017) collected student chat log data in a collaborative online learning environment, segmenting it by utterance. Using Epistemic Network Analysis to analyze the co-occurrence of concepts within discourse data, they identified the structure of connections in the coded student chat data. Another type of network analysis method examines learning behaviors. Hecking et al. (2014) investigated patterns of learners' resource usage (e.g., videos, scientific literature) collected from the event log in online courses. They used Clique Percolation to identify subgroups and found different resource usage patterns between different types of online courses. This method is useful for identifying patterns of learner-system interaction, a crucial component of learning profiling.

Advanced Statistical Techniques. Pattern analysis is not just for learners but can also be applied to instructors. Xu and Recker (2011) investigated clusters of teacher behaviors (e.g., browsing library resources, creating materials, and sharing activities) when using a digital library service. The researchers used Latent Class Analysis, providing different types of probability statistics, including the variables' significance. They identified multiple clusters of teachers, such as isolated users, goal-oriented teachers, and classroom practitioners.

When building a learner model, emotional aspects can be considered. Althoff et al. (2016) investigated psychotherapy counseling conversations about crisis intervention (e.g., anxiety, depression, suicidal thoughts) through text messages. Their dataset includes over 3.2 million messages. They used HMM to identify progression through counseling stages, specifically to capture the dialogue structure. The results revealed five stages of conversation progress: introduction, main issue and clarification request, problem discussion, actionable strategies, and wrap-up. Additionally, for time-varying data, such as learners' decisions in online learning environments, researchers have used Stochastic Gradient Descent for data analysis (e.g., Kassak et al. (2016)). Some studies have also employed temporal and sequential data processing methods, such as analyzing learner behaviors of body motions, gestures, and gaze (Andrade et al., 2017; Andrade, Delandshere, & Danish, 2016), student behavior in MOOCs (Geigle & Zhai, 2017), and learner interactions in intelligent systems (Shen et al., 2018).

Cognitive Strategy Profiling. *Ripley's K Function.* Mallavarapu et al. (2015) analyzed students' learning activities for spatial reasoning problems in a game. Ripley's K Function was employed to handle the spatial metric, quantifying the density of points at varying scales of distance. They examined students' exploration of two-dimensional spatial patterns (i.e., spatial strategies) in the game, with each solution represented by Ripley's metric. Learners' cognitive strategies were reflected in their problem-solving behavior in reasoning problems.

Bayesian Knowledge Tracing. Cui et al. (2019) utilized Bayesian Knowledge Tracing to analyze the student response process in a game-based assessment environment. They identified skills that can be learned in the game and created a model that updates the probability of mastering a skill using student answers. The technique revealed changes in a learner's status of skill or knowledge mastery over time during gameplay.

SRL Profiling. In learning environments, a significant portion of learning depends on learners' decisions, which are related to their SRL skills.

Decision Tree. Sabourin et al. (2013) analyzed middle school students' SRL behaviors in a game-based learning environment and classified the students into SRL categories. They used multiple features, including demographics, pretest scores, personality surveys, goal orientations, emotion surveys, the number of gameplays, and off-task behaviors. Each student's SRL levels were categorized into low, medium, and high levels. The data was trained with supervised ML techniques such as Naïve Bayes, Decision Tree, SVM, Logistic Regression, and Neural Network. As a result of cross-validation, Decision Tree showed the highest predictive accuracy. Adopting this technique in the learning support system could automatically identify students with low SRL levels.

3.3. Prediction Modeling

Based on a learner model and behaviors, an ideal system can provide the learner with information on new learning material, appropriate learning paths, timely feedback, comments, guidance, and scaffolds. This involves prediction models achievable through the Prediction Modeling process in SLAA.

Learner Performance. (*SVM*). Yoo and Kim (2014) examined how to predict student performance using the characteristics of online discussion in undergraduate computer science courses. The students participated in discussions for their programming group projects. Using Speech Acts (Searle, Kiefer, & Bierwisch, 1980), they defined the information roles that students play in their discussion as sink (seek information) and source (provide information). They identified different types of features, such as message-level and thread-level, and used different techniques for Sink and Source classifiers. Their results show that SVM performance is better and less sensitive to the number of features than other ML techniques.

Deep Learning. Mao (2018) compared Deep Learning with Bayesian Knowledge Tracing using data from intelligent systems. The researchers observed that Deep Learning exhibited the highest accuracy in predicting learning gains. However, Bayesian Knowledge Tracing outperformed the Deep Learning approach when predicting the students' posttest scores.

Learner Retention. Learners' retention rates are significant in online learning environments due to high course failures and dropout rates.

Naïve Bayes. Students at the College of First-Year Studies used an e-portal system, which Aguiar et al. (2014) investigated. The data label for student retention was predicted through features including admission intent, SAT scores, GPA, gender, ethnicity, and e-portfolio logins, submissions, and hits. When using the data set of academic and engagement information, Naïve Bayes performed the best among classification methods (e.g., Decision Tree, Logistic Regression, Random Forests). Specifically, their Naïve Bayes model correctly predicted 42 of the 48 students who were kept in school.

Random Forest. Spoon et al. (2016) examined students' final exam scores, course completion, LMS data, and institutional data. They utilized Random Forest to identify students who are at risk of failing a course. In their model, passing and non-passing grades were classified into trees, with final exam scores serving as the y-values in regression trees. If a student's successful course completion is predicted by less than 50% of the trees in their *forest*, the student can be considered at-risk. The researchers argue that their model's ROC (Receiver Operating Characteristic) curve, a graphical plot that reveals the classification power by representing the true positive rate against the false positive rate, shows better performance than the institution's traditional approach that uses quiz or exam scores.

K-Nearest Neighbor. Gray et al. (2016) investigated how to predict at-risk undergraduate students using data collected from the registration process, a learner profiling survey of first-year students, and their first-year academic performance. They looked at how accurate different modeling methods were (e.g., Naïve Bayes, Decision Tree, Logistic Regression, Back Propagation Neural Network, Support Vector Machine). K-Nearest Neighbor had the best accuracy for a certain dataset when predicting students who might be at risk.

4. DISCUSSION

Different types of AI techniques used in education research were identified, revealing the front- and back-end frameworks, and a learning design model, SLAA, was proposed. The explanation of the AI or ML techniques can be useful for a researcher who intends to adopt a specific technique for their studies. With the front- and back-end framework, researchers can differentiate their approaches when using AI in education.

Researchers can adopt SLAA to design their learning support system with specific AI techniques. Still, there are some issues we were not able to address in our model. First, for the learner-AI interaction, the interface's role involves noticing, interpreting, and responding; thus, AI-infused learning support systems need to understand natural human languages. Voice recognition and text-to-speech technologies have evolved rapidly in recent history. It seems that the learner interface could adopt these technologies to support conversational interactions between the learner and systems. Modeling learners' conversation content, patterns, and intention could be crucial for scaling natural language understanding of agent-based approaches, which recognize goals and tasks and interact with the environment and learners. Second, along with the cognitive and metacognitive scaffolds, the learner's affective aspects (e.g., emotion, sensation, and feeling) should be considered when designing an AI-infused learning support system. Affective aspects play a significant role in learning (Immordino-Yang & Damasio, 2007) and impact learning, such as learning judgment, learning interest and attention, motivation, knowledge retention, problem-solving skills, and decision-making (Blanchard, Volfson, Hong, & Lajoie, 2009). The learner's affective aspects can be taken into consideration in the prediction modeling to maximize the probability of learning success. Lastly, learning support systems need to support a group of learners' goal settings, analyze collaborative performances, and encourage the active cooperation of the learning community. Along with individual learner modeling, group learning or collaborative learning modeling needs to be considered. Future systems need to structure the flow of collaborative learners' knowledge. Learning environments can also adopt social learning networks, which support knowledge sharing as a type of learning community (Spoelstra et al., 2014), as human-AI collaboration will likely increase.

It is anticipated that AI will tackle a wider range of learning problems and offer effective scaffolding. However, it should also be noted that concerns and possible pessimism exist about embracing the future of AI and undertaking AI-led education initiatives.

REFERENCES

Aguiar, E., Ambrose, G. A. A., Chawla, N. V., Goodrich, V., & Brockman, J. (2014). Engagement vs performance: Using electronic portfolios to predict first semester engineering student persistence. *Journal of Learning Analytics*, *1*(3), 7-33. https://doi.org/10.18608/jla.2014.13.3

Albatayneh, N. A., Ghauth, K. I., & Chua, F.-F. (2018). Utilizing learners' negative ratings in semantic content-based recommender system for e-learning forum. *Journal of Educational Technology & Society*, *21*(1), 112-125.

Althoff, T., Clark, K., & Leskovec, J. (2016). Large-scale analysis of counseling conversations: An application of natural language processing to mental health. *Transactions of the Association for Computational Linguistics*, *4*, 463-476. https://doi.org/10.1162/tacl_a_00111

Andrade, A., Danish, J. A., & Maltese, A. V. (2017). A measurement model of gestures in an embodied learning environment: Accounting for temporal dependencies. *Journal of Learning Analytics*, *4*(3), 18–46. https://doi.org/10.18608/jla.2017.43.3

Andrade, A., Delandshere, G., & Danish, J. A. (2016). Using multimodal learning analytics to model student behavior: A systematic analysis of epistemological framing. *Journal of Learning Analytics*, *3*(2), 282-306. https://doi.org/10.18608/jla.2016.32.14

Beck, J., Stern, M., & Haugsjaa, E. (1996). Applications of AI in education. *Crossroads*, *3*(1), 11-15.

Belland, B. R., Kim, C., & Hannafin, M. J. (2013). A framework for designing scaffolds that improve motivation and cognition. *Educational Psychologist*, *48*(4), 243-270.

Blanchard, E. G., Volfson, B., Hong, Y.-J., & Lajoie, S. P. (2009). Affective artificial intelligence in education: From detection to adaptation. In V. Dimitrova, R. Mizoguchi, B. du Boulay & A. Grasser (Eds.), Artificial intelligence in education. Building learning systems that care: From knowledge representation to affective modelling. In (pp. 81-88): IOS Press. https://doi.org/10.3233/978-1-60750-028-5-81.

Boroujeni, M. S., & Dillenbourg, P. (2019). Discovery and temporal analysis of MOOC study patterns. *Journal of Learning Analytics*, *6*(1), 16—33. https://doi.org/10.18608/jla.2019.61.2

Cetintas, S., Si, L., Xin, Y. P., Zhang, D., Park, J. Y., & Tzur, R. (2010). A joint probabilistic classification model of relevant and irrelevant sentences in mathematical word problems. *Journal of Educational Data Mining*, *2*(1), 83-101. https://doi.org/10.5281/zenodo.3554741

Chen, C.-M. (2008). Intelligent web-based learning system with personalized learning path guidance. *Computers & Education*, *51*(2), 787-814. https://doi.org/10.1016/j.compedu.2007.08.004

Cui, Y., Chu, M.-W., & Chen, F. (2019). Analyzing student process data in game-based assessments with Bayesian knowledge tracing and dynamic Bayesian networks. *Journal of Educational Data Mining*, *11*(1), 80-100. https://doi.org/10.5281/zenodo.3554751

Dascalu, M.-I., Bodea, C.-N., Moldoveanu, A., Mohora, A., Lytras, M., & de Pablos, P. O. (2015). A recommender agent based on learning styles for better virtual collaborative learning experiences. *Computers in Human Behavior*, *45*, 243-253. https://doi.org/10.1016/j.chb.2014.12.027

De Medio, C., Limongelli, C., Sciarrone, F., & Temperini, M. (2020). MoodleREC: A recommendation system for creating courses using the moodle e-learning platform. *Computers in Human Behavior*, *104*, 106168. https://doi.org/10.1016/j.chb.2019.106168

Department of the Army. (2011). *The U.S. Army learning concept for 2015. Fort eustis: Training and doctrine command.* Retrieved from https://adminpubs.tradoc.army.mil/pamphlets/TP525-8-2.pdf

Desmarais, M. C., & Baker, R. S. d. (2012). A review of recent advances in learner and skill modeling in intelligent learning environments. *User Modeling and User-Adapted Interaction, 22*(1-2), 9-38. https://doi.org/10.1007/s11257-011-9106-8

Dorça, F. A., Lima, L. V., Fernandes, M. A., & Lopes, C. R. (2013). Comparing strategies for modeling students learning styles through reinforcement learning in adaptive and intelligent educational systems: An experimental analysis. *Expert Systems with Applications, 40*(6), 2092-2101. https://doi.org/10.1016/j.eswa.2012.10.014

Ezen-Can, A., & Boyer, K. E. (2015). Understanding student language: An unsupervised dialogue act classification approach. *Journal of Educational Data Mining, 7*(1), 51-78. https://doi.org/10.5281/zenodo.3554707

Ezen-Can, A., Grafsgaard, J. F., Lester, J. C., & Boyer, K. E. (2015). *Classifying student dialogue acts with multimodal learning analytics.* Paper presented at the Proceedings of the Fifth International Conference on Learning Analytics and Knowledge, New York, NY: ACM.

Ferguson, R., & Clow, D. (2015). Consistent commitment: Patterns of engagement across time in massive open online courses (MOOCs). *Journal of Learning Analytics, 2*(3), 55-80. https://doi.org/10.18608/jla.2015.23.5

Fincham, E., Gašević, D., & Pardo, A. (2018). From social ties to network processes: Do tie definitions matter? *Journal of Learning Analytics, 5*(2), 9–28. https://doi.org/10.18608/jla.2018.52.2

Gardner, J., & Brooks, C. (2018). Evaluating predictive models of student success: Closing the methodological gap. *Journal of Learning Analytics, 5*(2), 105-125. https://doi.org/10.18608/jla.2018.52.7

Geigle, C., & Zhai, C. (2017). Modeling student behavior with two-layer hidden markov models. *Journal of Educational Data Mining, 9*(1), 1-24.

Gray, G., McGuinness, C., Owende, P., & Hofmann, M. (2016). Learning factor models of students at risk of failing in the early stage of tertiary education. *Journal of Learning Analytics, 3*(2), 330-372. https://doi.org/10.18608/jla.2016.32.20

Gruzd, A., Paulin, D., & Haythornthwaite, C. (2016). Analyzing social media and learning through content and social network analysis: A faceted methodological approach. *Journal of Learning Analytics, 3*(3), 46-71. https://doi.org/10.18608/jla.2016.33.4

Hao, J., Shu, Z., & von Davier, A. (2015). Analyzing process data from game/scenario-based tasks: An edit distance approach. *Journal of Educational Data Mining, 7*(1), 33-50. https://doi.org/10.5281/zenodo.3554705

Harley, J. M., Bouchet, F., Trevors, G. J., & Azevedo, R. (2013). Clustering and profiling students according to their interactions with an intelligent tutoring system fostering self-regulated learning. *Journal of Educational Data Mining, 5*(1), 104-146. https://doi.org/10.5281/zenodo.3554613

Hecking, T., Ziebarth, S., & Hoppe, H. U. (2014). Analysis of dynamic resource access patterns in online courses. *Journal of Learning Analytics, 1*(3), 34-60. https://doi.org/10.18608/jla.2014.13.4

Immordino-Yang, M. H., & Damasio, A. (2007). We feel, therefore we learn: The relevance of affective and social neuroscience to education. *Mind, Brain, and Education, 1*(1), 3-10. https://doi.org/10.1111/j.1751-228X.2007.00004.x

Kai, S., Almeda, M. V., Baker, R. S., Heffernan, C., & Heffernan, N. (2018). Decision tree modeling of wheel-spinning and productive persistence in skill builders. *Journal of Educational Data Mining, 10*(1), 36-71. https://doi.org/10.5281/zenodo.3344810

Kamaraj, A., Nidhyananthan, S., & Sundaram, K. (2019). Voice biometric for learner authentication: Biometric authentication. In Biometric Authentication in Online Learning Environments. In (pp. 150-181): IGI Global. https://doi.org/10.4018/978-1-5225-7724-9.ch007.

Kassak, O., Kompan, M., & Bielikova, M. (2016). Student behavior in a web-based educational system: Exit intent prediction. *Engineering Applications of Artificial Intelligence, 51*, 136-149. https://doi.org/10.1016/j.engappai.2016.01.018

Knight, S., Buckingham Shum, S., Ryan, P., Sándor, Á., & Wang, X. (2018). Designing academic writing analytics for civil law student self-assessment. *International Journal of Artificial Intelligence in Education, 28*, 1-28. https://doi.org/10.1007/s40593-016-0121-0

Kotsiantis, S. B., Zaharakis, I. D., & Pintelas, P. E. (2006). Machine learning: A review of classification and combining techniques. *Artificial Intelligence Review, 26*, 159-190. https://doi.org/10.1007/s10462-007-9052-3

Kowalski, J., Zhang, Y., & Gordon, G. J. (2014). Statistical modeling of student performance to improve Chinese dictation skills with an intelligent tutor. *Journal of Educational Data Mining, 6*(1), 3-27. https://doi.org/10.5281/zenodo.3554679

Lee, A. V. Y., & Tan, S. C. (2017). Promising ideas for collective advancement of communal knowledge using temporal analytics and cluster analysis. *Journal of Learning Analytics, 4*(3), 76-101. https://doi.org/10.18608/jla.2017.43.5

Lester, J. C., Ha, E. Y., Lee, S. Y., Mott, B. W., Rowe, J. P., & Sabourin, J. L. (2013). Serious games get smart: Intelligent game-based learning environments. *AI Magazine, 34*(4), 31-45. https://doi.org/10.1609/aimag.v34i4.2488

Ludwig-Hardman, S., & Dunlap, J. C. (2003). Learner support services for online students: Scaffolding for success. *International Review of Research in Open and Distributed Learning, 4*(1), 1-15. https://doi.org/10.19173/irrodl.v4i1.131

Mahzoon, M. J., Maher, M. L., Eltayeby, O., Dou, W., & Grace, K. (2018). A sequence data model for analyzing temporal patterns of student data. *Journal of Learning Analytics, 5*(1), 55–74. https://doi.org/10.18608/jla.2018.51.5

Mallavarapu, A., Lyons, L., Shelley, T., Minor, E., Slattery, B., & Zellner, M. (2015). Developing computational methods to measure and track learners' spatial reasoning in an open-ended simulation. *Journal of Educational Data Mining, 7*(2), 49-82. https://doi.org/10.5281/zenodo.3554669

Mao, Y. (2018). Deep learning vs. Bayesian knowledge tracing: Student models for interventions. *Journal of Educational Data Mining, 10*(2), 28-54. https://doi.org/10.5281/zenodo.3554691

Miller, L. D., Soh, L.-K., Samal, A., Kupzyk, K., & Nugent, G. (2015). A comparison of educational statistics and data mining approaches to identify characteristics that impact online learning. *Journal of Educational Data Mining, 7*(3), 117-150. https://doi.org/10.5281/zenodo.3554731

Mirriahi, N., Liaqat, D., Dawson, S., & Gašević, D. (2016). Uncovering student learning profiles with a video annotation tool: Reflective learning with and without instructional norms. *Educational Technology Research and Development, 64*, 1083-1106. https://doi.org/10.1007/s11423-016-9449-2

Morsy, S., & Karypis, G. (2019). Will this course increase or decrease your GPA? towards grade-aware course recommendation. *Journal of Educational Data Mining, 11*(2), 20-46.

Muis, K. R., Psaradellis, C., Chevrier, M., Di Leo, I., & Lajoie, S. P. (2016). Learning by preparing to teach: Fostering self-regulatory processes and achievement during complex mathematics problem solving. *Journal of Educational Psychology, 108*(4), 474-492. https://doi.org/10.1037/edu0000071

Oh, Y. E., Song, D., & Hong, H. (2020). Interactive computing technology in anti-bullying education: The effects of conversation-bot's role on K-12 students' attitude change toward bullying problems. *Journal of Educational Computing Research, 58*(1), 200-219. https://doi.org/10.1177/0735633119839177

Pelaez, K., Levine, R., Fan, J., Guarcello, M., & Laumakis, M. (2019). Using a latent class forest to identify at-risk students in higher education. *Journal of Educational Data Mining, 11*(1), 18-46. https://doi.org/10.5281/zenodo.3554747

Pintrich, P. R. (1999). The role of motivation in promoting and sustaining self-regulated learning. *International Journal of Educational Research, 31*(6), 459-470. https://doi.org/10.1016/S0883-0355(99)00015-4

Russell, S., & Norvig, P. (2010). *Artificial intelligence: A modern approach* (3rd ed.). Upper Saddle River: Prentice-Hall.

Sabourin, J. L., Shores, L. R., Mott, B. W., & Lester, J. C. (2013). Understanding and predicting student self-regulated learning strategies in game-based learning environments. *International Journal of Artificial Intelligence in Education, 23*, 94-114. https://doi.org/10.1007/s40593-013-0004-6

Schneider, B., & Pea, R. (2015). Does seeing one another's gaze affect group dialogue? A computational approach. *Journal of Learning Analytics, 2*(2), 107-133. https://doi.org/https://doi.org/10.18608/jla.2015.22.9

Searle, J. R., Kiefer, F., & Bierwisch, M. (1980). Speech act theory and pragmatics. In (Vol. 10, pp. 205-220). Dordrecht: D. Reidel.

Serban, I. V., Lowe, R., Henderson, P., Charlin, L., & Pineau, J. (2018). A survey of available corpora for building data-driven dialogue systems: The journal version. *Dialogue & Discourse, 9*(1), 1-49. https://doi.org/10.5087/dad.2018.101

Shen, S., Mostafavi, B., Barnes, T., & Chi, M. (2018). Exploring induced pedagogical strategies through a Markov decision process framework: Lessons learned. *Journal of Educational Data Mining, 10*(3), 27-68. https://doi.org/10.5281/zenodo.3554713

Shermis, M. D., & Burstein, J. C. (2003). *Automated essay scoring: A cross-disciplinary perspective.* Hillsdale, NJ: Lawrence Erlbaum.

Siebert-Evenstone, A. L., Irgens, G. A., Collier, W., Swiecki, Z., Ruis, A. R., & Shaffer, D. W. (2017). In search of conversational grain size: Modeling semantic structure using moving stanza windows. *Journal of Learning Analytics, 4*(3), 123–139. https://doi.org/10.18608/jla.2017.43.7

Song, D. (2018). Learning analytics as an educational research approach. *International Journal of Multiple Research Approaches, 10*(1), 102-111. https://doi.org/10.29034/ijmra.v10n1a6

Song, D., Rice, M., & Oh, E. Y. (2019). Participation in online courses and interaction with a virtual agent. *International Review of Research in Open and Distributed Learning, 20*(1), 43-62. https://doi.org/10.19173/irrodl.v20i1.3998

Sonnenberg, C., & Bannert, M. (2016). Evaluating the impact of instructional support using data mining and process mining: A micro-level analysis of the effectiveness of metacognitive prompts. *Journal of Educational Data Mining, 8*(2), 51-83. https://doi.org/10.5281/zenodo.3554597

Spoelstra, H., Van Rosmalen, P., & Sloep, P. (2014). Toward project-based learning and team formation in open learning environments. *Journal of Universal Computer Science, 20*(1), 57-76. https://doi.org/10.3217/jucs-020-01-0057

Spoon, K., Beemer, J., Whitmer, J. C., Fan, J., Frazee, J. P., Stronach, J., . . . Levine, R. A. (2016). Random forests for evaluating pedagogy and informing personalized learning. *Journal of Educational Data Mining, 8*(2), 20-50. https://doi.org/10.5281/zenodo.3554595

Sullivan, F. R., & Keith, P. K. (2019). Exploring the potential of natural language processing to support microgenetic analysis of collaborative learning discussions. *British Journal of Educational Technology, 50*(6), 3047-3063. https://doi.org/10.1111/bjet.12875

Taherkhani, A., & Malmi, L. (2013). Beacon-and schema-based method for recognizing algorithms from students' source code. *Journal of Educational Data Mining, 5*(2), 69-101. https://doi.org/10.5281/zenodo.3554635

Taub, M., & Azevedo, R. (2018). Using sequence mining to analyze metacognitive monitoring and scientific inquiry based on levels of efficiency and emotions during game-based learning. *Journal of Educational Data Mining, 10*(3), 1-26. https://doi.org/10.5281/zenodo.3554711

Vaessen, B. E., Prins, F. J., & Jeuring, J. (2014). University students' achievement goals and help-seeking strategies in an intelligent tutoring system. *Computers & Education, 72*, 196-208. https://doi.org/10.1016/j.compedu.2013.11.001

Waters, A. E., Studer, C., & Baraniuk, R. G. (2014). Collaboration-type identification in educational datasets. *Journal of Educational Data Mining, 6*(1), 28-52. https://doi.org/10.5281/zenodo.3554681

Wolters, C., & Taylor, D. (2012). A self-regulated learning perspective on student engagement. In S. L. Christenson, A. Reschly, & C. Wylie (Eds.), Handbook of research on student engagement. In (pp. 635–651): Springer. https://doi.org/10.1007/978-1-4614-2018-7_30.

Xiao, J., Wang, M., Jiang, B., & Li, J. (2018). A personalized recommendation system with combinational algorithm for online learning. *Journal of Ambient Intelligence and Humanized Computing, 9*, 667-677. https://doi.org/10.1007/s12652-017-0466-8

Xu, B., & Recker, M. (2011). Understanding teacher users of a digital library service: A clustering approach. *Journal of Educational Data Mining, 3*(1), 1-28. https://doi.org/10.5281/zenodo.3554701

Yoo, J., & Kim, J. (2014). Can online discussion participation predict group project performance? Investigating the roles of linguistic features and participation patterns. *International Journal of Artificial Intelligence in Education, 24*, 8-32. https://doi.org/10.1007/s40593-013-0010-8

Zimmerman, B. J. (1990). Self-regulated learning and academic achievement: An overview. *Educational Psychologist, 25*(1), 3-17. https://doi.org/10.1207/s15326985ep2501_2

Zimmerman, B. J. (2002). Becoming a self-regulated learner: An overview. *Theory into Practice, 41*(2), 64-70. https://doi.org/10.1207/s15430421tip4102_2

Online Science Publishing *is not responsible or answerable for any loss, damage or liability, etc. caused in relation to/arising out of the use of the content. Any queries should be directed to the corresponding author of the article.*