Article:
Huang, J. orcid.org/0000-0002-0905-0915, Yang, K., Wang, Q. et al. (4 more authors) (2025) Bayesian deep multi-instance learning for student performance prediction based on campus big data. Neurocomputing, 647. 130538. ISSN 0925-2312

https://doi.org/10.1016/j.neucom.2025.130538

Bayesian Deep Multi-Instance Learning for Student Performance Prediction Based on Campus Big Data

Jiayang Huang[a,b], Keyi Yang[a,b], Quan Wang[a,b,*], Pengfei Yang[a,b], Ziling Ruan[a,b], Jiaxi Wang[a,b] and Zhi-Qiang Zhang[c,**]

[a] School of Computer Science and Technology, Xidian University, Xi'an, 710126, China
[b] The Key Laboratory of Smart Human-Computer Interaction and Wearable Technology of Shaanxi Province, Xi'an, 710126, China
[c] School of Electronic and Electrical Engineering, Institute of Robotics, Autonomous Systems and Sensing, University of Leeds, Leeds, U.K

ARTICLE INFO

Keywords:
Academic Performance
Educational Data Mining
Bayesian Neural Network
Multi-instance Learning
Deep Learning

ABSTRACT

Problem: Predicting student performance using campus big data and deep learning methods has emerged as a promising alternative to traditional psychological assessments, which are often delayed and subjective. However, existing deep learning approaches face significant challenges, including the sparsity of campus big data and susceptibility to overfitting. **Objective:** To address these issues, this study aims to develop a robust and effective framework for predicting student performance by overcoming data sparsity and overfitting problems. **Method:** We propose a novel Bayesian Deep Multi-Instance Learning (Bay-DeepMIL) framework. The method integrates multi-instance learning (MIL) to handle data sparsity and incorporates a Bayesian framework to treat network parameters as probabilistic distributions, enhancing robustness and mitigating overfitting. Additionally, a Bayesian multi-head attention mechanism is introduced to dynamically assign importance to different instances, improving the extraction of key information from multi-instance data. **Results:** Extensive experiments demonstrate that Bay-DeepMIL outperforms state-of-the-art methods in prediction accuracy. The framework also provides uncertainty estimates for each prediction, offering valuable confidence measures and decision support for educational stakeholders. **Conclusion:** The proposed Bay-DeepMIL framework not only advances the technical capabilities of predictive models but also provides practical tools for enhancing educational decision-making. This study underscores the effectiveness of integrating Bayesian inference with multi-instance learning to address core challenges such as data sparsity and model overfitting in student performance prediction.

1. Introduction

In traditional face-to-face higher education settings, it is challenging to address the diverse needs of students who come from varied backgrounds and may have limited resources (Fernández et al., 2023). Statistics from China's Ministry of Education reveal that nearly 20,000 students did not receive a degree in 2022 (Cao, 2023). Students often face stress, anxiety (Serra et al., 2020), procrastination (Wang et al., 2021), and even academic burnout (Liu et al., 2023), leading to significant academic challenges. While interventions like Mindfulness Meditation (Bamber & Morpeth, 2019) have been shown to improve performance, the slow feedback mechanisms in educational systems can delay support. Therefore, it's essential to develop more precise methods to proactively predict and improve student performance (Feng & Fan, 2024; Bai et al., 2021; Lu et al., 2021).

In the current higher education system, student performance prediction relies on psychological assessment data such as surveys and psychological tests (Zhao & Wang, 2023; Anthonysamy & Singh, 2023; Hanaysha et al., 2023). These methods have limitations due to the challenges associated with data collection, the influence of subjective factors, restrictions on update frequencies, and their generalizability. They often require specialized psychological expertise, frequent evaluations, and may be impacted by individual differences and situational variables. As universities continue to enhance their information technology infrastructures, an increasing number of institutions can automatically and comprehensively record and monitor student behavior and status via their data centers (Wu et al., 2020; Rivas et al., 2021). Consequently, methods for predicting student performance based on this extensive campus data have started to emerge.

Currently, two primary methods, traditional machine learning and deep learning (DL), are used to predict student performance using campus big data. Traditional machine learning methods are highly interpretable and computationally inexpensive. For instance, Phan et al. (Phan et al., 2023) collected structured data on students (sociodemographics, major and course registration information, grades, academic status) as well as unstructured data (students' textual feedback on each elective course) to predict student dropout. Matz et al. (Matz et al., 2023) apply elastic net and random forest algorithms to predict student retention using institutional, engagement, and combined feature sets, assessing their effectiveness with out-of-sample benchmark experiments against a non-informative baseline model. Cheng et al. (Cheng et al., 2024) utilize a comprehensive dataset of 33 attributes from Portuguese education institutions to predict and classify students' performance using machine learning methods, enhanced by metaheuristic algorithms. Christou et

*Corresponding author 1.
**Corresponding author 2.
huangjiayang@xidian.edu.cn (J. Huang); 21031211507@stu.xidian.edu.cn (K. Yang); qwang@xidian.edu.cn (Q. Wang); pfyang@xidian.edu.cn (P. Yang); 21031211775@stu.xidian.edu.cn (Z. Ruan); 22031212174@stu.xidian.edu.cn (J. Wang); z.zhang3@leeds.ac.uk (Z. Zhang)

al. (Christou et al., 2023) proposed a grammatical evolution-based feature selection method for radial basis function networks to predict students' future performance. Nevertheless, these models encounter performance bottlenecks when dealing with highly nonlinear and complex data patterns. However, these methods rely on feature engineering but can not learn complex or implicit patterns from the data, causing unsatisfactory prediction performances in some scenarios.

Deep learning methods do not require feature engineering and automatically identify and learn deep patterns and associations in the data through neural network models, therefore improving prediction performance.For instance, Chen et al. (Chen et al., 2023) proposed a model for predicting grades and failed subjects based on students' behavioral features, which accurately extracts key behavioral characteristics through a multiple self-attention mechanism. Rodriguez et al. (Rodríguez-Hernández et al., 2021) used artificial neural networks to predict student performance in higher education and analyzed the main predictors. Barany et al. (Baranyi et al., 2020) proposed deep neural networks to predict college student dropout, enhancing model interpretation with techniques like permutation importance and SHAP values. Yang et al. (Yang et al., 2020b) proposed one-channel and three-channel learning image recognition methods to transform student course involvement into images for early warning predictive analysis, demonstrating through experiments with 5235 students that these methods outperform traditional machine learning algorithms in identifying at-risk students, while also offering visual insights for personalized interventions. Nayani et al. (Nayani & P, 2023) proposed a hybrid deep learning model using optimized entropy rough set theory and a novel Galactic Rider Swarm Optimization algorithm to predict student performance.

However, existing deep learning approaches in the field still face several challenges: 1) Data sparsity: Different students have different course completion records, resulting in many null values in the data. This hinders the ability to accurately capture the correlation between students' academic performance and learning behaviors across different courses, thus reducing the prediction performance. 2) Model overfitting: Due to their numerous parameters and high complexity, current DL methods tend to overemphasize detailed features in student data, including noise and outliers, resulting in decreased prediction accuracy when models are applied to test data, i.e., the prediction models overfit.

Therefore, this study focuses on the problem of predicting whether a student is at risk of failing a course, based on their historical academic performance. The key challenge lies in the sparse and heterogeneous nature of educational data of students who enroll in different combinations of courses, leading to non-uniform and irregular records that traditional supervised learning models struggle to handle effectively. To address this, we formulate the task as a Multi-Instance Learning (MIL) problem, where each student is treated as a bag containing a variable number of course instances. However, due to the variability in bag sizes and potential noise in course-level features, standard MIL approaches are prone to overfitting or under-generalization. To overcome these issues, we propose a Bayesian Deep Multi-Instance Learning (Bay-DeepMIL) framework that combines deep feature extraction with variational Bayesian inference. The Bayesian component allows the model to capture uncertainty in parameters and improve robustness, especially in the presence of sparse and noisy academic data. The ultimate objective is to accurately identify students at risk of academic failure under real-world data conditions.

The main contributions of this paper include:

- Handling Data Sparsity: We introduce multi-instance learning (MIL) to address the sparsity of campus big data effectively. By treating each student as a "bag" of instances, our approach captures the relationships between different instances and improves feature extraction from sparse data.
- Mitigating Overfitting: We incorporate a Bayesian framework into the deep learning model, treating network parameters as probabilistic distributions rather than fixed values. This enhances the model's robustness to unknown data and reduces overfitting, leading to better generalization performance.
- Dynamic Attention Mechanism: We propose a Bayesian multi-head attention mechanism that dynamically allocates the importance of different instances within the multi-instance framework. This mechanism enhances the model's ability to extract key information from complex and noisy data.
- Uncertainty Estimation: Our framework provides uncertainty estimates for each prediction, offering valuable insights into the reliability of the results. This feature supports more informed decision-making by educational stakeholders.
- Comprehensive Evaluation: We conduct extensive experiments to validate the effectiveness of the proposed Bay-DeepMIL framework. The results demonstrate superior prediction performance compared to state-of-the-art methods, highlighting the practical utility of our approach.

The proposed Bay-DeepMIL framework advances the state-of-the-art in student performance prediction by addressing the challenges of data sparsity and overfitting. This study not only contributes to the technical development of predictive models but also provides practical tools for improving educational outcomes.

The remainder of this paper is arranged as follows: Section 2 reviews related work on multi-instance learning and Bayesian neural networks and highlights the limitations of existing approaches. Section 3 presents the proposed Bay-DeepMIL framework, including datasets, Bay-DeepMIL architecture, Bayesian multi-attention mechanism, and uncertainty estimation technique. Section 4 describes the results and discusses the performance of Bay-DeepMIL compared

to state-of-the-art methods. Section 5 concludes the paper and outlines future research directions.

2. Related work

This section presents related work, including both multi-instance learning and Bayesian neural networks.

2.1. Multi-instance learning

In the context of Multiple Instance Learning (MIL), data are organized into collections termed "bags," each containing a variable number of instances (Waqas et al., 2024). Within this framework, only labels for the bags are provided, with individual instance labels remaining undisclosed. These bags are typically bifurcated into two categories: positive and negative bags. The primary emphasis of most MIL methodologies revolves around the binary classification challenge, aiming to accurately assign labels to these bags.

Currently, MIL has been widely used in many fields such as student performance prediction (Ma et al., 2020), medical image analysis (Struski et al., 2024; Pérez-Cano et al., 2024; Kang et al., 2024), text categorization (Pal et al., 2022; Li et al., 2021), audio processing (Korkmaz & Boyacı, 2022), remote sensing (Li et al., 2022, 2023), and so on. For instance, Ma et al. (Ma et al., 2020) proposed a multi-instance multi-task learning method, MIML-Circle, for predicting the performance of students from different majors, which better represents the student sample and exploits the correlation between courses. Perez et al. (Pérez-Cano et al., 2024) proposed an end-to-end MIL model that trains a GP classifier along with a CNN backbone and attention mechanism, which improves the robustness and accuracy of bag prediction by optimizing feature extraction based on GP classification. Pal et al. (Pal et al., 2022) propose the use of graphs to model interactions between bags and employ a graph neural network (GNN) to facilitate end-to-end learning where uncertainty is introduced using a Bayesian framework, and finally validate the effectiveness of the approach on a 20-text dataset. Korkmaz et al. (Korkmaz & Boyacı, 2022) proposed a bag-level MNIST model for Voice Activity Detection (VAD) that implements MIL in the embedding layer of CNNs using Noisy-And pooling. Li et al. (Li et al., 2023) proposed a deep multi-instance convolutional neural network (DMCNN) model for disaster classification of high-resolution remote sensing images. The features are first extracted by CNN, then a prototype learning layer with a distance metric is introduced to map the extracted features into a series of instance feature prototypes with bagging levels, and two types of features are used for classification.

Despite these advancements, existing MIL methods face several limitations when applied to student performance prediction. First, while MIL can handle sparse data by treating each student as a bag of instances, existing methods often fail to effectively capture the complex interactions between instances due to the lack of robust feature extraction mechanisms. Second, many MIL models are prone to overfitting, especially when dealing with high-dimensional and noisy data, such as campus big data. This limits their generalization ability in real-world educational settings. Third, most MIL frameworks do not provide uncertainty estimates for their predictions, which are crucial for educational decision-makers to assess the reliability of the results.

2.2. Bayesian neural network

Bayesian neural networks (BNNs) impose prior distributions on weights and biases, as shown in Fig. 1. Combining Bayesian statistical methods with a neural network architecture allows the network to prevent overfitting and to quantify the uncertainty in its predictions (Liu et al., 2020; Chen et al., 2021; Tai et al., 2024; Deka et al., 2024; Bai & Chandra, 2023).

Bayesian neural networks are now widely used in many applications such as fault detection (Niu et al., 2024; Zhou et al., 2022), remaining useful life prediction (Liang et al., 2024; Xiang et al., 2024; Mazaev et al., 2021; Li et al., 2020; Wang et al., 2023), load forecasting (Tziolis et al., 2023), traffic flow prediction (Fu et al., 2024; Xia et al., 2024), 3D human pose estimation (Ramirez et al., 2020) and so on. For instance, Niu et al. (Niu et al., 2024) propose a scoring Bayesian neural network that improves model performance by solving for surface defect segmentation probabilities using Bayesian neural computation to provide expressions for uncertain regions and using the variance of the segmentation probabilities to assess the quality of the labels. Tziolis et al. (Tziolis et al., 2023) proposed a method for predicting short-term net loads at the distribution level using a Bayesian neural network model and optimized the proposed model with decision heuristics in the statistical post-processing stage. Fu et al. (Fu et al., 2024) proposed the Bayesian graph convolutional network framework to better describe the spatial relationships between traffic conditions by considering the graph structure as a stochastic realization of a parametric generative model and inferring the posterior from the observed road network topology and traffic data. Liang et al. (Liang et al., 2024) proposed a hybrid method combining a model-based approach and a data-driven approach for lifetime prediction and uncertainty quantification for a lithium battery dataset using Bayesian neural networks, demonstrating accurate prediction performance with a small sample dataset. Ramirez et al. (Ramirez et al., 2020) proposed Bayesian Capsule Networks as a novel DNN archi-

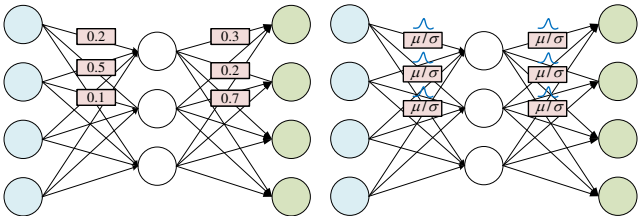


Figure 1: On the left is the structure of a traditional neural network with fixed weights; on the right is the structure of a Bayesian neural network with probability distributions for the weights.

tecture that provides richer representations by representing each concept as many different vectors, addressing the ill-posed problem of estimating 3D human pose from a single 2D image.

However, existing BNN-based approaches have the following limitations. First, while BNNs have been widely used in various domains, their integration with MIL frameworks for student performance prediction remains underexplored. This integration could address the challenges of data sparsity and overfitting in educational data mining. Second, existing BNNs often lack dynamic attention mechanisms that can adaptively weigh the importance of different instances within a bag, which is crucial for extracting meaningful patterns from sparse and noisy data.

Based on the above discussion, the key research gaps in existing approaches are as follows. First of all, existing MIL and BNN models often struggle with overfitting when applied to high-dimensional and noisy educational data, limiting their practical utility. Second, while MIL can handle sparse data, existing methods fail to fully exploit the relationships between instances due to inadequate feature extraction and fusion mechanisms. Third, most existing frameworks do not provide uncertainty estimates for their predictions, which are essential for educational stakeholders to make informed decisions. Finally, the integration of MIL and BNNs for student performance prediction remains underexplored, despite its potential to address the challenges of data sparsity and overfitting. To address these gaps, this paper proposes a novel Bay-DeepMIL framework, which integrates MIL with BNNs to handle data sparsity, mitigate overfitting, and provide uncertainty estimates for predictions. The proposed framework also introduces a Bayesian multi-head attention mechanism to dynamically weigh the importance of different instances, enhancing feature extraction and fusion.

3. Methodology

In this section, a new method is introduced to predict students' performance. First, the student academic performance dataset is briefly described. Then, a prediction framework based on Bay-DeepMIL is described, which consists of three main parts, each of which will be explained in detail.

3.1. Data Preparation

In this study, a dataset of academic performance records from undergraduate Computer Science students was collected from the University's Data Center, covering six academic semesters from September 2018 to July 2021. The dataset includes behavioral engagement logs, course enrollment information, and grade records. Student samples with more than 25% missing data were excluded from analysis. Finally, the student academic performance dataset is denoted as $\mathcal{D} = \{\boldsymbol{X}_i, y_i\}_{i=0}^{S}$, where $S = 1048$ represents 1048 students (721 males, 327 females, aged from 17 to 19) enrolled in 2018. All data were obtained through a formal data-sharing agreement with the University's Data Center. After signing a non-disclosure agreement (NDA), the research team was granted access to anonymized student data solely for academic research purposes. All participants had provided informed consent for their de-identified data to be used in this context. This study was reviewed and approved by the University's Research Ethics Committee to ensure full compliance with data privacy and research ethics standards.

Table 1
Detailed features of the student in the dataset

Feature Type	Feature
Student Performance (7)	Continuous Assessment Grade
	Final Examination Grade
	Composite Score
	Cumulative Average Point (CAP)
	Academic Performance Index
	Intra-class Grade Position
	Course Repetition Count
Student Behavior (3)	Session Absenteeism Count
	Tardiness Frequency
	Attendance Record Count

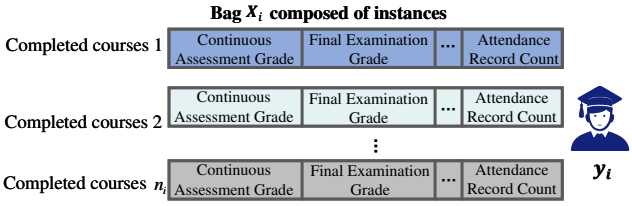


Figure 2: Multi-instance representation for student performance. Each instance represents information about one of the student's training courses.

As shown in Fig. 2, for student i ($i = 1, 2, \cdots, S$), his(or her) data record is denoted as $\{\boldsymbol{X}_i, y_i\}$, where $\boldsymbol{X}_i$ is a "bag" consisting of multiple instances and y_i is the ground truth of the student i to be predicted on a target course j in a future term ($y_i = 1$ indicates the student ranks in the bottom 20% of the target course j, otherwise it is 0). The target courses are distributed across the second to sixth terms, with each term comprising five courses, and each student has a recorded grade for the target course. Each bag includes instances of courses completed before the target course term, and these completed courses vary from student to student, covering a total of 124 different courses. Each bag is defined as

$$\boldsymbol{X}_i = \left\{\boldsymbol{x}_i^1, \boldsymbol{x}_i^2, \cdots, \boldsymbol{x}_i^{n_i}\right\} (i = 1, 2, \cdots, S), \tag{1}$$

where n_i is the number of courses the student i has completed. Instance $\boldsymbol{x}_{i,c} (1 \leq c \leq n_i)$ in $\boldsymbol{X}_i$ is a d-dimensional ($d = 10$) vector representing the features of c-th training course from the student i. The d features mainly contain features of student performance and student behavior. And the details of the d features are described in Table 1.

In addition to the student features, the dataset also comprises 93 courses, each described by 10 features categorized into 3 types: 4 category features, 5 numerical features, and 1 text feature outlined in Table 2. Numerical features are normalized to [0,1] using min-max normalization, while text

Table 2
Detailed features of the courses in the dataset

Feature Type	Feature	Feature Value
Category (4)	Course Type	$\{01, 02, \cdots, 12\}$
	Teaching Mode	$\{01, 02, \cdots, 06\}$
	Course Affiliation	$\{01, 02, \cdots, 08\}$
	Course Examination Types	$\{01, 02, , 03\}$
Numerical (5)	Term	$[1, 6]$
	Credit	$[0.5, 6.0]$
	Credit Hours	$[2, 88]$
	Practical Class Hours	$[0, 40]$
	Lecture Hours	$[0, 78]$
Text (1)	Course Outline	Cleaned Text

features are derived from cleaned course introduction documents, involving the removal of punctuation, stop words, and corpus normalization.

3.2. Student performance prediction framework based on Bay-DeepMIL

In this paper, a Bay-DeepMIL method is proposed for predicting student performance. In this framework, each student's data is represented as a bag $\boldsymbol{X}_i$, which contains multiple instances. Unlike traditional models that require complete or imputed feature vectors, the proposed Bay-DeepMIL model naturally handles the variable-length and sparse structure of student-course data by leveraging the flexibility of the Multi-Instance Learning framework. This formulation eliminates the need for data imputation and allows the model to operate directly on the available course instances per student.

Firstly, $\boldsymbol{X}_i$ is processed by a Bayesian feature extractor to obtain instance features $\boldsymbol{H}_i$. Then, the instance features $\boldsymbol{H}_i$ are fused into a bag-level feature representation $\tilde{\boldsymbol{h}}_i$ by the Bayesian Multi-Head Attention Mechanism (BMHA). $\tilde{\boldsymbol{h}}_i$ is then processed at the classification layer to obtain a classification probability vector $\tilde{\boldsymbol{z}}_i = [\mathcal{G}, 1 - \mathcal{G}]$.

All parameters ω in Bay-DeepMIL, including the weights and biases of the convolutional and fully connected layers as well as the attention mechanism, are regarded as random variables, and their probability distributions parameterized by the variational inference θ are learned during the network training process. The optimization objective of θ is to minimize the KL divergence between the posterior distribution $P(\omega|\mathcal{D})$ and the approximate distribution $q_\theta(\omega)$. In the network update phase, the Monte Carlo sampling method is used to approximate the loss function, evaluate the performance of the network under different parameter values by multiple sampling, and update the variational parameter θ accordingly to optimize the KL divergence.

In addition, for each new input sample $\boldsymbol{X}_{\text{new}}$, a quantitative estimate of the prediction uncertainty is obtained by calculating the standard deviation of the prediction results obtained from multiple sampling.

3.3. Architecture of Bay-DeepMIL

Given a dataset represented as $\mathcal{D}$, for each student i, the Bayesian framework provides a probability estimate of the student failure risk:

$$P(y_i|\boldsymbol{X}_i;\omega) = \mathcal{B}(F_\omega(\boldsymbol{X}_i)), \tag{2}$$

where $F_\omega(\boldsymbol{X}_i)$ is the neural network with parameters ω, mapping the input $\boldsymbol{X}_i$ to a probability score. The $\mathcal{B}$ denotes a Bernoulli distribution, reflecting the binary nature of the classification task, with the network output being the probability of the student's failure risk.

Unlike traditional deterministic deep learning models, where the parameters are presumed to have fixed values, hence producing a single output $\hat{y}_i$ for an input $\boldsymbol{X}_i$, BNNs postulate model parameters ω as stochastic variables governed by a prior distribution $P(\omega)$. Consequently, for an input $\boldsymbol{X}_i$, BNNs can yield a range of potential outcomes $\{\hat{y}_i^1, \hat{y}_i^2, \ldots, \hat{y}_i^S\}$, each representing a plausible prediction. The variability among these outcomes, quantified by their standard deviation, constitutes an estimate of the predictive uncertainty embedded within the model's outputs. This variance reflects the epistemic uncertainty inherent in the model parameters and is a direct result of integrating over the posterior distribution of these parameters:

$$P(\hat{y}_i|\boldsymbol{X}_i, \mathcal{D}) = \int P(\hat{y}_i|F_\omega(\boldsymbol{X}_i))P(\omega|\mathcal{D})\, d\omega, \tag{3}$$

where $P(\hat{y}_i|\boldsymbol{X}_i, \mathcal{D})$ represents the model's predicted probability for the input $\boldsymbol{X}_i$ leading to the output $\hat{y}_i$, and $P(\omega|\mathcal{D})$ is the posterior probability of the model parameters given data $\mathcal{D}$. The function $P(\hat{y}_i|F_\omega(\boldsymbol{X}_i))$ denotes the likelihood of observing $\hat{y}_i$ given the model output for $\boldsymbol{X}_i$.

The optimization of Bayesian neural networks aims at identifying the ideal parameters ω distribution. A primary obstacle in Bayesian deep learning inference is the computational intensity required to calculate the posterior probability $P(\omega|\mathcal{D})$, especially given the potential for models to possess millions of weights. To address this, approximation techniques such as variational inference are employed to estimate posterior distributions. Specifically, the Gaussian distribution is often utilized to model the uncertainty within these networks. The divergence between the precise posterior and its Gaussian approximation, measured by the Kullback-Leibler (KL) divergence, is minimized throughout the training stage.

The posterior distribution $P(\omega|\mathcal{D})$ of the deep learning model is approximated by the variational distribution $q_{\dot{\theta}}(\omega)$, which is parameterized by $\theta \sim \mathcal{N}(\mu, \sigma)$ to strike a balance between computational efficiency and inference accuracy. The variational family $q_\theta(\omega)$ is selected, and the optimal variational parameter $\dot{\theta}$ is determined by minimizing the KL divergence between $P(\omega|\mathcal{D})$ and $q_\theta(\omega)$:

$$KL\left(q_\theta(\omega) \| P(\omega \mid \mathcal{D})\right) = \int q_\theta(\omega) \log \frac{q_\theta(\omega)}{P(\omega \mid \mathcal{D})} d\omega \tag{4}$$

According to Bayes' law, the posterior distribution is ex-

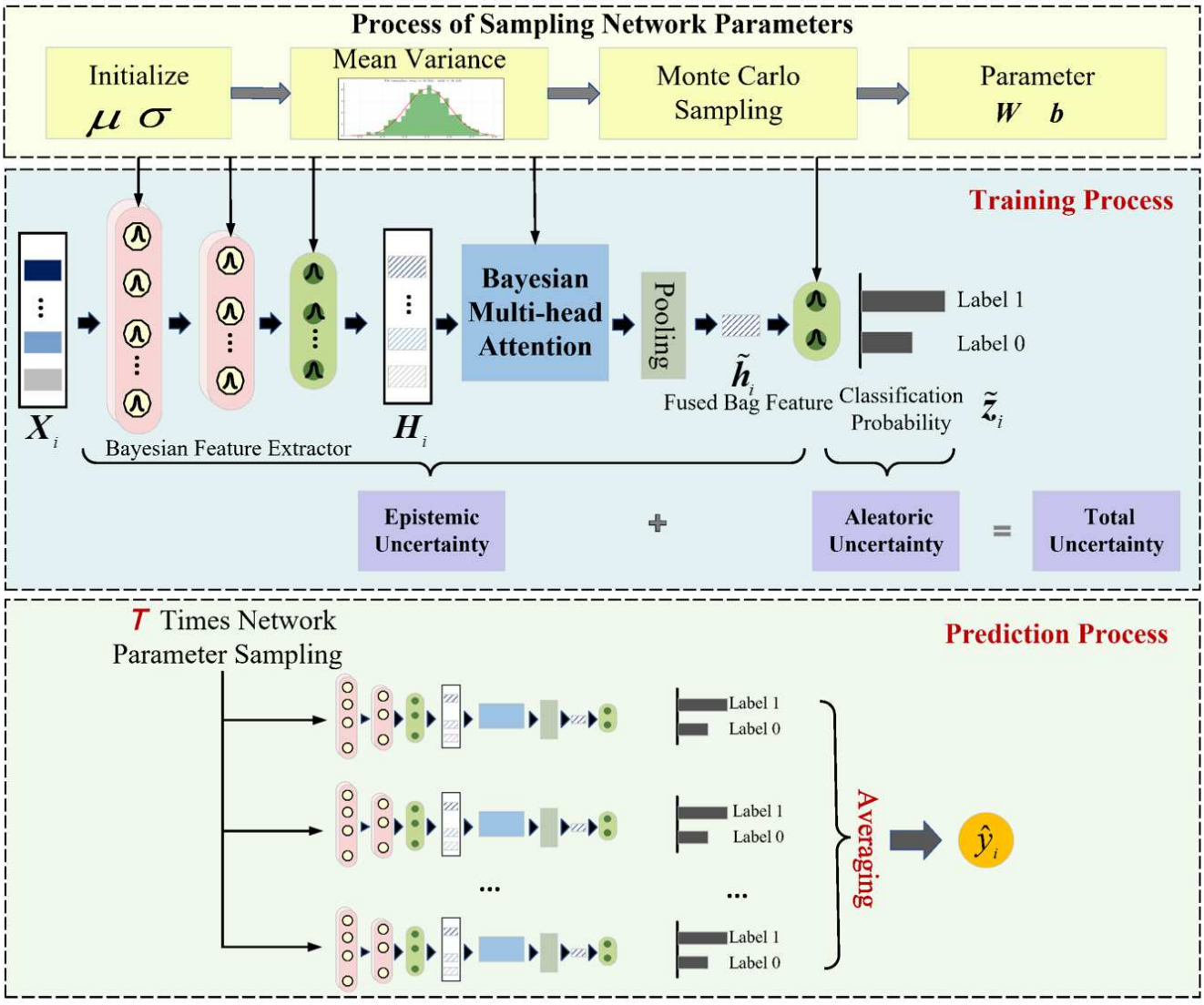


Figure 3: Main prediction framework based on Bay-DeepMIL. During training, parameters are sampled using variational inference. During inference, multiple forward passes are used to estimate both aleatoric and epistemic uncertainty components.

pressed as

$$P(\omega \mid \mathcal{D}) = \frac{P(\mathcal{D} \mid \omega)P(\omega)}{P(\mathcal{D})}. \tag{5}$$

The KL divergence can be expressed as follows:

$$\begin{aligned} KL\left(q_\theta(\omega) \| P(\omega \mid \mathcal{D})\right) &= \int q_\theta(\omega) \log \frac{q_\theta(\omega)P(\mathcal{D})}{P(\mathcal{D} \mid \omega)P(\omega)} d\omega \\ &= \int q_\theta(\omega) \log q_\theta(\omega) d\omega \\ &+ \log P(\mathcal{D}) \int q_\theta(\omega) d\omega \\ &- \int q_\theta(\omega) \log P(\mathcal{D} \mid \omega) d\omega \\ &- \int q_\theta(\omega) \log P(\omega) d\omega \end{aligned} \tag{6}$$

Express the equation(4) in expectation form as follows:

$$\begin{aligned} KL\left(q_\theta(\omega) \| P(\omega \mid \mathcal{D})\right) &= \mathbb{E}_{q_\theta(\omega)}\left[\log q_\theta(\omega)\right] + \log P(\mathcal{D}) \\ &- \mathbb{E}_{q_\theta(\omega)}[\log P(\mathcal{D} \mid \omega)] \\ &- \mathbb{E}_{q_\theta(\omega)}[\log P(\omega)] \end{aligned} \tag{7}$$

It is worth noting that the prediction task is formulated as a binary classification problem, where students in the bottom 20% of each course are labeled as positive cases. This setup introduces an inherent class imbalance with an approximate 1:4 positive-to-negative ratio. Instead of applying conventional techniques such as weighted cross-entropy, the Bayesian variational inference framework inherently accounts for data distribution through the expected log-

Table 3
Bay-DeepMIL architecture

Layer Type	Kernel/Head	Units/Filters	Input Size	Output Size	Activation/Function
1D Convolution	Kernel: 2	Filters: r_1	$n_i \times d \times 1$	$n_i \times (d-1) \times r_1$	ReLU
1D Convolution	Kernel: 2	Filters: r_2	$n_i \times (d-1) \times r_1$	$n_i \times (d-2) \times r_2$	ReLU
Fully Connected	-	Units: r_3	$n_i \times (d-2) \times r_2$	$n_i \times r_3$	ReLU
Multi-head Attention	Heads: ρ	-	$n_i \times r_3$	$n_i \times r_4$	Softmax, Linear
Pooling	-	-	$n_i \times r_4$	$1 \times r_4$	Avg/Max Pooling
Fully Connected (Classifier)	-	Units: 2	$1 \times r_4$	1×2	Softmax

likelihood term $\log P(\mathcal{D}|\omega)$ in the objective function. Since the logarithm of the data probability $\log P(\mathcal{D})$ is a constant and does not influence the optimization, the loss function can be framed as:

$$\begin{aligned}\mathcal{L}(\theta) :=& \arg\min_{\theta} \mathbb{E}_{q_\theta(\omega)}\left[\log q_\theta(\omega)\right] \\ &-\mathbb{E}_{q_\theta(\omega)}[\log P(\mathcal{D} \mid \omega)] \\ &-\mathbb{E}_{q_\theta(\omega)}[\log P(\omega)],\end{aligned} \tag{8}$$

Given the assumption of parameter independence within the model, an estimation of the network's loss function can be conducted through Monte Carlo sampling, as detailed below:

$$\mathcal{L}(\theta) \approx \frac{1}{N}\sum_{n=1}^{N}\left(\log q_\theta\left(\omega^n\right) - \log P\left(\mathcal{D} \mid \omega^n\right) - \log P\left(\omega^n\right)\right) \tag{9}$$

where N is the number of samples and ω^n is the parameter of the n-th sample. This approach not only approximates the loss function but also increases the robustness of the model by capturing the uncertainty in the parameters. By minimizing $\mathcal{L}(\theta)$, the optimal distribution of the model parameters can be learned for prediction.

Each weight $\boldsymbol{W}$ and bias $\boldsymbol{b}$ in Bay-DeepMIL is a distribution with trainable parameters, and its initialization distribution is a standard normal distribution, which is trained to obtain its optimal mean and variance.

For the convolutional layer, the weights and biases are parameterized as follows:

$$\boldsymbol{W}_{\text{conv}} = \mu_{\text{conv}} + \epsilon_{\boldsymbol{W}} \odot \sigma_{\text{conv}}, \quad \epsilon_{\boldsymbol{W}} \sim \mathcal{N}(0, I) \tag{10}$$

$$\boldsymbol{b}_{\text{conv}} = \mu_{\boldsymbol{b}_{\text{conv}}} + \epsilon_{\boldsymbol{b}} \odot \sigma_{\boldsymbol{b}_{\text{conv}}}, \quad \epsilon_{\boldsymbol{b}} \sim \mathcal{N}(0, I) \tag{11}$$

where μ_{conv} and $\mu_{\boldsymbol{b}_{\text{conv}}}$ represent the mean parameters of weights and biases, respectively, and σ_{conv} and $\sigma_{\boldsymbol{b}_{\text{conv}}}$ represent the corresponding standard deviation parameters, respectively. The $\epsilon_{\boldsymbol{W}}$ and $\epsilon_{\boldsymbol{b}}$ are random noises sampled from a standard normal distribution to model the randomness of the weights and biases. The result of the convolution operation is transformed into the input of the next layer using an activation function, calculated as follows:

$$h_{\text{conv}} = ReLU(\boldsymbol{W}_{\text{conv}} * X_i + \boldsymbol{b}_{\text{conv}}) \tag{12}$$

For the fully connected layer, the parameterization of the weights and biases follows the same way as described above, represented by the trainable mean and variance:

$$\boldsymbol{W}_{\text{fc}} = \mu_{\text{fc}} + \epsilon_{\boldsymbol{W}} \odot \sigma_{\text{fc}}, \quad \epsilon_{\boldsymbol{W}} \sim \mathcal{N}(0, I) \tag{13}$$

$$\boldsymbol{b}_{\text{fc}} = \mu_{\boldsymbol{b}_{\text{fc}}} + \epsilon_{\boldsymbol{b}} \odot \sigma_{\boldsymbol{b}_{\text{fc}}}, \quad \epsilon_{\boldsymbol{b}} \sim \mathcal{N}(0, I) \tag{14}$$

The output of the fully connected layer is calculated as follows by combining the activation function with the output of the previous layer:

$$h_{\text{fc}} = ReLU(\boldsymbol{W}_{\text{fc}} \cdot h_{\text{prev}} + \boldsymbol{b}_{\text{fc}}) \tag{15}$$

In the above formulation, h_{prev} represents the output of the previous layer of the network, $\odot$ denotes the Hadamard product (i.e., multiplication between elements), and $*$ denotes the convolution operation.

This formulation allows the network to take into account parameter uncertainty during training and estimate the uncertainty in the model output through subsequent Bayesian inference. The θ (μ and σ) of each parameter is the goal of model optimization.

Bay-DeepMIL uses a Bayesian feature extractor containing a three-layer architecture with two one-dimensional convolutional layers and one fully connected layer. The detailed parameter configuration of each layer is shown in Table 3. The input to the feature extractor is $\boldsymbol{X}_i = \{\boldsymbol{x}_i^1, \boldsymbol{x}_i^2, \cdots, \boldsymbol{x}_i^{n_i}\} \in \mathbb{R}^{n_i \times d \times 1}$. The feature extractor is computed as follows:

$$\boldsymbol{H}_i = f(\boldsymbol{X}_i), \tag{16}$$

where $\boldsymbol{H}_i = \{\boldsymbol{h}_i^1, \boldsymbol{h}_i^2, \cdots, \boldsymbol{h}_i^{n_i}\} \in \mathbb{R}^{n_i \times r_3}$ is the output feature representation of each bag, $f(\cdot)$ denotes the 3-layer Bayesian feature extractor detailed above, and $\boldsymbol{h}_i^c$ represents the feature representation of the c-th instance in the i-th bag.

The feature extractor's output is weighted via the Bayesian multi-head attention mechanism, which are further detailed in Subsection 3.4, to form a fused instance representation $\tilde{\boldsymbol{h}}_i \in \mathbb{R}^{1 \times r_3}$ for each bag.

In the classification layer, as in the fully connected layer in the feature extractor, the weights $\boldsymbol{W}$ and biases $\boldsymbol{b}$ are treated as random variables, and the distributions are parameterized by the mean μ and the standard deviation σ. The final output of the neural network is obtained as:

$$\tilde{\boldsymbol{z}}_i = softmax(\boldsymbol{W} \cdot \tilde{\boldsymbol{h}}_i + \boldsymbol{b}). \tag{17}$$

$\tilde{\boldsymbol{z}}_i = [\mathcal{G}, 1-\mathcal{G}]$ represents the classification probability vector of the i-th bag, where $\mathcal{G}$ is the probability that student i is at risk of failing target course j.

Once training is complete, in the testing phase, the predicted label of student i is gotten by averaging over T Monte Carlo samples of the neural network parameters, calculated as:

$$\hat{y}_i = \begin{cases} 1, & \text{if } \frac{1}{S}\sum_{s=1}^{S} \mathcal{G}^s > 0.5; \\ 0, & \text{otherwise;} \end{cases} \tag{18}$$

where $\mathcal{G}^s$ is the model's output for the t-th Monte Carlo sample and $\hat{y}_i$ is the prediction result given by the model ($\hat{y}_i = 1$ means the student i is judged by the model to have failed. $\hat{y}_i = 0$ means the student i is judged by the model to have not failed).

3.4. Bayesian multi-head attention mechanism

In this subsection, we propose the Bayesian Multi-Head Attention (BMHA) mechanism, which fuses instance features $\boldsymbol{H}_i = \{\boldsymbol{h}_i^1, \boldsymbol{h}_i^2, \cdots, \boldsymbol{h}_i^{n_i}\} \in \mathbb{R}^{n_i \times r_3}$ into bag features by assigning trainable probability distributions to the parameters of the attention heads, as opposed to fixed values.

For ρ-th head, calculated as follows:

$$Attention(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = softmax\left(\frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{d_\rho}}\right)\boldsymbol{V}, \tag{19}$$

where $\boldsymbol{Q} = \boldsymbol{H}_i \boldsymbol{W}_\rho^Q \in \mathbb{R}^{n_i \times d_\rho}$ represents the query matrix, $\boldsymbol{K} = \boldsymbol{H}_i \boldsymbol{W}_\rho^K \in \mathbb{R}^{n_i \times d_\rho}$ represents the key matrix, $\boldsymbol{V} = \boldsymbol{H}_i \boldsymbol{W}_\rho^V \in \mathbb{R}^{n_i \times d_\rho}$ represents the value matrix, d_ρ denotes the dimensionality of the key vectors, ensuring that the scaling factor $\sqrt{d_\rho}$ normalizes the dot products to avoid excessively large values during softmax. $\boldsymbol{W}_\rho^Q \in \mathbb{R}^{r_3 \times d_\rho}$, $\boldsymbol{W}_\rho^K \in \mathbb{R}^{r_3 \times d_\rho}$ and $\boldsymbol{W}_\rho^V \in \mathbb{R}^{r_3 \times d_\rho}$ are calculated as follows:

$$\boldsymbol{W}_\rho^Q = \mu_Q^\rho + \epsilon_Q \odot \sigma_Q^\rho, \quad \epsilon_Q \sim \mathcal{N}(0, I) \tag{20}$$

$$\boldsymbol{W}_\rho^K = \mu_K^\rho + \epsilon_K \odot \sigma_K^\rho, \quad \epsilon_K \sim \mathcal{N}(0, I) \tag{21}$$

$$\boldsymbol{W}_\rho^V = \mu_V^\rho + \epsilon_V \odot \sigma_V^\rho, \quad \epsilon_V \sim \mathcal{N}(0, I) \tag{22}$$

Following the computation of attention for each head, the outputs are concatenated and linearly transformed to produce the final bag-level feature representation. Specifically, the output of the Bayesian Multi-Head Attention mechanism is obtained by concatenating the outputs of all ρ heads and then applying a parameterized linear transformation:

$$f_{\text{BMHA}}(\boldsymbol{H}_i) = \text{Concat}(head_1, head_2, \ldots, head_\rho)\boldsymbol{W}^O, \tag{23}$$

where $\text{Concat}(head_1, head_2, \ldots, head_\rho)$ denotes the concatenation of the output vectors from all ρ attention heads. $\boldsymbol{W}^O \in \mathbb{R}^{(\rho \cdot d_\rho) \times r_4}$ is the weight matrix for the final linear transformation applied to the concatenated attention head outputs, computed as follows:

$$\boldsymbol{W}^O = \mu_O + \epsilon_O \odot \sigma_O, \quad \epsilon_O \sim \mathcal{N}(0, I) \tag{24}$$

After attention-based feature fusion, a pooling layer aggregates the fused features, reducing the instance-level representations into a singular, comprehensive bag-level representation:

$$\tilde{\boldsymbol{h}}_i = \text{Pooling}(f_{\text{BMHA}}(\boldsymbol{H}_i)). \tag{25}$$

where $\text{Pooling}(\cdot)$ refers to a pooling operation that aggregates information across the different instance features to form a unified representation.

3.5. Uncertainty estimation

For a new input sample $\boldsymbol{X}_{\text{new}}$ that is forward propagated as in subsection E, the classification probability is obtained as $\tilde{\boldsymbol{z}}_{\text{new}} = [\mathcal{G}_{\text{new}}, 1-\mathcal{G}_{\text{new}}]$, where $\mathcal{G}_{\text{new}}$ is the probability that student is at risk of failing target course. The expected value of $\mathcal{G}_{\text{new}}$ is as follows:

$$\mathbb{E}\left[\mathcal{G}_{\text{new}}\right] = \int \mathcal{G}_{\text{new}}\, P\left(\mathcal{G}_{\text{new}} \mid \boldsymbol{X}_{\text{new}}, \mathcal{D}\right) d\mathcal{G}_{\text{new}} \tag{26}$$

where $P\left(\mathcal{G}_{\text{new}} \mid \boldsymbol{X}_{\text{new}}, \mathcal{D}\right)$ is unfolded according to full probability as follows:

$$P\left(\mathcal{G}_{\text{new}} \mid \boldsymbol{X}_{\text{new}}, \mathcal{D}\right) = \int P\left(\mathcal{G}_{\text{new}} \mid \boldsymbol{X}_{\text{new}}, \omega\right) P(\omega \mid \mathcal{D}) d\omega \tag{27}$$

where $P\left(\mathcal{G}_{\text{new}} \mid \boldsymbol{X}_{\text{new}}, \omega\right)$ is the probability of outputting $\mathcal{G}_{\text{new}}$ given the particular parameter ω, and $P(\omega|\mathcal{D})$ is the posterior probability distribution for the parameter ω given the data $\mathcal{D}$. Combining Eq. 26 and Eq. 27, it can be obtained:

$$\begin{aligned} &\mathbb{E}_{P(\mathcal{G}_{\text{new}} \mid \boldsymbol{X}_{\text{new}}, \mathcal{D})}\left[\mathcal{G}_{\text{new}}\right] \\ &= \int \left(\int \mathcal{G}_{\text{new}}\, P\left(\mathcal{G}_{\text{new}} \mid \boldsymbol{X}_{\text{new}}, \omega\right) d\mathcal{G}_{\text{new}} \right) P(\omega \mid \mathcal{D}) d\omega \\ &= \int \mathbb{E}_{P(\mathcal{G}_{\text{new}} \mid \boldsymbol{X}_{\text{new}}, \omega)}\left[\mathcal{G}_{\text{new}}\right] P(\omega \mid \mathcal{D}) d\omega \end{aligned}$$

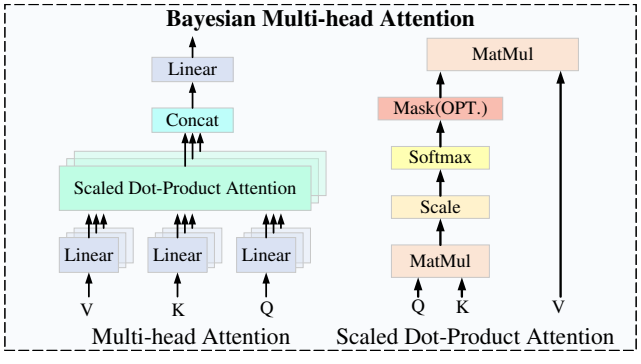


Figure 4: Structure of Bayesian multi-head attention model.

(28)

Similarly, the expectation of $\mathcal{G}_{\text{new}}^2$ is calculated as follows:

$$\begin{aligned}&\mathbb{E}_{P(\mathcal{G}_{\text{new}}|\boldsymbol{X}_{\text{new}},\mathcal{D})}\left[\mathcal{G}_{\text{new}}^2\right]\\&=\int \mathbb{E}_{P(\mathcal{G}_{\text{new}}|\boldsymbol{X}_{\text{new}},\omega)}\left[\mathcal{G}_{\text{new}}^2\right]P(\omega\mid\mathcal{D})d\omega\end{aligned}\tag{29}$$

The variance of $\mathcal{G}_{\text{new}}$ is then calculated as follows:

$$\begin{aligned}&\text{Var}_{P(\mathcal{G}_{\text{new}}|\boldsymbol{X}_{\text{new}},\mathcal{D})}\left(\mathcal{G}_{\text{new}}\right)\\&=\mathbb{E}_{P(\mathcal{G}_{\text{new}}|\boldsymbol{X}_{\text{new}},\mathcal{D})}\left(\mathcal{G}_{\text{new}}^2\right)-\left[\mathbb{E}_{P(\mathcal{G}_{\text{new}}|\boldsymbol{X}_{\text{new}},\mathcal{D})}\left(\mathcal{G}_{\text{new}}\right)\right]^2\\&=\int \text{Var}_{P(\mathcal{G}_{\text{new}}|\boldsymbol{X}_{\text{new}},\omega)}\left(\mathcal{G}_{\text{new}}\right)P(\omega\mid\mathcal{D})d\omega\\&+\int\left[\mathbb{E}_{P(\mathcal{G}_{\text{new}}|\boldsymbol{X}_{\text{new}},\omega)}\left(\mathcal{G}_{\text{new}}\right)\right]^2P(\omega\mid\mathcal{D})d\omega\\&-\left[\mathbb{E}_{P(\mathcal{G}_{\text{new}}|\boldsymbol{X}_{\text{new}},\mathcal{D})}\left(\mathcal{G}_{\text{new}}\right)\right]^2\\&=\int \text{Var}_{P(\mathcal{G}_{\text{new}}|\boldsymbol{X}_{\text{new}},\omega)}\left(\mathcal{G}_{\text{new}}\right)P(\omega\mid\mathcal{D})d\omega\\&+\int \mathbb{E}_{P(\mathcal{G}_{\text{new}}|\boldsymbol{X}_{\text{new}},\omega)}\left(\mathcal{G}_{\text{new}}\right)\\&-\mathbb{E}_{P(\mathcal{G}_{\text{new}}|\boldsymbol{X}_{\text{new}},\mathcal{D})}\left(\mathcal{G}_{\text{new}}\right)^2P(\omega\mid\mathcal{D})d\omega\end{aligned}\tag{30}$$

Using the optimal distribution parameters θ^* obtained from the training process, the output of the model is sampled T times using Monte Carlo, each time taking values from the optimal variational distribution of weights and biases and performing one forward propagation. The estimator of Eq. 30 is as follows:

$$\hat{v}^2=\frac{1}{S}\sum_{s=1}^{S}\left[\mathcal{G}_{\text{new},s}-\overline{\mathcal{G}}_{\text{new}}\right]^2+\left[\frac{1}{S}\sum_{s=1}^{S}\mathcal{G}_{\text{new},s}-\overline{\mathcal{G}}_{\text{new}}\right]^2\tag{31}$$

The first of these is aleatoric uncertainty, which reflects the inherent randomness of the data set and is an inherent property of the data set that cannot be eliminated by any means. The second term is epistemic uncertainty, which reflects the prediction uncertainty caused by insufficient data or information limitations. As the data size increases and the model understands the data better, the cognitive uncertainty will gradually decrease.

3.6. Hyper-parameter settings

This study shows the network structure in Fig. 3 and Table 3. For the two 1D CNN layers, the numbers of filters are $r_1 = 20$ and $r_2 = 50$ respectively. For the network layer FC of the feature extractor, $r_3 = 100$. In BMHA, the number of attention heads $\rho = 2$ and $r_4 = 100$. In the model training phase, the batch size is set as 1. The model is trained by stochastic gradient descent (SGD). The learning rate is set as 0.001. Furthermore, we employ a 5-fold stratified cross-validation approach on our dataset comprising 1048 samples, with 210 positives and 838 negatives, using an 80-20 train-test split. Each fold of the training set will undergo 10 random runs to ensure model robustness, where model parameters are reinitialized and data shuffled for each run, maintaining the stratified ratio of classes. The training set's performance will be evaluated on a validation subset within each fold, while the final model's effectiveness will be tested on the separate 20% test subset.

3.7. Evaluation metrics

For the prediction models trained in each target course, this paper utilizes the following four metrics to measure performance: accuracy, precision, recall, and composite metric measure F_γ. These metrics are defined as:

$$\text{accuracy} = \frac{TP+TN}{TP+TN+FP+FN},\tag{32}$$

$$\text{precision} = \frac{TP}{TP+FP},\tag{33}$$

$$\text{recall} = \frac{TP}{TP+FN},\tag{34}$$

$$F_\gamma = \left(1+\gamma^2\right)\cdot\frac{\text{precision}\ \cdot\ \text{recall}}{\gamma^2\cdot\ \text{precision}\ +\ \text{recall}}.\tag{35}$$

where TP, TN, FP, and FN indicate the number of true positives, true negatives, false positives, and false negatives. γ is set to 1.

4. Experiment Results With Discussions

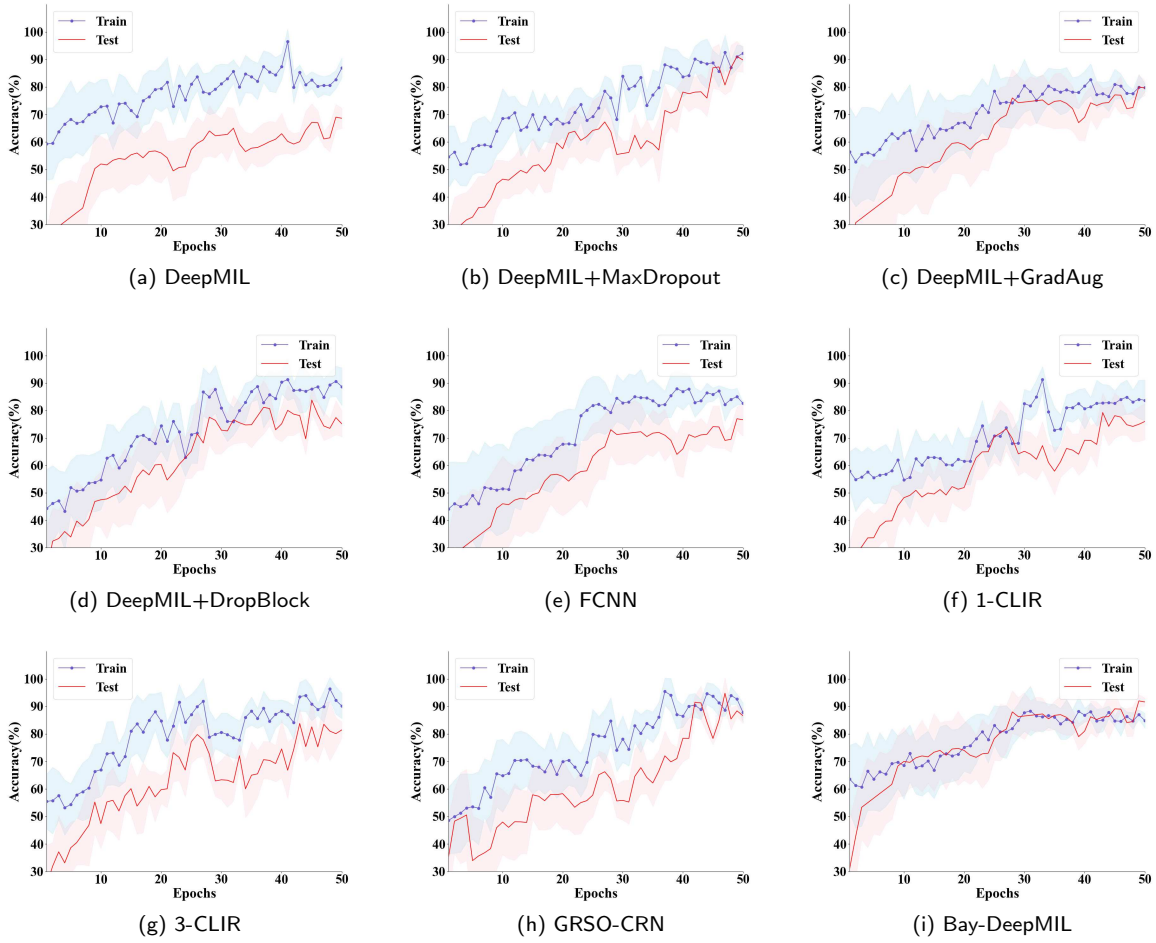
In this section, we first validate the superiority of the proposed method to predict student performance on a self-collected student academic dataset. Secondly, we verify the ability of the proposed method to mitigate model overfitting relative to other methods. Finally, the effectiveness of the proposed pooling method is verified relative to existing pooling methods.

4.1. The Overall Prediction Performance

To evaluate the effectiveness of the proposed Bay-DeepMIL framework in student performance prediction, we compare it with several state-of-the-art deep learning models. The detailed comparison results are summarized in Table 4. Specifically, we include the following representative methods: FCNN (Baranyi et al., 2020), a feedforward neural network designed for interpretable educational prediction; 1-CLIR and 3-CLIR (Yang et al., 2020b), which integrate course-level influence representation using 1-hop and 3-hop relational structures; GRSO-CRN (Nayani & P, 2023), a recently proposed gated recurrent neural network incorporating sequential and relational features; and DeepMIL, a simplified variant of our model using deterministic weights (point estimates) instead of Bayesian distributions. All methods are evaluated on the same dataset (covering semesters 2 to 6) using identical experimental settings. As shown in Table 4, the proposed Bay-DeepMIL

Table 4
Comparison of Classification Result for Different Models (MEAN±STD UNIT:%)

Methods	Accuracy	Precision	Recall	F_1score
FCNN	83.7±13.6	74.5±16.1	82.7±9.6	76.6±7.4
1-CLIR	86.1±11.3	78.3±11.2	83.7±9.1	81.6±7.4
3-CLIR	87.2±7.4	79.9±10.3	85.1±8.8	82.9±10.4
GRSO-CRN	89.3±5.1	82.2±8.4	87.6±7.3	83.6±9.4
DeepMIL	88.2±8.3	79.4±12.4	84.7±6.9	82.4±12.6
Bay-DeepMIL	**94.2±5.3**	**84.4±7.7**	**90.8±4.9**	**86.6±7.2**



(a) DeepMIL (b) DeepMIL+MaxDropout (c) DeepMIL+GradAug

(d) DeepMIL+DropBlock (e) FCNN (f) 1-CLIR

(g) 3-CLIR (h) GRSO-CRN (i) Bay-DeepMIL

Figure 5: Comparison of overfitting of different models on a specific course.

achieves the best performance across all metrics, with an accuracy of 94.2 ± 5.3%, precision of 84.4 ± 7.7%, recall of 90.8 ± 4.9%, and an F1-score of 86.6 ± 7.1%. These results demonstrate that Bay-DeepMIL outperforms existing state-of-the-art methods in student performance prediction, particularly showing significant improvements in recall and overall prediction reliability.

4.2. Evaluation of Overfitting Mitigation

To validate the effectiveness of the proposed Bay-DeepMIL framework in mitigating overfitting through Bayesian inference, we conducted comparative experiments on a representative target course, "Microcomputer Principles and System Design." We compared Bay-DeepMIL with baseline models and DeepMIL variants enhanced by traditional regularization methods, including MaxDropout (do Santos et al., 2021), GradAug (Yang et al., 2020a), and DropBlock (Ghiasi et al., 2018). All models were trained for 50 epochs, and we tracked training and test accuracy throughout. As shown in Figure 5, traditional models without regularization (e.g., DeepMIL, FCNN, 1-CLIR, etc.) show a substantial gap between training and test ac-

Table 5
Comparison of overfitting of different models on all courses(MEAN±STD UNIT:%)

Methods	Training Accuracy	Test Accuracy	Difference
DeepMIL	94.1±2.2	88.2±8.3	5.9
DeepMIL+MaxDropout	91.9±4.8	90.1±4.1	1.8
DeepMIL+GradAug	91.3±5.9	89.0±6.7	2.3
DeepMIL+DropBlock	91.2±2.5	89.2±4.5	2.0
FCNN	93.9±4.3	83.7±13.6	10.2
1-CLIR	90.9±6.1	86.1±11.3	4.8
3-CLIR	92.4±3.3	87.2±7.4	5.2
GRSO-CRN	95.1±3.6	89.3±5.1	5.8
Bay-DeepMIL	**94.9±3.1**	**94.2±5.3**	**0.7**

curacy, indicating overfitting. While the inclusion of traditional regularization strategies helps reduce this gap, Bay-DeepMIL consistently achieves the smallest discrepancy between training and test accuracy. Bay-DeepMIL leverages the Bayesian framework to consider the uncertainty in weights instead of mere point estimation, providing not only quantification of uncertainty but also enhancing the model's robustness to varying data distributions.

To eliminate the influence of data specific to any particular course, experiments were carried out across all courses from the second to the sixth terms, and average predictive performance metrics were calculated. These methods were assessed based on average training accuracy, validation accuracy, and test accuracy across all target courses. As shown in Table 5, Bay-DeepMIL demonstrated the best capacity to mitigate model overfitting, thereby affirming the effectiveness of the approach introduced in this paper. These results validate that the Bayesian modeling approach adopted in Bay-DeepMIL offers improved generalization performance compared to deterministic deep models with traditional regularization techniques, demonstrating its effectiveness in addressing overfitting and uncertainty under sparse and irregular data conditions.

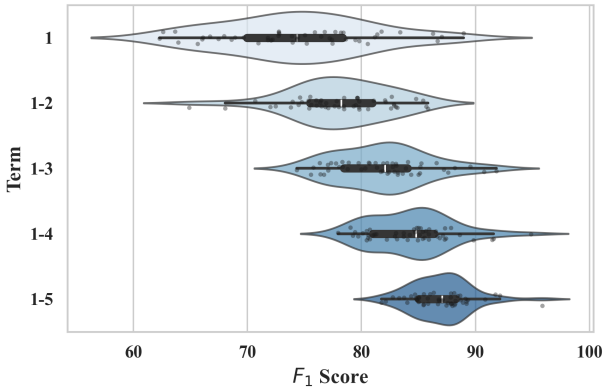


Figure 6: Distribution of F_1 score across different terms using Bay-DeepMIL. The violin plots illustrate the model's performance for each term, reflecting the central tendency and distribution spread. In each boxplot, the top and bottom whiskers indicate the range excluding outliers, the box bounds represent the interquartile range, and the line inside the box denotes the median. The surrounding scatter points show the raw F_1 score data for each course.

4.3. The influence of different parameters

4.3.1. The Influence of the Number of Training Courses

At the end of each term, students complete a certain number of selected courses. In the context of Multi-Instance Learning (MIL), this leads to an increase in the number of instances within each bag. Since the richness of instance information may affect the model's prediction performance, a series of experiments were conducted to explore this influence. Specifically, all sixth-term courses were designated as the target courses, and the task was to predict whether students were at risk of failing them. The F_1 score was used as the evaluation metric and averaged across all target courses. We used course instances from different combinations of early terms—namely Term 1, Term 1–2, ..., up to Term 1–5—to train the model. The resulting performance is shown in Fig. 6. As the number of training terms increases, the corresponding number of course instances per student also increases. The figure demonstrates a general upward trend in F_1 score across terms, indicating that incorporating more historical course data can enhance predictive accuracy.

4.3.2. The influence of different pooling methods on model performance

To verify the effectiveness of the pooling method based on the Bayesian multi-head attention mechanism proposed in this paper, it is compared with two traditional pooling methods in multi-instance learning, and the results of the quadratic test are shown in Fig. 7. Specifically, the instance features output from the Bayesian feature extractor are fused into a bag feature using Average pooling, Maximum pooling, BMHA+Average pooling, and BMHA+Maximum pooling methods, respectively, and the proposed framework in this paper is used to predict students' performance in their term 2 to term 6 courses, respectively, and to compute the average accuracy to evaluate the performance of the model. The experimental results show that BMHA effectively improves

the model performance, and both the BMHA+Average pooling and BMHA+Maximum methods outperform the two traditional pooling methods, Average pooling and Maximum pooling, in predicting the tasks for each term course. Furthermore, the interquartile ranges and outliers present in the boxplots articulate the distribution and variability of the model accuracies. The consistency in performance across different terms suggests that the BMHA mechanism is not only effective but also stable under varying educational contexts.

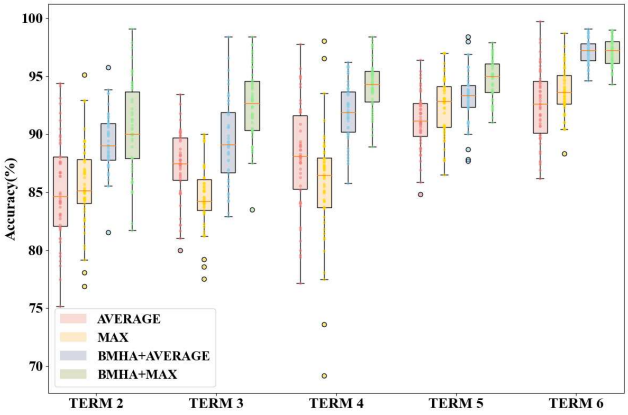


Figure 7: Comparison of accuracy across terms using different pooling methods. This is a box-and-line plot, with the top and bottom of the box corresponding to the upper and lower quartiles of the data, and the horizontal line within the box representing the median, or center of the data. The vertical lines or "whiskers" extending from the box identify the overall spread of the data, while points beyond the endpoints of the whiskers indicate outliers.

4.3.3. The influence of dataset size on model performance

To validate the robustness of the proposed Bay-DeepMIL on small datasets, this study trained various methods, including DeepMIL, DeepMIL+MaxDropout, DeepMIL+GradAug, DeepMIL+DropBlock, FCNN, 1-CLIR, 3-CLIR, GRSO-CRN, and Bay-DeepMIL, with different proportions of training data (75%, 50%, 25%). All courses were treated as target courses for evaluating the models' performance using the average accuracy. The experimental results, as depicted in Fig. 8, indicate that Bay-DeepMIL maintains relatively high accuracy even with substantial reductions in training data. Compared to other tested methods, Bay-DeepMIL demonstrates superior generalization from limited data. Notably, at only 50% of the training data, Bay-DeepMIL's accuracy was significantly higher than that of the other methods, underscoring its robustness on small datasets. Furthermore, the methods DeepMIL+MaxDropout, DeepMIL+GradAug, and DeepMIL+DropBlock, which introduce randomness during model training by adding regularization techniques, avoided overfitting to the training data. When training data was scarce, these models with regularization mechanisms showed a smaller performance decline compared to models without such mechanisms.

4.3.4. The influence of pruning on model performance

Weight pruning is a strategy for optimizing neural networks, aimed at identifying and removing weights that have minimal impact on the model's output to reduce model complexity and the risk of overfitting. In traditional point-estimate-based neural networks, the importance of a weight is judged by the magnitude of its absolute value; smaller weights are typically considered to have less impact on the model and are thus candidates for pruning. In the Bay-DeepMIL model, weight pruning employs a method based on weight uncertainty. As each weight in Bay-DeepMIL is regarded as a probability distribution, the uncertainty of a weight is deemed an important indicator of redundancy. Blundell et al. (Blundell et al., 2015) assessed this uncertainty by calculating the signal-to-noise ratio (SNR) for each weight. Weights with lower SNRs are considered to have higher uncertainty and are more likely to be redundant, allowing for pruning by setting these low SNR weights to 0.

To verify the robustness of Bay-DeepMIL post-pruning, this study employed Blundell et al.'s method to prune the Bay-DeepMIL model. In comparison, traditional pruning based on weight magnitude was applied to the DeepMIL model, removing the smallest absolute weights. Experiments were conducted across all courses from the second to the sixth semester, and average predictive performance metrics were calculated. As illustrated in Fig. 9, the results demonstrate that under high pruning rates (85%, 65%), the performance of the Bay-DeepMIL model was superior to that of DeepMIL. When the pruning rate was reduced to 25%, the performance of Bay-DeepMIL was nearly unaffected. This indicates that the Bay-DeepMIL model can maintain good predictive accuracy even when a large num-

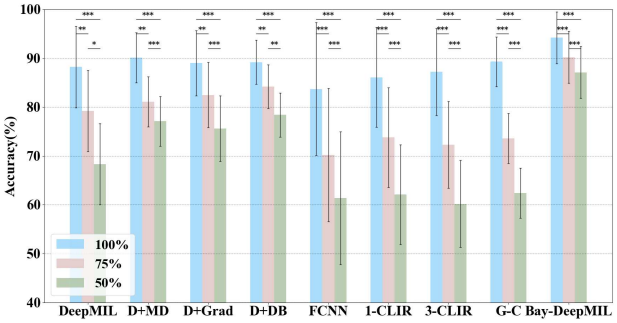


Figure 8: Comparison of accuracy using different methods. The abbreviations "D+MD", "D+Grad", "D+DB", "G-C" stand for DeepMIL+MaxDropout, DeepMIL+GradAug, DeepMIL+DropBlock, and GRSO-CRN, respectively. Asterisks are used to mark significant differences from paired t-tests, where a single asterisk represents a p-value less than 0.05 (* $p < 0.05$), a double asterisk indicates a p-value less than 0.01 (** $p < 0.01$), and a triple asterisk indicates a p-value less than 0.001 (*** $p < 0.001$).

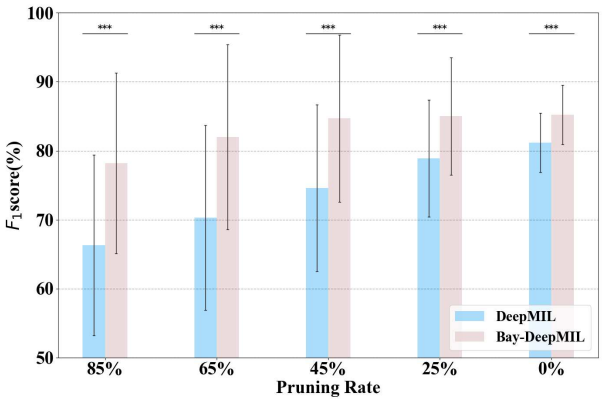


Figure 9: Comparison of the accuracy of the two methods using different pruning rates. Asterisks are used to mark significant differences from paired t-tests, where a triple asterisk indicates a p-value less than 0.001 (*** p <0.001).

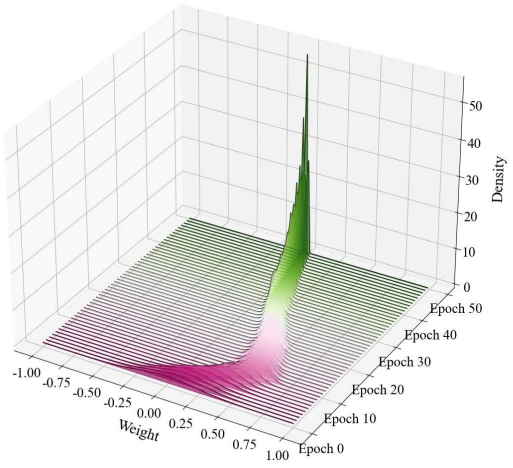


Figure 10: Evolution of the probability distribution for a model parameter during training. The standard deviation of the posterior decreases with training epochs, indicating reduced epistemic uncertainty.

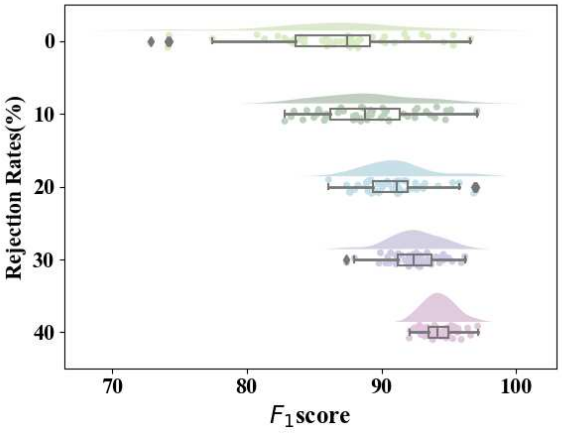


Figure 11: F_1 score across different rejection rates based on epistemic uncertainty. A raincloud plot combines the distribution, scatter points, and a box plot. Higher rejection rates correspond to increased model reliability on retained predictions.

ber of weights are pruned. According to the analysis in this paper, Bay-DeepMIL represents weights through probability distributions during training, thereby better distinguishing and retaining weights crucial for model performance during pruning. Traditional pruning methods based on weight magnitude exhibited greater performance fluctuations, suggesting that such methods are overly simplistic and cannot accurately identify and retain key weights. The uncertainty-based pruning method can effectively remove redundant weights while maintaining model performance, proving the significance of weight uncertainty information in network pruning. Conversely, traditional pruning methods based on absolute weight values sacrifice network performance during the pruning process. By employing the uncertainty-based pruning method, Bay-DeepMIL can reduce computational costs during inference without compromising accuracy, leading to more efficient network operation.

4.3.5. The Influence of Uncertainty on Model Performance

The Bay-DeepMIL framework adopts a Bayesian formulation that models network parameters as probabilistic distributions, enabling the estimation of both aleatoric and epistemic uncertainty. Each parameter is represented as a Gaussian distribution with trainable mean and standard deviation, allowing the model to express predictive uncertainty in a principled manner. Figure 10 illustrates the change in the posterior distribution of a model parameter during training. As the number of training epochs increases, the variance of the distribution narrows, indicating reduced epistemic uncertainty and increased model confidence. This evolution suggests that the model becomes more certain about its predictions as it learns from data. To further demonstrate the practical value of uncertainty estimation, we evaluated the model's performance under different uncertainty-based rejection rates. Specifically, for each test sample, the epistemic uncertainty (approximated by the variance of the Monte Carlo predictions) was computed. Samples with the highest uncertainty were iteratively rejected, and the performance metrics were recalculated on the remaining subset. As shown in Fig. 11, the F_1 score of Bay-DeepMIL improves as the rejection rate increases. This trend highlights that highly uncertain samples are more likely to lead to incorrect predictions. Thus, uncertainty-aware rejection allows the model to avoid unreliable decisions, improving overall predictive quality. This feature is particularly valuable in high-stakes educational settings, where confidence-aware prediction can inform cautious and responsible interventions.

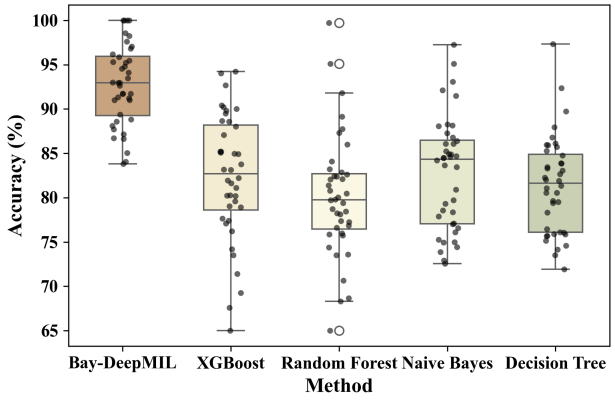


Figure 12: Accuracy comparison with different machine learning methods. The top and bottom horizontal lines represent the overall maximum and minimum values; the box edges indicate the interquartile range, and the central line represents the median accuracy. Black dots denote raw data points from each cross-validation run, and circles indicate outliers.

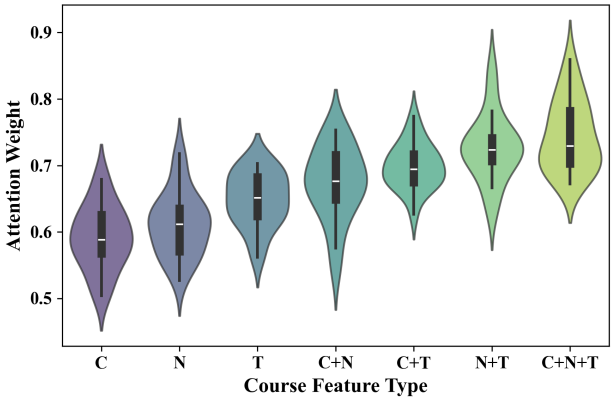


Figure 13: Distribution of attention weights across different course feature types. Courses with more comprehensive features (C+N+T) tend to receive higher attention scores, indicating a stronger influence on student performance prediction.

4.4. Discussions

4.4.1. Comparisons with Traditional Machine Learning Models

To further evaluate the effectiveness of the proposed Bay-DeepMIL framework, we conducted comparative experiments against several widely used traditional machine learning (ML) models. While conventional ML approaches such as Random Forest or XGBoost have been successfully applied in educational prediction tasks, they often rely on flattened feature representations and may fail to capture complex instance-level dependencies within sparse and heterogeneous student data. Specifically, we implemented four classical ML models: eXtreme Gradient Boosting (XGBoost) (Chen & Guestrin, 2016), Random Forest (Biau & Scornet, 2016), Naive Bayes (Yang, 2018), and Decision Tree (Charbuty & Abdulazeez, 2021). All models were trained and evaluated on the same dataset used for Bay-DeepMIL, following a consistent 10-run 5-fold stratified cross-validation protocol. Each model was applied to predict student performance across all target courses from the second to the sixth semesters. The average classification accuracy was reported as the main evaluation metric. As shown in Fig. 12, Bay-DeepMIL consistently achieves higher prediction accuracy than traditional models. This superiority can be attributed to its deep architecture that leverages multi-instance representations and hierarchical attention mechanisms, enabling it to capture intricate relationships among course instances and student behaviors. Moreover, the Bayesian framework enhances robustness under sparse data conditions, distinguishing Bay-DeepMIL from conventional point-estimate models.

4.4.2. Interpretability of the Bayesian Attention Mechanism

In addition to improving predictive performance, the proposed Bayesian Multi-Head Attention (BMHA) mechanism enhances the interpretability of the Bay-DeepMIL model by explicitly modeling the relative importance of different course instances. Specifically, BMHA assigns attention weights to each course instance in a student's bag, reflecting its influence on the prediction of academic risk. To explore how the model allocates importance across various types of course information, we visualize the distribution of attention weights for different course feature types in Fig. 13. As shown, course instances that combine categorical (C), numerical (N), and textual (T) features receive higher average attention weights, indicating their greater contribution to the final decision. This trend confirms that the model effectively leverages richer information to make more informed predictions. Furthermore, the Bayesian formulation introduces a probabilistic treatment of attention parameters, allowing the model to quantify uncertainty in the attention weights. This uncertainty-aware mechanism enables the model to indicate not only which course features are important, but also how confident it is in those assessments. Such capability is particularly valuable in educational settings, where interventions may benefit from both prediction and risk-awareness.

4.4.3. Ablation Study of Each Key Component

To evaluate the contribution of each key component in the Bay-DeepMIL architecture, we conducted an ablation study by systematically removing the following modules: the Bayesian variational inference layer, the multi-head attention mechanism, and the multi-instance learning (MIL) structure. The resulting ablated variants are defined as follows: **DeepMIL-MHA**: A deterministic variant of Bay-DeepMIL in which the Bayesian layer is replaced with standard point-estimate weights; **DeepMIL-Bayesian**: A model using single-head attention instead of multi-head attention; **Flat-Bayesian-MHA**: A variant that replaces the MIL structure with flat averaging across course-level instances. Each variant was trained and evaluated using the same 10-run 5-fold stratified cross-validation protocol as the full model.

Table 6
Ablation study results showing the impact of removing key components from Bay-DeepMIL (mean accuracy ± standard deviation).

Model Variant	Accuracy (%)
Bay-DeepMIL (full model)	**94.2 ± 5.3**
DeepMIL-MHA (no Bayesian)	88.2 ± 8.3
DeepMIL-Bayesian (no MHA)	90.1 ± 5.7
Flat-Bayesian-MHA (no MIL)	85.6 ± 9.4

The performance results, reported in Table 6, show that removing any of the three components leads to a measurable drop in accuracy, confirming that each component contributes meaningfully to the overall predictive performance. Among them, the Bayesian inference module contributes most significantly, supporting both generalization through regularization and uncertainty-aware estimation.

4.4.4. Feasibility and Computational Considerations

To evaluate the real-world feasibility of the proposed Bay-DeepMIL framework, we report the computational resources and time requirements involved during model training and inference. All experiments were conducted on a workstation equipped with an Intel Xeon Silver 4214R CPU, an NVIDIA GeForce RTX 3090 GPU (24 GB VRAM), and 256 GB RAM. The average training time for a single 5-fold cross-validation run (5 epochs per fold) is approximately 42 minutes per target course. The complete 10-run evaluation across all target courses requires around 17.5 hours. During training, the peak GPU memory usage is approximately 8.1 GB, and the average CPU memory consumption remains below 30 GB. While the Bayesian variational inference and Monte Carlo sampling introduce additional training overhead compared to standard deep learning models, the process is fully parallelizable across folds and target courses. This makes the approach scalable in batch-mode settings. Importantly, once the Bay-DeepMIL model is trained, the inference phase is highly efficient. A single prediction with uncertainty estimation via Monte Carlo sampling takes less than 100 milliseconds, making the method well-suited for real-time or near-real-time educational applications. This efficiency ensures that the model can be readily integrated into student monitoring platforms without latency concerns. Overall, these results demonstrate that Bay-DeepMIL remains practical for deployment in institutional environments with modern but not necessarily high-end computational resources, and supports both robust training and fast prediction in real-world educational scenarios.

4.4.5. Limitations and Future Works

While the proposed Bay-DeepMIL framework provides an effective and interpretable solution for predicting students' academic performance, several limitations remain and offer important opportunities for future enhancement. First, the present study is limited to a single-institution dataset consisting of Computer Science students. Although the dataset spans three academic years and includes diverse student demographics, the generalizability of the proposed model to other academic disciplines or institutional contexts remains to be verified. Future work will focus on conducting cross-institutional evaluations via privacy-preserving federated learning frameworks, which can mitigate institutional bias while respecting data privacy constraints. Second, due to its probabilistic formulation, the Bayesian implementation introduces additional computational overhead compared to deterministic deep learning models. Although inference is efficient after training, future studies will investigate computationally efficient Bayesian inference techniques, such as low-rank variational approximations or amortized uncertainty estimation, to reduce training costs without compromising uncertainty quantification. Third, despite efforts to incorporate fairness-aware training, the model's effectiveness remains contingent on the availability of structured campus data. This assumption may not hold in all educational environments. As a future direction, we plan to explore alternative data modalities (e.g., unstructured behavioral logs, textual feedback) and integrate human-in-the-loop mechanisms to support adaptive and context-aware deployment in diverse real-world settings. Together, these directions aim to expand the scalability, fairness, and real-world applicability of Bay-DeepMIL in next-generation educational analytics systems.

5. Conclusion

This study introduces the first Bayesian Deep Multi-Instance Learning (Bay-DeepMIL) framework for student performance prediction, making several important theoretical contributions to the field of educational data mining. Specifically, our work presents: (1) A novel probabilistic formulation that models sparse and irregular academic records through a multi-instance representation; (2) A principled integration of Bayesian inference with multi-head attention to mitigate overfitting while enabling uncertainty-aware prediction; (3) A complete end-to-end framework for robust and interpretable educational risk modeling under data sparsity.

Research Contributions: The proposed Bay-DeepMIL approach offers four core contributions: (1) A novel MIL-based model architecture that transforms sparse academic histories into structured instance-bag representations; (2) The first integration of Bayesian learning and multi-head attention in the education domain, achieving both generalization and interpretability; (3) An effective uncertainty quantification mechanism that enhances the transparency and reliability of predictions; (4) Empirical validation on a real-world 1048-student dataset, where Bay-DeepMIL outperforms FCNN, CLIR, and GRSO-CRN baselines across all major evaluation metrics.

Practical Advantages: Bay-DeepMIL also demonstrates strong applicability in real-world educational settings: (1) It robustly handles incomplete or sparse academic records without requiring imputation; (2) It achieves high

predictive accuracy (94.2%) on large-scale campus data; (3) It supports decision-making by outputting risk scores along with confidence estimates; (4) It enables early intervention by identifying students at risk of failure before performance deteriorates.

Limitations and Future Work: Despite these advances, several limitations remain and suggest promising directions for future research: (1) *Cross-institutional generalizability:* The current model is evaluated on a single-institution, single-discipline dataset. Future work will incorporate federated learning to test its transferability across different academic environments. (2) *Computational cost:* While effective, the current Bayesian implementation incurs high training costs. More efficient variational inference and sampling techniques will be explored. (3) *Structured data dependency:* The reliance on well-organized campus records may limit scalability. Future iterations will consider integrating unstructured behavioral signals and human-in-the-loop refinement to broaden deployment scope.

In summary, Bay-DeepMIL provides a theoretically sound and practically useful framework for student performance prediction. We believe this work lays a foundation for future uncertainty-aware, interpretable, and equitable learning analytics systems that can operate under real-world data constraints.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Funding

This work was jointly supported by the following projects: the Key Research and Development Program of Shaanxi Grant No.2024GX-YBXM-039 and No.2024GX-ZDCYL-02-15, and the Fundamental Research Funds for the Central Universities No.QTZX25034.

References

Anthonysamy, L., & Singh, P. (2023). The impact of satisfaction, and autonomous learning strategies use on scholastic achievement during covid-19 confinement in malaysia. *Heliyon*, *9*.

Bai, G., & Chandra, R. (2023). Gradient boosting bayesian neural networks via langevin mcmc. *Neurocomputing*, *558*, 126726.

Bai, X., Zhang, F., Li, J., Guo, T., Aziz, A., Jin, A., & Xia, F. (2021). Educational big data: Predictions, applications and challenges. *Big Data Research*, *26*, 100270.

Bamber, M. D., & Morpeth, E. (2019). Effects of mindfulness meditation on college student anxiety: A meta-analysis. *Mindfulness*, *10*, 203–214.

Baranyi, M., Nagy, M., & Molontay, R. (2020). Interpretable deep learning for university dropout prediction. In *Proceedings of the 21st annual conference on information technology education* (pp. 13–19).

Biau, G., & Scornet, E. (2016). A random forest guided tour. *Test*, *25*, 197–227.

Blundell, C., Cornebise, J., Kavukcuoglu, K., & Wierstra, D. (2015). Weight uncertainty in neural network. In *International conference on machine learning* (pp. 1613–1622). PMLR.

Cao, J. (2023). Number of students in higher education institutions. http://www.moe.gov.cn/jyb_sjzl/moe_560/2022/quanguo/202401/t20240110_1099530.html.

Charbuty, B., & Abdulazeez, A. (2021). Classification based on decision tree algorithm for machine learning. *Journal of Applied Science and Technology Trends*, *2*, 20–28.

Chen, J., Pi, D., Wu, Z., Zhao, X., Pan, Y., & Zhang, Q. (2021). Imbalanced satellite telemetry data anomaly detection model based on bayesian lstm. *Acta Astronautica*, *180*, 232–242.

Chen, T., & Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (pp. 785–794).

Chen, Y., Wei, G., Liu, J., Chen, Y., Zheng, Q., Tian, F., Zhu, H., Wang, Q., & Wu, Y. (2023). A prediction model of student performance based on self-attention mechanism. *Knowledge and Information Systems*, *65*, 733–758.

Cheng, B., Liu, Y., & Jia, Y. (2024). Evaluation of students' performance during the academic period using the xg-boost classifier-enhanced aeo hybrid model. *Expert Systems with Applications*, *238*, 122136.

Christou, V., Tsoulos, I., Loupas, V., Tzallas, A. T., Gogos, C., Karvelis, P. S., Antoniadis, N., Glavas, E., & Giannakeas, N. (2023). Performance and early drop prediction for higher education students using machine learning. *Expert Systems with Applications*, *225*, 120079.

Deka, B., Nguyen, L. H., & Goulet, J.-A. (2024). Analytically tractable heteroscedastic uncertainty quantification in bayesian neural networks for regression tasks. *Neurocomputing*, *572*, 127183.

Feng, G., & Fan, M. (2024). Research on learning behavior patterns from the perspective of educational data mining: Evaluation, prediction and visualization. *Expert Systems with Applications*, *237*, 121555.

Fernández, D. P., Ryan, M. K., & Begeny, C. T. (2023). Recognizing the diversity in how students define belonging: evidence of differing conceptualizations, including as a function of students' gender and socioeconomic background. *Social Psychology of Education*, *26*, 673–708.

Fu, J., Zhou, W., & Chen, Z. (2024). Bayesian graph convolutional network for traffic prediction. *Neurocomputing*, *582*, 127507.

Ghiasi, G., Lin, T.-Y., & Le, Q. V. (2018). Dropblock: A regularization method for convolutional networks. *Advances in neural information processing systems*, *31*.

Hanaysha, J. R., Shriedeh, F. B., & In'airat, M. (2023). Impact of classroom environment, teacher competency, information and communication technology resources, and university facilities on student engagement and academic performance. *International Journal of Information Management Data Insights*, *3*, 100188.

Kang, H., Xu, Q., Chen, D., Ren, S., Xie, H., Wang, L., Gao, Y., Gong, M., & Chen, X. (2024). Assessing the performance of fully supervised and weakly supervised learning in breast cancer histopathology. *Expert Systems with Applications*, *237*, 121575.

Korkmaz, Y., & Boyacı, A. (2022). milvad: A bag-level mnist modelling of voice activity detection using deep multiple instance learning. *Biomedical Signal Processing and Control*, *74*, 103520.

Li, C., Zhang, Z., Liu, L., Kim, J. Y., & Sangaiah, A. K. (2023). A novel deep multi-instance convolutional neural network for disaster classification from high-resolution remote sensing images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, .

Li, G., Yang, L., Lee, C.-G., Wang, X., & Rong, M. (2020). A bayesian deep learning rul framework integrating epistemic and aleatoric uncertainties. *IEEE Transactions on Industrial Electronics*, *68*, 8829–8841.

Li, J., Li, Z., Liu, L., & Jiao, C. (2022). Hyperspectral target detection via ensemble learning deep multiple instance neural network. In *IGARSS 2022-2022 IEEE International Geoscience and Remote Sensing Symposium* (pp. 875–878). IEEE.

Li, X.-C., Zhan, D.-C., Yang, J.-Q., & Shi, Y. (2021). Deep multiple instance selection. *Science China Information Sciences*, *64*, 1–15.

Liang, J., Liu, H., & Xiao, N.-C. (2024). A hybrid approach based on neural network and double exponential model for remaining useful life prediction. *Expert Systems with Applications*, (p. 123563).

Liu, T., Liu, Y., Liu, J., Wang, L., Xu, L., Qiu, G., & Gao, H. (2020). A bayesian learning based scheme for online dynamic security assessment

and preventive control. *IEEE Transactions on Power Systems*, *35*, 4088–4099.

Liu, Z., Xie, Y., Sun, Z., Liu, D., Yin, H., & Shi, L. (2023). Factors associated with academic burnout and its prevalence among university students: a cross-sectional study. *BMC Medical Education*, *23*, 317.

Lu, X., Zhu, Y., Xu, Y., & Yu, J. (2021). Learning from multiple dynamic graphs of student and course interactions for student grade predictions. *Neurocomputing*, *431*, 23–33.

Ma, Y., Cui, C., Yu, J., Guo, J., Yang, G., & Yin, Y. (2020). Multi-task miml learning for pre-course student performance prediction. *Frontiers of Computer Science*, *14*, 1–10.

Matz, S. C., Bukow, C. S., Peters, H., Deacons, C., Dinu, A., & Stachl, C. (2023). Using machine learning to predict student retention from socio-demographic characteristics and app-based engagement metrics. *Scientific Reports*, *13*, 5705.

Mazaev, G., Crevecoeur, G., & Van Hoecke, S. (2021). Bayesian convolutional neural networks for remaining useful life prognostics of solenoid valves with uncertainty estimations. *IEEE Transactions on Industrial Informatics*, *17*, 8418–8428.

Nayani, S., & P, S. R. (2023). Combination of deep learning models for student's performance prediction with a development of entropy weighted rough set feature mining. *Cybernetics and Systems*, (pp. 1–43).

Niu, T., Chen, B., Lyu, Q., Li, B., Luo, W., Wang, Z., & Li, B. (2024). Scoring bayesian neural networks for learning from inconsistent labels in surface defect segmentation. *Measurement*, *225*, 113998.

Pal, S., Valkanas, A., Regol, F., & Coates, M. (2022). Bag graph: Multiple instance learning using bayesian graph neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence* (pp. 7922–7930). volume 36.

Pérez-Cano, J., Wu, Y., Schmidt, A., López-Pérez, M., Morales-Álvarez, P., Molina, R., & Katsaggelos, A. K. (2024). An end-to-end approach to combine attention feature extraction and gaussian process models for deep multiple instance learning in ct hemorrhage detection. *Expert Systems with Applications*, *240*, 122296.

Phan, M., De Caigny, A., & Coussement, K. (2023). A decision support framework to incorporate textual data for early student dropout prediction in higher education. *Decision Support Systems*, *168*, 113940.

Ramirez, I., Cuesta-Infante, A., Schiavi, E., & Pantrigo, J. J. (2020). Bayesian capsule networks for 3d human pose estimation from single 2d images. *Neurocomputing*, *379*, 64–73.

Rivas, A., Gonzalez-Briones, A., Hernandez, G., Prieto, J., & Chamoso, P. (2021). Artificial neural network analysis of the academic performance of students in virtual learning environments. *Neurocomputing*, *423*, 713–720.

Rodríguez-Hernández, C. F., Musso, M., Kyndt, E., & Cascallar, E. (2021). Artificial neural networks in academic performance prediction: Systematic implementation and predictor evaluation. *Computers and Education: Artificial Intelligence*, *2*, 100018.

do Santos, C. F. G., Colombo, D., Roder, M., & Papa, J. P. (2021). Max-dropout: deep neural network regularization based on maximum output values. In *2020 25th International Conference on Pattern Recognition (ICPR)* (pp. 2671–2676). IEEE.

Serra, R., Kiekens, G., Vanderlinden, J., Vrieze, E., Auerbach, R. P., Benjet, C., Claes, L., Cuijpers, P., Demyttenaere, K., Ebert, D. D. et al. (2020). Binge eating and purging in first-year college students: Prevalence, psychiatric comorbidity, and academic performance. *International Journal of Eating Disorders*, *53*, 339–348.

Struski, Ł., Janusz, S., Tabor, J., Markiewicz, M., & Lewicki, A. (2024). Multiple instance learning for medical image classification based on instance importance. *Biomedical Signal Processing and Control*, *91*, 105874.

Tai, Y., Tan, Y., Zou, E., Lei, B., Fan, Q., & He, Y. (2024). Where to model the epistemic uncertainty of bayesian convolutional neural networks for classification. *Neurocomputing*, *583*, 127568.

Tziolis, G., Spanias, C., Theodoride, M., Theocharides, S., Lopez-Lorente, J., Livera, A., Makrides, G., & Georghiou, G. E. (2023). Short-term electric net load forecasting for solar-integrated distribution systems based on bayesian neural networks and statistical post-processing. *Energy*, *271*, 127018.

Wang, L., Cao, H., Ye, Z., & Xu, H. (2023). Bayesian large-kernel attention network for bearing remaining useful life prediction and uncertainty quantification. *Reliability Engineering & System Safety*, *238*, 109421.

Wang, Y., Gao, H., Liu, J., Fan, X.-l. et al. (2021). Academic procrastination in college students: The role of self-leadership. *Personality and Individual Differences*, *178*, 110866.

Waqas, M., Ahmed, S. U., Tahir, M. A., Wu, J., & Qureshi, R. (2024). Exploring multiple instance learning (mil): A brief survey. *Expert Systems with Applications*, (p. 123893).

Wu, F., Zheng, Q., Tian, F., Suo, Z., Zhou, Y., Chao, K.-M., Xu, M., Shah, N., Liu, J., & Li, F. (2020). Supporting poverty-stricken college students in smart campus. *Future Generation Computer Systems*, *111*, 599–616.

Xia, J., Wang, S., Wang, X., Xia, M., Xie, K., & Cao, J. (2024). Multi-view bayesian spatio-temporal graph neural networks for reliable traffic flow prediction. *International Journal of Machine Learning and Cybernetics*, *15*, 65–78.

Xiang, F., Zhang, Y., Zhang, S., Wang, Z., Qiu, L., & Choi, J.-H. (2024). Bayesian gated-transformer model for risk-aware prediction of aero-engine remaining useful life. *Expert Systems with Applications*, *238*, 121859.

Yang, F.-J. (2018). An implementation of naive bayes classifier. In *2018 International conference on computational science and computational intelligence (CSCI)* (pp. 301–306). IEEE.

Yang, T., Zhu, S., & Chen, C. (2020a). Gradaug: A new regularization method for deep neural networks. *Advances in neural information processing systems*, *33*, 14207–14218.

Yang, Z., Yang, J., Rice, K., Hung, J.-L., & Du, X. (2020b). Using convolutional neural network to recognize learning images for early warning of at-risk students. *IEEE Transactions on Learning Technologies*, *13*, 617–630.

Zhao, X., & Wang, D. (2023). Grit, emotions, and their effects on ethnic minority students' english language learning achievements: A structural equation modelling analysis. *System*, *113*, 102979.

Zhou, T., Han, T., & Droguett, E. L. (2022). Towards trustworthy machine fault diagnosis: A probabilistic bayesian deep learning framework. *Reliability Engineering & System Safety*, *224*, 108525.

Jiayang Huang was born in 1995. She received her bachelor's degree from Shandong University, Jinan, China, in 2017, and her Ph.D. degree from Xidian University, Xi'an, China, in 2023. Currently, she is a post-doctoral researcher at Xidian University. Her research focuses on human-computer interactions, wearable sensing, machine learning, and smart education, primarily emphasizing data analytics in smart education to enhance learning outcomes and optimize educational systems.

Keyi Yang received the B.Eng and M.Eng degree from Xidian University, China respectively in 2017 and 2020. Her primary research interests include data analytics, machine learning, neural networks, and wearable sensing.

Quan Wang (CCF Fellow) was born in 1970. He received his B.Sc, M.Sc, and Ph.D degrees from Xidian University, Xi'an, China. He is currently a professor at Xidian University. His research interests include smart education, input and output technologies and systems, heterogeneous parallel computing, and human-computer interactions.

Pengfei Yang (Senior Member, CCF) received the B.Sc., M.Sc., and Ph.D. degrees from Xidian University, Xi'an, China, in 2008, 2011, and 2015, respectively. He was an academic visitor for one year at the University of Leeds, Leeds, U.K. He is currently an Associate Professor at Xidian University. His research interests include smart education, embedded system architecture, heterogeneous parallel computing, and human-computer interactions.

Ziling Ruan received the B.Eng and M.Eng degree from Xidian University, China respectively in 2017 and 2020. Currently, she is working towards the Ph.D degree at the University of Leeds, Leeds, U.K. Her primary research interests include wearable sensing, machine learning, neural networks, and data mining.

Jiaxi Wang received a B.Eng degree from Xidian University, China in 2018. Currently, she is working towards an M.Sc. degree in electronic information, at Xidian University. Her primary research interests include human-computer interactions, machine learning, and smart education.

Zhi-Qiang Zhang received his B.Eng in Computer Science and Technology from Tianjin University in 2005 and Ph.D. in Electrical Engineering from the University of Chinese Academy of Sciences in 2010, he joined Imperial College London as a research associate, where he worked for five and a half years before moving to the University of Leeds in 2016. He currently holds the position of Chair of Biomedical Engineering. His research lies at the intersection of artificial intelligence, robotics, and biomedical engineering.