Journal of Theoretical and Applied Information Technology
15th August 2025. Vol.103. No.15

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

MODEL AGNOSTIC META LEARNING – LONG SHORT-TERM MEMORY FOR LEARNING STYLE CLASSIFICATION

AMRUTH K JOHN[1], BINU THOMAS[2]

[1,2] Department of Computer Applications, Marian College Kuttikkanam (Autonomous), Idukki, India

E-mail: [1]amruth.john@mariancollege.org, [2]binu.thomas@mariancollege.org

ABSTRACT

An adaptive learning system aims to enhance the effectiveness of the educational process by tailoring it to individual students. A key aspect of this adaptation involves identifying the most suitable learning approach, based on Visual, Auditory, and Kinesthetic (VAK) learning styles. However, accurately classifying learning styles remains a challenge due to the presence of concept drift, which affects the network's ability to generalize across different learners. To address this, the Model Agnostic Meta Learning – Long Short-Term Memory (MAML-LSTM) model is proposed in this research study for effective learning style classification. MAML is incorporated into the LSTM network to identify shifts in classification patterns and adapt to new learners quickly. Rather than retraining the network from the beginning, the model dynamically fine-tunes the LSTM in response to concept drift, thereby improving its generalization capability. The MAML-LSTM integration enables rapid adaptation to concept drift by fine-tuning on limited new data, eliminating the need for complete retraining. This enhances the model's ability to maintain high classification accuracy across dynamic learner behaviors. Additionally, Local Interpretable Model-agnostic Explanations (LIME) are employed after classification to highlight key features, ensuring greater transparency and interpretability. The proposed MAML-LSTM achieves 97.77% accuracy, 97.72% precision, 97.72% recall, 97.72% F1-score, 97.72% specificity, and 99.81% AUC on the VAK learning style dataset, outperforming existing algorithms.

Keywords: *Auditory, Kinesthetic, Learning style, Long Short-Term Memory, Model Agnostic Meta Learning and Visual.*

1. INTRODUCTION

Recent advancements in the education system, driven by Information Technology (IT) and the internet, have enabled predictive models to improve the data services provided by educational institutions [1]. E-learning, an educational technique that incorporates technical tools, is facilitated through Learning Management Systems (LMSs) [2]. Through LMSs, students can access lecture materials, discussion boards, chat rooms, and retrieve assignments provided by instructors [3]. Student activities and engagement in online learning environments are monitored through platforms like Moodle, with data recorded in Moodle logs [4]. Despite its advantages, e-learning presents several challenges, as undergraduates often show low engagement and frequently leave virtual classrooms. Therefore, understanding students' learning preferences by considering their learning styles in different situations is crucial [5,6]. Students experience various phases of knowledge acquisition throughout the learning process. The concept of learning style refers to the approach a student adopts to effectively analyze and comprehend information [7]. Students recognize their preferred learning methods based on familiarity with specific techniques. However, Moodle cannot automatically identify students' learning preferences [8]. Student behavior is analyzed based on the number of times they access specific e-learning modules in Moodle. Completing a learning style questionnaire is necessary to determine the most suitable learning style for each student [9].

Students access video lectures and educational sources on Massive Open Online Course (MOOC) platforms. Additionally, upon successful completion of a course, students receive a certificate [10]. The content on MOOC platforms is typically free from punctuation, grammar and spelling errors. Although MOOC platforms are widespread, they suffer from high dropout rates and poor performance metrics, often leading to student frustration. As a result, key contributing factors include a lack of interest in the course, low participation, and difficulty in tracking activities and resources for assessments [11]. Hence, in an online education system, student participation is a crucial component of a course's success. While a virtual learning environment with certificate

completion is referred to as a MOOC, online learning platforms are generally considered more hybrid [12]. Student satisfaction and quality of educational experience are primarily linked to student participation. To minimise dropout rates, it is crucial to understand how students engage and sustain their interest in professional education [13]. An improved learning environment supports and promotes self-control and motivation, enabling students to stay focused and perform well. Understanding how students learn and interact within online learning platforms is fundamental to developing effective learning environments [14]. Optimal learning techniques and customization of learning environments are necessary, particularly when guided by high student participation. Recently, learning analytics has been applied to predict student performance through advancements in educational data analysis. Compared with MOOC platforms, Virtual Learning Environments (VLEs) provide more structured instruction and facilitate monitoring of student activities and assessments, aimed to transform passive learners into active participants [15]. Long Short-Term Memory (LSTM) networks are employed to model sequential dependencies on learning behavior data, effectively extracting long-term patterns. However, concept drift, referring to changes in user behavior over time, reduces the generalization ability of the model. To address this, Model Agnostic Meta Learning (MAML) is integrated with LSTM, enabling the model to quickly adapt to new learners through fine-tuning on limited data, rather than retraining from scratch. This approach enhances classification accuracy by accommodating shifts in data distribution. Meta-learning dynamically adjusts hyperparameters and thresholds in response to behavioral changes, thereby improving adaptability and robustness.

1.1 Objective

The main objective of the study is to developed a hybrid MAML-LSTM model to adaptively classify the learning styles as visual, auditory and kinesthetic, while efficiently addressing the challenges of concept drift. The performance of this model is evaluated in terms of the measures of accuracy, precision, recall, f1-score, specificity and AUC using the VAK learning style dataset. To ensure interpretability, the study includes LIME to highlight the much significant features of model predictions.

Contributions

The primary contributions of this research are described as follows:

- Term Frequency – Inverse Document Frequency (TF-IDF) and GloVe embeddings are employed during the feature extraction phase to capture meaningful features that differentiate between the classes of different learning styles.
- A Model Agnostic – Meta Learning (MAML) - Long Short-Term Memory (LSTM) model is developed during classification to efficiently manage concept drift and enhance generalization ability.
- The integration of the MAML-LSTM model is developed to dynamically fine-tune the LSTM parameters using meta-learning, enabling fast adaptation to shifting learner behavior with less data and without complete retraining, thereby enhancing robustness and generalization.
- Finally, Local Interpretable Model-agnostic Explanations (LIME) are applied after the classification process to highlight the key features for ensuring greater model transparency and interpretability.

This research paper is further organized as follows: Section 2 analyzes the existing algorithms along with their advantages and limitations. Section 3 presents the details of the proposed algorithm for learning style classification. Section 4 provides the results and discussion of the proposed algorithm, and Section 5 concludes the research.

2. LITERATURE REVIEW

Sayed et al. [16] presented an integrated method for classifying learners based on their learning activity clicks by integrating Machine Learning (ML) techniques such as K-Nearest Neighbor (KNN), Random Forest (RF), Support Vector Machine (SVM), and Logistic Regression (LR) with semantic integration, which was used to map learning activities to the VAK learning styles. This process ensured the classification of learners and identified their preferred learning techniques. Learning styles provided a reliable basis for validation methods and strategies.

Kanchon et al. [17] explored diverse endeavours for formulating an efficient technique to determine a learner's chosen learning styles and adapt a learning content to align with the chosen style. The analysis revealed that web tracking of learners for activity classification and individual responses for feedback classification were highly effective in detecting learning styles such as visual, auditory, and

kinesthetic. Additionally, Decision Tree, RF, SVM, LR, and Convolutional Neural Network (CNN), each with optimized hyperparameters and the Synthetic Minority Oversampling Technique (SMOTE), were employed to classify learner behaviors.

Villegas-Ch et al. [18] developed a personalized learning method using ML techniques by adapting educational content to classify various learning styles. Focusing on a cohort of students, classification techniques and neural networks were implemented to diagnose learning styles and personalize educational resources. The outcomes showed that students' average grades experienced a significant increase. Moreover, engagement improved through substantial interaction with educational materials, aligning with individual learning preferences.

Sayed et al. [19] introduced a technique for analyzing the student engagement trends in Virtual Learning Environments (VLE), and defined student prevalent preferences and learning styles to formulate recommendations for effective learning evaluation approaches. This hybridization method was linked to different activities within the VAK learning model, and therefore, with different learning preferences driven by the patterns and behaviours throughout the learning process.

Muhammad et al. [20] implemented a learning style detection method named Graph Representation Learning – Learning Style (GRL-LS), based on graph representation learning. This model used a bipartite graph representing interactions among various groups of learners and learning sources. Then, a graph embedding method was introduced to understand the latent representation of learners and resources. Then, learned representation was planned to Felder-Silverman Learning Style Model (FSLSM) for detecting and grouping learners by K-means algorithms. The implemented model was employed under various education settings and customized to different learning methods. The primary factors requiring this adaptation included identifying an ideal learning approach for students, based on the VAK learning styles. However, classifying learning styles remains challenging due to the difficulty in handling concept drift, which reduced the network's generalization ability.

In order to address the aforementioned challenges, this study proposes the MAML-LSTM model for effective classification of learning styles. MAML is incorporated into the LSTM network to identify changes in classification patterns and enable rapid adaptation to new learners.

To ensure strong alignment between the reviewed literature and the present study, it is highlighted that the VAK learning style dataset exhibits non-stationary behavior due to varying learner preferences over time. While previous studies employ static classifiers or ensemble models, they do not address the dynamic shifts in user behavior. The proposed MAML-LSTM framework offers a meta-learning algorithm that fine-tunes the model to accommodate such shifts. This connection between the drawbacks identified in previous work and the nature of the collected data provides a clear methodological direction. By leveraging MAML's capability to rapidly adapt to new tasks, the MAML-LSTM model dynamically adjusts to changes in learner behavior over time, efficiently addressing the non-stationarity inherent in the VAK dataset.

Additionally, rather than retraining the network from the beginning, the model fine-tunes the LSTM dynamically as concept drift occurs, thereby improving the generalization ability of the network. Finally, LIME is applied after the classification process to highlight key features, ensuring greater transparency and interpretability.

3. PROPOSED METHOD

The proposed MAML-LSTM method is presented for the precise classification of learning styles. The VAK learning style dataset is used and pre-processed through stopword removal, lemmatization, and label encoding to enhance data quality. TF-IDF and GloVe embedding are employed in the feature extraction phase to capture meaningful features that differentiate the classes of learning styles. In the classification phase, MAML-LSTM is used to accurately classify the learning styles. Finally, LIME is applied to highlight key features and ensure model interpretability.

Study design

The study follows a structured experimental design to classify learning styles using the VAK dataset. Initially, raw data is pre-processed through stopword removal, lemmatization, and label encoding to standardize the inputs. Feature extraction is performed using TF-IDF and GloVe embeddings to capture both statistical and semantic features.

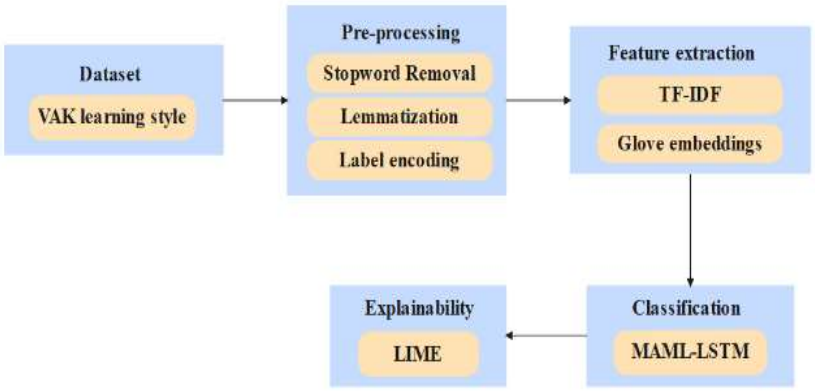

Figure 1: Process of learning style classification

The primary model combines LSTM with MAML to address concept drift and enhance adaptability. LIME is employed after classification to improve model interpretability. Figure 1 illustrates the process of learning style classification.

3.1 Dataset

The VAK learning style dataset is employed in this study for data collection [21]. The dataset includes three classes with 5527 visual samples, 4572 kinesthetic samples and 4496 auditory samples. Figure 2 represents the dataset distribution.

3.2 Pre-processing

Pre-processing involves the following steps:

- Stopword removal – Stopwords are words that frequently appear in a document but carry minimal meaningful information. Examples include common English words such as an, as, are, and, and. Removing stopwords reduces vector space and enhances performance through improving computation speed, calculation efficiency, and overall accuracy [22]. Therefore, eliminating stopwords removes low-information content, without negatively impacting the training process.
- Lemmatization – Lemmatization is defined as the vocabulary and morphological analysis of words to eliminate inflectional endings and return a base form, called the lemma. This process replaces a word with its root form, standardizing variations that may convey similar meanings based on context. Lemmatization improves text processing by unifying word forms, thereby enhancing accuracy.
- Label encoding – Label encoding converts categorical labels in the dataset, such as visual, auditory, and kinesthetic, into numerical values, facilitating the training process of the model.

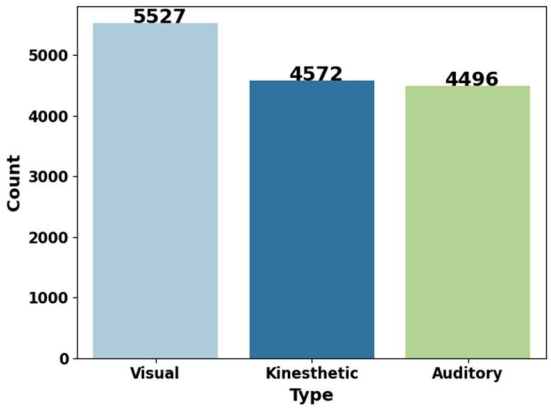

Figure 2: Dataset distribution

3.3 Feature extraction

The pre-processed data is given as input for feature extraction to capture meaningful features from the text. In this phase, TF-IDF and GloVe embedding techniques are employed to extract features and differentiate them across the dataset classes.

3.3.1 TF-IDF

Term Frequency (TF) and Inverse Document Frequency (IDF) are two widely used elements in text classification, collectively referred to as TF-IDF. TF-IDF is a statistical measure that evaluates the significance of a word in a document relative to a set of documents [23]. This is achieved by multiplying the inverse document frequency of a word across the documents. The TF-IDF value is calculated by multiplying the TF and IDF values of

a word, and its mathematical expression is given in Equation (1).

$$TF - IDF = TF \times IDF \quad (1)$$

3.3.2 GloVe Embeddings

The GloVe method is an efficient technique that utilizes global corpus statistics and optimizes a learning process based on a context window. The primary objective is to vectorize words and generate word vectors from the input corpus. The implementation of the algorithm proceeds as follows: first, a word co-occurrence matrix is constructed based on the entire corpus. Then, word vector learning is performed using the co-occurrence matrix in conjunction with the GloVe method. The mathematical expression for the GloVe method is presented in Equation (2).

$$J = \sum_{i,j}^{N} f(X_{ij}) \left(V_i^T V_j + b_i + b_j - In(X_{ij})\right)^2 \quad (2)$$

In the above Equation (2), X represents a cooccurrence matrix and the number of times a word occurs in single window is denoted as X_{ij}. The size of a window is generally 5 to 10, and V_i and V_j are word vectors of i and j words, b_i and b_j represent deviation terms, N represents the dimension of cooccurrence matrix and f represents weight function, where f contains the below characteristics.

- When a count of cooccurrence of words is 0, the weight is also 0.
- When the co-occurrence count is high, the weight does not decrease, that is $f(x)$ maintains continuity and is non-decrementing.
- When words exist with high frequently, there is null weight, and $f(x)$ is employed for smaller values. The weight function $f(x)$ and its mathematical expression is given as Equation (3).

$$f(x) = \begin{cases} (x/x_{max})^{\alpha}, & x < x_{max} \\ 1, & x \geq x_{max} \end{cases} \quad (3)$$

The experimental results show optimal performance when $x_{max} = 100$ and $\alpha = 0.75$, as expressed in Equation (3). The GloVe directly utilizes the corpus word vectors for measurement, offering high manoeuvrability and flexibility.

3.4 Classification

In the classification phase, LSTM is integrated with MAML to address the challenge of concept drift and to enhance the generalization capability of LSTM. A detailed explanation of LSTM and MAML is provided below.

3.4.1 LSTM

LSTM is a prominent variant of the Recurrent Neural Network (RNN) that has achieved significant success in recent years. In LSTM, the memory cell is the central component and includes a gating mechanism. Each LSTM cell typically contains three main gates: input, forget, and output gates. To recognize long-term dependencies, LSTM uses individual cells that update based on the current input value. The parameters of the LSTM used in this research include 20 epochs, categorical cross-entropy as the loss function, a batch size of 64, the Adam optimizer, and the softmax activation function. The mathematical expressions for the three gates are given in Equation (4) to Equation (6).

$$i(g) = \sigma[w_i \cdot (y_{r-1}, h_r)] + b_i) \quad (4)$$
$$f(g) = \sigma[w_f \cdot (y_{r-1}, h_r)] + b_f) \quad (5)$$
$$o(g) = \sigma[w_o \cdot (y_{r-1}, h_r)] + b_o) \quad (6)$$

In the above equations, $i(g), f(g)$ and $o(g)$ represent input gate, forget gate and output gate, and σ represents the sigmoid activation function. The b_i, b_f and b_o represent the bias functions, and w_i, w_f and w_o represent the weight functions. The y_{r-1} represents the hidden state and h_r is the input state. Additionally, mathematical expressions for hidden and cell states are given in Equations (7)-(9).

$$c_i = \tanh[w_c \cdot (y_{r-1}, h_r)] + b_c \quad (7)$$
$$c_{i1} = f(g) \circ c_{r-1} + i(g) \circ c_i \quad (8)$$
$$y_r = o(g) \circ \tanh(c_i) \quad (9)$$

In the above equations, the hyperbolic activation function, weight and bias function in terms of cell state are denoted as $tanh, w_c$ and b_c, respectively.

3.4.2 Meta learning - LSTM

Meta-learning is a learning method that learns from past, similar tasks and predicts unseen, relevant tasks. Specifically, it aids in selecting optimal hyperparameters to active learning tasks in streaming data, thereby enhancing the model's ability to recommend optimal scores. Meta-learning using hyperparameter tuning is successfully employed in various domains, including image processing.

By employing stream-dependent active learning with uncertainty sampling for query labeling, the stream classifier is trained to reduce labeling costs. However, this process involves hyperparameters that require tuning for each task. The Z value, or ambiguity threshold, defines the level of uncertainty used in the active learning method to determine the value of a sample for labeling.

The method aims at dynamically tuning the Z value by layering a meta-learning approach on top of

the active learning method. This approach, derived from a meta-learning mechanism, is responsible for selecting an appropriate Z for each stream chunk. Since the data stream modifies its behavior due to concept drift, the meta-method continuously determines the Z value. The method assigns triggers based on a change detector, which identifies potential behavioral modifications representing concept drift. When a probable drift point is detected, data samples limited by the change detector are used to extract meta-features. These samples are then passed to the meta-method to determine a new Z value for uncertainty sampling. The implementation of the meta-learning method involves five phases:

Meta Feature Extraction – This phase represents the characteristics of the event stream based on lightweight temporal time series attributes.

Meta-Target Definition – This phase detects an appropriate Z value which offers an appropriate trade-off among accuracy and low label querying. This is essential for exploring suitable Z values capable of covering the probability search space defined by the trade-off moulds.

Meta-database – In this phase, meta-features and meta-targets are combined to develop meta-samples, which are then used to train the meta-method.

Meta Learner – This phase involves the induction of the meta-method using the acquired meta-instances. The meta-method serves as the final step, capable of recommending a suitable Z.

Meta Recommending – When a change is identified, its samples have their features extracted to predict a new Z through meta-methods. The output of the meta-method serves as input for uncertainty sampling, which chooses labeling instances for the stream classifier. LSTM is used to model the sequential dependencies in learning behavior data, effectively extracting long-term patterns. However, concept drift, referring to changes in user behavior over time, reduces the model's generalization capability. To address this, MAML is integrated with LSTM, enabling the method to quickly adapt to new learners by fine-tuning on limited data rather than retraining from scratch. This process enhances classification accuracy by accommodating shifts in data distribution. Meta-learning dynamically adjusts hyperparameters and thresholds in response to behavioral changes, thereby improving adaptability and robustness.

3.4.3 MAML with LSTM

In the proposed model, MAML is integrated with LSTM to enable adaptive learning style classification in the presence of concept drift. While LSTM efficiently captures long-term dependencies in sequential learning behavior data, it struggles to generalize when learner behavior patterns shift over time. MAML addresses this by incorporating a meta-learner that fine-tunes the parameters of LSTM on a small amount of new data, allowing rapid adaptation without complete retraining. When a drift in behavior is identified, MAML uses prior learning experience to quickly optimize LSTM for the new task distribution, preserving classification accuracy and robustness. This integration allows the model to dynamically adapt to evolving learning patterns while maintaining effective performance across different learner profiles.

Algorithm for best Z selection

Input – R, the classification outcomes for candidate Z values, containing both accuracy and query quantities for choosing margin to top values.
Output – Optimal Z value with lesser query rate within the top chosen interval.
$R_{acc} \leftarrow Z$ with high accuracy in R
$T \leftarrow \{R_i \in R | ACC(R_{acc}) - s \leq ACC(R_i) \leq ACC(R_{acc})\}$;
$R_{top} \leftarrow argmin_T(QRY(T))$;
Return R_{top};

Algorithm for Meta-Recommending

Input: S data stream, α_x drift detector, β_x classifier, θ_x meta-learning method, p_β number of samples utilized for pre-train classifier.
$Z \leftarrow 0.5$;
Pre-trained on initial samples of S;
for $S_i \in \{S_{p\beta+1}, \dots, S_n\}$ do
    $E_i \leftarrow run\ ACT_z^\beta$ on S_i and return the error from prediction;
    Update α_x with E_i;
    if α change detected then
        $F \leftarrow$ features produced from $\{S_{last\_drift}, \dots, S_i\}$
        $Z \leftarrow$ prediction from θ_x with input F:
    end
end

3.5 Local Interpretable Model-Agnostic Explanations (LIME)

LIME (Local Interpretable Model-Agnostic Explanations) is an algorithm designed to locally approximate complex classifiers using interpretable models to provide precise explanations of its predictions. It focuses on two key features: interpretable representation, which offers a

qualitative understanding of the model's decisions, and local fidelity, which measures the trustworthiness of the explanation near the predicted samples. The term model-agnostic means that LIME can explain any classification model by treating it as a black box. LIME is particularly useful for text-based models and enhances the interpretability of complex datasets, making it a valuable tool for understanding and explaining predictions in machine learning.

Algorithm for LIME explanations

Input – Classifier f – Black box method to explained
Instance x – Data sample to explained, N – Number of instances, π_x – distance measure, the function which calculates distance among samples. The $\Omega(g)$ complexity measure is, a measure of complexity for interpretability.
Output - $\varepsilon(x)$ Produced explanation for method's prediction on sample x
$Z \leftarrow \{\}$ Initialize an empty set for storing perturbed instances. Select samples for interpretation and perturbing.
$for\ i\ \in \{1, 2, 3, \dots, N\}$ do
$Z \leftarrow non\ zero\ instances\ (x, \pi_x)$ process search for non-zero samples
$z' \leftarrow perturbed\ sample\ (z)$ Perturb the non-zero elements
$Z \leftarrow Z + z'$ Add perturbed instance to set Z
end for
Fix weights for instances
$weights \leftarrow fix\_weights\ (Z)$ Fix weights for perturbed instances depended on π_x
Learn interpretable method and develop the explanations:
$g \leftarrow learn_{model}(Z, weights)$ Develop a learning method with weights
$untruthiness \leftarrow measure\ untruthiness\ (f, g, s, \pi_x)$ Measure untruthiness by weighted instances
Return $\varepsilon \leftarrow optimize\_explanation\ (untruthiness, \Omega)$ optimize for explanations by reducing $\mathcal{L}(f, g, \pi_x) + \Omega(g)$

The term G represents the class of interpretable methods, with $g \in G$ representing a specific method represented through visual or text artifacts. The domain of g, represented as $\{0,1\}^d$ indicates the presence or absence of interpretable elements. For each $g \in G$ that is interpretable, $\Omega(g)$ is used to calculate the complexity. In the case of a function $f: R^d \rightarrow R$ which requires interpretability, $f(x)$ denotes the probability that x belongs to a specific category. To define the locality around x, the function $\pi_x(z)$ is used to calculate the distance between the instances z and x. Finally, $L(f, g, \pi_x)$ quantifies the degree of untruthiness when explaining f within the local region defined by π_x. The mathematical expression for LIME is given in Equation (10).

$$\epsilon(x) = argmin_{g \in G} \mathcal{L}(f, g, \pi_x) + \Omega(g) \quad (10)$$

In the above Equation (10), $\mathcal{L}(f, g, \pi_x)$ and $\Omega(g)$ respectively denote the interpretability and local fidelity. The empty set Z is initialized for storing the non-zero samples selected from the linear method. This denotes that N data samples around x' are randomly disturbed. The disturbed instances are described as $z' \in \{0,1\}\ d'$ and includes certain non-zero components of x'. The actual representation of instance is redetermined as $z \in R^d$. In classification, $f(x)$ represents the possibility that z belongs to a specific class. The perturbed instances are assigned to set Z and given to the black box method. The $f(x)$ is utilized for obtaining classification labels. The further phase is to fix the weights for selecting the instances. The main aim of LIME is to develop a better local approximation by π_x where instances with greater weight are present close to x', and those with lesser weights are farther from x'. Hence, for learning the interpretable method, LIME fixes the weights for perturbed instances by its proximity to x'. Instances near to x' are provided more weight and instances far from x' are provided lesser weights. The method with perturbed data instances Z are utilized for developing the learning method by applying weights in the form $g(z') = wg \times z'$. The mathematical expression for a new function is given in Equation (11).

$$\mathcal{L}(f, g, \pi_x) = \sum_{z, z' \in Z} \pi_x(z) \big(f(z) - g(z')\big)^2 \quad (11)$$

In the above Equation (11), the weight $\pi_x(z) = e^{\frac{-D(x,z)^2}{\sigma^2}}$ is defined using the distance function D, with σ controlling the width of the locality. Given dataset Z of perturbed and weighted instances with integrated labels is optimized for explanation $\varepsilon(x)$. By default, LIME uses a linear interpretable model with sparse attributes, which is trained on these weighted instances and provides meaningful explanations for the prediction. The transformed instance x' represents the interpretability of the original input, and the learned linear weights indicate the importance of each feature in driving the model's prediction.

4. EXPERIMENTAL EVALUATION

The performance of the proposed MAML-LSTM algorithm is simulated in python 3.7 environment with system configurations being i5 processor, 8 GB

RAM and Windows 10 (64 bit) operating system. The performance of MAML-LSTM is evaluated based on the metrics of precision, specificity, recall, f1-score and accuracy. The mathematical expressions for these metrics are given in Equations (12) – (16).

$$Accuracy = \frac{TP+TN}{Total\ no.of\ classes} \quad (12)$$

$$Recall = \frac{TN}{TN+} \quad (13)$$

$$Specificity = \frac{TP}{TP+FN} \quad (14)$$

$$Precision = \frac{TP}{TP+FP} \quad (15)$$

$$F1-score = \frac{2\times precision\times recall}{precision+rec} \quad (16)$$

Table 1 displays the performance outcomes evaluated across individual classes of the dataset with visual, kinesthetic and auditory learning styles based on the metrics of recall, precision and f1-score. The proposed method demonstrates an average precision of 97.72%, an average recall of 97.72% and an average f1-score of 97.72%. Table 2 presents a comprehensive comparison of the proposed MAML-LSTM model against several existing classification algorithms, including RNN, LSTM, RNN-LSTM, and CNN-LSTM, evaluated on both the VAK Learning dataset and the Student Performance & Learning Style dataset based on the same performance metrics. On the VAK dataset, MAML-LSTM significantly outperforms all baseline models, achieving 97.77% accuracy, 97.72% across precision, recall, F1-score, and specificity, and a notably high AUC of 99.81%. Similarly, on the Student Performance & Learning Style dataset, the proposed method achieves superior results with 98.90% accuracy, 98.93% precision, 98.88% recall, 98.89% F1-score, 99.92% specificity, and an AUC of 99.25%.

Table 1: Performance of individual classes

Classes	Precision (%)	Recall (%)	F1-score (%)
Visual	97.64	97.53	97.59
Kinesthetic	97.21	97.32	97.26
Auditory	98.32	98.32	98.32
Average	97.72	97.72	97.72

Table 2: Performance of proposed classifier with different classifiers

Methods	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	Specificity (%)	AUC (%)
VAK Learning						
RNN	87.26	87.42	87.30	87.16	87.20	91.55
LSTM	91.67	90.45	90.10	90.27	92.30	93.82
RNN-LSTM	93.12	92.80	92.35	92.57	94.05	95.45
CNN-LSTM	95.60	95.35	95.10	95.22	96.00	97.02
MAML-LSTM	97.77	97.72	97.72	97.72	97.72	99.81
Student Performance & Learning Style Dataset						
RNN	93.21	92.83	92.74	92.68	94.13	95.27
LSTM	95.14	94.87	94.72	94.69	95.61	96.33
RNN-LSTM	96.24	96.12	95.97	96.18	96.93	97.43
CNN-LSTM	97.33	97.18	97.04	97.26	98.02	98.29
MAML-LSTM	98.90	98.93	98.88	98.89	99.92	99.25

Table 3 represents the performance analysis of meta learning with different performance metrics. The various existing algorithms like periodic retraining, ensemble learning, transfer learning and drift detection are considered to evaluate the performance of meta learning. While existing algorithms require retraining the network from the beginning, meta-learning identifies changes in VAK patterns and quickly adapts to new learners. Furthermore, instead of retraining, it fine-tunes the network dynamically when concept drift occurs. The developed meta learning obtains 97.77% accuracy, 97.72% precision, 97.72% recall, 97.72% f1score, 97.72% specificity, 99.81% AUC-ROC at a training time of 14.62s, and 210MB computational time.

Table 4 represents the performance of a proposed algorithm with k-fold validation.

Table 3: Performance of meta learning with traditional algorithms

Methods	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	Specificity (%)	AUC-ROC (%)	Training time	Computational time (MB)
Periodic Retraining	85.32	84.90	85.10	84.95	86.00	88.20	78.35	512
Ensemble learning	91.45	91.10	91.30	91.20	92.50	94.30	20.58	1024
Transfer learning	93.82	93.50	93.70	93.60	94.10	96.00	35.21	356
Drift detection	95.30	95.10	95.20	95.15	96.00	97.50	48.90	425
Meta-learning	97.77	97.72	97.72	97.72	97.72	99.81	14.62	210

Table 4: K-fold validation

K-Value	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
2	95.73 ± 0.22	96.07 ± 0.31	96.05 ± 0.28	96.05 ± 0.26
3	96.83 ± 0.19	96.95 ± 0.24	96.93 ± 0.27	96.93 ± 0.23
5	97.77 ± 0.17	97.72 ± 0.22	97.72 ± 0.20	97.72 ± 0.21
6	90.65 ± 0.25	97.27 ± 0.28	97.25 ± 0.30	97.25 ± 0.29
7	96.58 ± 0.21	96.42 ± 0.26	96.71 ± 0.24	96.54 ± 0.22

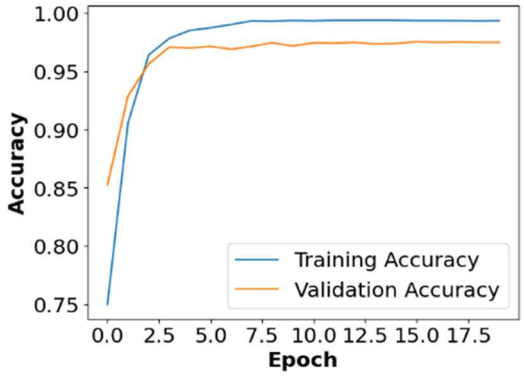

Figure 3: Accuracy vs Epochs

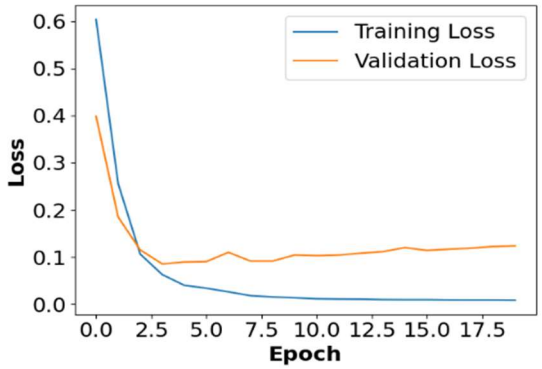

Figure 4: Loss vs Epochs

Figure 3 represents the accuracy vs epochs graph for the proposed algorithm, while Figure 4 represents the loss vs epochs graph for the proposed algorithm. Figure 5 represents ROC curve for proposed algorithm, and Figure 6 represents the confusion matrix for proposed algorithm. Figure 7 represents explainability of auditory learning. Auditory learning is integrated with spoken language, discussion and verbal processes. The word "speak" is a strong indicator of auditory learning, which explains why the method classifies the text as auditory. The kinesthetic and visual classes have a 0% probability, as there are no significant terms relevant to kinesthetic or visual representation.

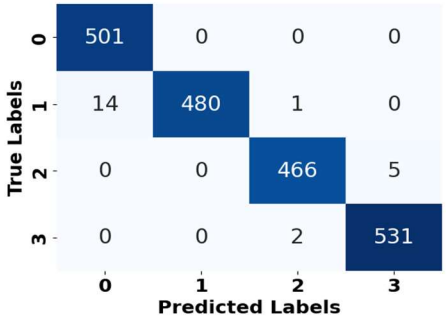

Figure 5: Confusion matrix

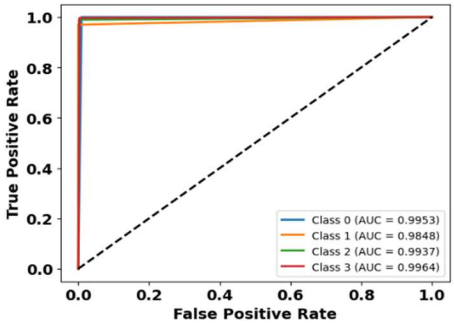

Figure 6: ROC curve

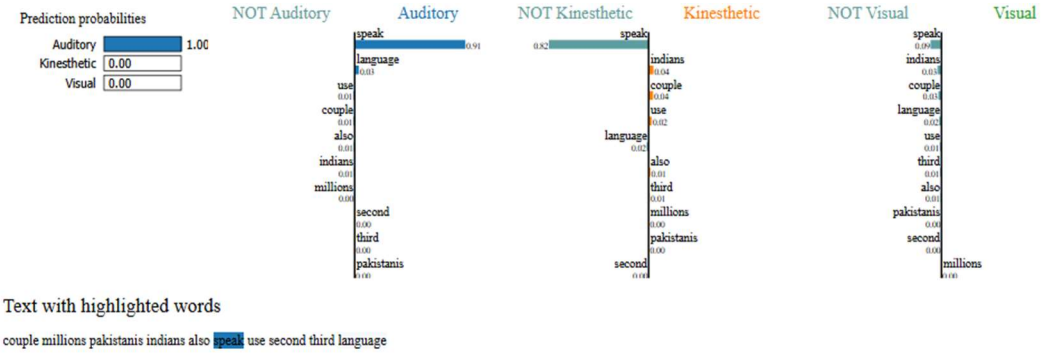

Figure 7: Explainability of auditory learning

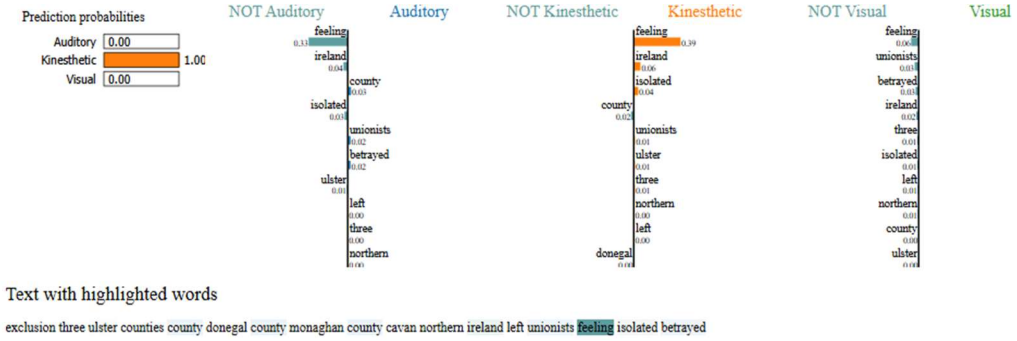

Figure 8: Explainability of kinesthetic learning

Figure 8 represents the explainability of kinesthetic learning. Kinesthetic learning is linked with emotions, sensations, and physical experiences. The word "feeling" is a high indicator of emotional or physical experience, aligning with the kinesthetic learning style. There are no auditory or visual relevant words, resulting in 0% probabilities for the auditory and visual classes.

4.1 Comparative analysis

Table 5 presents a comparative analysis of the developed algorithm based on the metrics of accuracy, f1-score, precision and recall. The existing algorithms like RF [16], Blending [17] and RF [18] with VAK learning style dataset are considered for comparison. The proposed MAML-LSTM obtains 97.77% accuracy, 97.72% precision, 97.72% recall, 97.72% f1-score, 97.72% specificity and 99.81% AUC when compared to existing algorithms. Figure 9 represents the explainability of visual learning. Visual learning is associated with seeing, images, and spatial representation. The words "picture" and "image" are directly linked to visual perception, which explains why the method classifies the text as 100% visual. There are no auditory or kinesthetic-related terms present, which justifies the 0% probability assigned to the auditory and kinesthetic classes.

Table 5: Comparative analysis of MAML-LSTM with existing algorithms

Methods	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	Specificity (%)	AUC
RF [16]	98	97	99	99	NA	NA
Blending [17]	97.56	96.94	96.59	96.76	96.41	0.96
RF [18]	98	97	99	98	NA	NA
Proposed MAML-LSTM	97.77	97.72	97.72	97.72	97.72	99.81

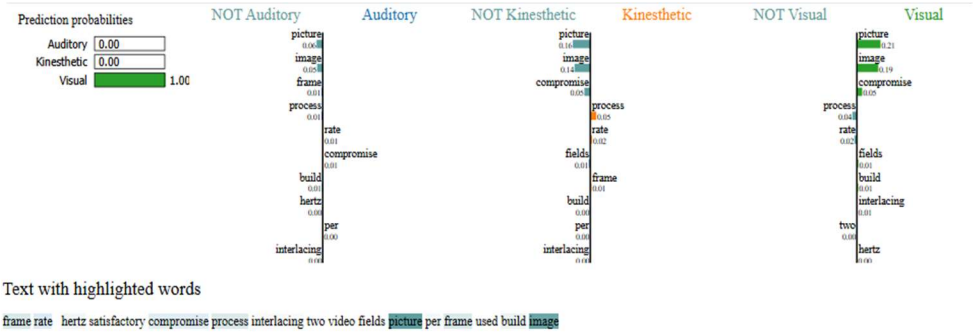

Figure 9: Explainability of visual learning

4.2 Discussion

The proposed MAML-LSTM model demonstrates improved performance compared to recent studies such as RF [16], Blending [17], and RF [18] using the VAK learning style dataset for learning style classification. While these models showed promising results, they lacked adaptability to concept drift. Our proposed meta-learning-based algorithm provides rapid adaptation to changes in learner behavior, resulting in consistently high accuracy and AUC scores. However, this study is limited by its reliance on a single dataset, which may not generalize across different educational platforms. Moreover, although LIME improves interpretability, it focuses on local explanations and does not fully capture the global model behavior.

5. CONCLUSION

The classification of learning styles is challenging due to the difficulty in handling concept drift, which reduces the generalization ability of the network. In this manuscript, LSTM is incorporated with a meta-learning method to address concept drift and enhance the generalization ability of the network. Initially, the VAK learning style dataset is used, which includes three classes: visual, auditory, and kinesthetic. Unwanted words are removed using stopword removal, root words are identified through lemmatization, and categorical features are converted into numerical values using label encoding. Next, the TF-IDF and GloVe embedding techniques are employed in the feature extraction phase to capture meaningful features for class differentiation. Finally, the classes are accurately classified using the developed MAML-LSTM model, which demonstrates high generalization capability. Subsequently, LIME is applied after the MAML-LSTM classification to highlight key features, ensuring greater transparency and interpretability. The proposed MAML-LSTM achieves 97.77% accuracy, 97.72% precision, 97.72% recall, 97.72% F1-score, 97.72% specificity, and 99.81% AUC on the VAK learning style dataset. Future work will focus on developing different deep learning-based algorithms to further enhance model performance.

Notation table

Notations	Descriptions
X	Cooccurrence Matrix
X_{ij}	Count of Times Words that Existed in Single Window
V_i and V_j	Word Vector of i and j Words
b_i and b_j	Deviation Term
N	Dimension of Cooccurrence Matrix
f	Weight Function
$i(g), f(g)$ and $o(g)$	Input, Forget and Output Gate
σ	Sigmoid Activation Function
b_i, b_f and b_o	Bias Function
w_i, w_f and w_o	Weight Function

y_{r-1}	Hidden State
h_r	Input State
$\mathcal{L}(f,g,\pi_x)$ and $\Omega(g)$	Interpretability And Local Fidelity
$\pi_x(z) = e^{\frac{-D(x,z)^2}{\sigma^2}}$	Certain Distance Function

REFERENCES

[1] L.A. Shoaib, S.H. Safii, N. Idris, R. Hussin, and M.A.H. Sazali, "Utilizing decision tree machine learning model to map dental students' preferred learning styles with suitable instructional strategies", *BMC Medical Education,* Vol. 24, No. 1, p. 58.

[2] B. Bazán-Perkins and J.A. Santibañez-Salgado, "Relationship between the learning gains and learning style preferences among students from the school of medicine and health sciences", *BMC Medical Education,* Vol. 25, No. 1, 2025, pp. 71.

[3] C.W. Teoh, S.B. Ho, K.S. Dollmat, and C.H. Tan, "Machine Learning Prediction Model for Early Student Academic Performance Evaluation in Video-Based Learning", *International Journal,* Vol. 10, No. 2, 2023, pp. 1529-1544.

[4] A.B. Rashid, R.R.R. Ikram, Y. Thamilarasan, L. Salahuddin, N.F. Abd Yusof, and Z.B. Rashid, "A student learning style auto-detection model in a learning management system", *Engineering, Technology & Applied Science Research,* Vol. 13, No. 3, 2023, pp. 11000-11005.

[5] R. Nazempour and H. Darabi, "Personalized learning in virtual learning environments using students' behavior analysis", *Education Sciences,* Vol. 13, No. 5, 2023, p. 457.

[6] V. Sugumaran and S.J.A. Ibrahim, "Rough set based on least dissimilarity normalized index for handling uncertainty during E-learners learning pattern recognition", *International Journal of Intelligent Networks,* Vol. 3, 2022, pp. 133-137.

[7] A. Agarwal, D.S. Mishra, and S.V. Kolekar, "Knowledge-based recommendation system using semantic web rules based on Learning styles for MOOCs", *Cogent Engineering,* Vol. 9, No. 1, 2022, p. 2022568.

[8] Y. Xu, Q. Ni, S. Liu, Y. Mi, Y. Yu, and Y. Hao, "Learning style integrated deep reinforcement learning framework for programming problem recommendation in online judge system", *International Journal of Computational Intelligence Systems,* Vol. 15, No. 1, 2022, p. 114.

[9] I.M. Al-Shdaifat, L.M. Obeidat, S.N. Mabdeh, L. Alzoubi, and S.H. Al-Khazaleh, "Integrating video feedback into architectural design education to engage diverse learning styles", *Cogent Engineering,* Vol. 10, No. 2, 2023, p. 2269651.

[10] W.S. Sayed, A.M. Noeman, A. Abdellatif, M. Abdelrazek, M.G. Badawy, A. Hamed, and S. El-Tantawy, "AI-based adaptive personalized content presentation and exercises navigation for an effective and engaging E-learning platform", *Multimedia Tools and Applications,* Vol. 82, No. 3, 2023, pp. 3303-3333.

[11] D. Chinnapun and U. Narkkul, "Enhancing learning in medical biochemistry by teaching based on VARK learning style for medical students", *Advances in Medical Education and Practice,* Vol. 15, 2024, pp. 895-902.

[12] H. Banaruee, D. Farsani, and O. Khatin-Zadeh, "EFL learners' learning styles and their reading performance", *Discover Psychology,* Vol. 2, No. 1, 2022, p. 45.

[13] H. Yan, H. Zhang, S. Su, J.F. Lam, and X. Wei, "Exploring the online gamified learning intentions of college students: A technology-learning behavior acceptance model", *Applied Sciences,* Vol. 12, No. 24, 2022, p. 12966.

[14] D. Hashem, "Preferred learning styles of dental students in Madinah, Saudi Arabia: Bridging the gender gap", *Advances in Medical Education and Practice,* Vol. 13, 2022, pp. 275-282.

[15] S. Sabarun, U. Widiati, N. Suryanti, and H. Hajimia, "Exploring the Effectiveness of Graphic Organizers on EFL Learners' Writing Performance Across Different Learning Style Preference and Gender at Higher Education", *Journal of Higher Education Theory and Practice,* Vol. 23, No. 16, 2023, pp. 60-73.

[16] A.R. Sayed, M.H. Khafagy, M. Ali, and M.H. Mohamed, "Predict student learning styles and suitable assessment methods using click stream", *Egyptian Informatics Journal,* Vol. 26, 2024, p.100469.

[17] M.K.H. Kanchon, M. Sadman, K.F. Nabila, R. Tarannum, and R. Khan, "Enhancing personalized learning: AI-driven identification of learning styles and content modification strategies", *International Journal of Cognitive Computing in Engineering,* Vol. 5, 2024, pp. 269-278.

[18] W. Villegas-Ch, J. García-Ortiz, and S. Sánchez-Viteri, "Personalization of learning: Machine learning models for adapting educational content

to individual learning styles", *IEEE Access*, Vol. 12, 2024, pp. 121114-121130.

[19] A.R. Sayed, M.H. Khafagy, M. Ali, and M.H. Mohamed, "Exploring the VAK model to predict student learning styles based on learning activity", *Intelligent Systems with Applications,* Vol. 25, 2025, p. 200483.

[20] B.A. Muhammad, C. Qi, Z. Wu, and H.K. Ahmad, "GRL-LS: A learning style detection in online education using graph representation learning", *Expert Systems with Applications,* Vol. 201, 2022, p. 117138.

[21] Dataset: https://www.kaggle.com/datasets/zeyadkhalid/learning-style-vak

[22] S. Gite, S. Patil, D. Dharrao, M. Yadav, S. Basak, A. Rajendran, and K. Kotecha, "Textual feature extraction using ant colony optimization for hate speech classification", *Big data and cognitive computing,* Vol. 7, No. 1, 2023, p. 45.

[23] M.M. Danyal, S.S. Khan, M. Khan, S. Ullah, M.B. Ghaffar, and W. Khan, "Sentiment analysis of movie reviews based on NB approaches using TF–IDF and count vectorizer", *Social network analysis and mining,* Vol. 14, No. 1, 2024, p. 87.