Beyond Clicks: Modeling User Behavior to Predict Student Outcomes and Guide Learning

Võ Hoàng An[1,2,3]

[1] Faculty of Information Science and Engineering .
[2] University of Information Technology, VNU-HCM, Vietnam.
[3] 21520555@gm.uit.edu.vn.

Abstract

In the field of Educational Data Mining (EDM), the application of clickstream has emerged as a valuable tool for comprehending and monitoring student behavior during examinations. In this research, we delves into the exploration of using clickstream data, which involves tracking mouse clicks, keystrokes, and navigation, to gain insights into student conduct during exams. We have implemented a framework designed for loading, storing, preprocessing, and predicting using clickstream data. Our framework utilizes Streamlit as the front-end, providing a user-friendly interface for analyzing real-time data and predicting whether a student will pass or fail based on their clickstream patterns. In this binary classification, our experiments show that BiLSTM achieved the highest performance with accuracy score of 0.7146, AUC score of 0.6654 and macro F1 score of 0.6600.

Keywords: Clickstream, Time-series, Deep Learning, Big Data

1 Introduction

The Educational Data Mining (EDM) as research domain was well defined in [1] as "the area of scientific inquiry centered around the development of methods for making discoveries within the unique types of data that come from educational settings, and using those methods to better understand students and the settings which they learn in."

However, traditional static datasets have their limitations. They fall short in capturing the dynamic nature of learning processes, especially when it comes to understanding how students engage with educational content over time. Recognizing this, researchers are now focusing on time series analysis [2], a tool that allows us to examine patterns and trends in educational data as they evolve.

One key area gaining attention in EDM is the use of clickstream data in education. Clickstream data records the sequence of a user's actions within a digital environment, offering a detailed view of students' online behaviors. This data, covering everything from accessing learning materials to submitting assignments, is a valuable source for in-depth analysis and prediction. By delving into clickstream data, researchers aim to uncover the complexities of student learning experiences, providing insights that can inform effective teaching strategies, identify areas for improvement, and support the development of personalized learning approaches.

This paper focuses on exploring the analysis and prediction potential offered by educational clickstream data, specifically utilizing the NAEP Competition dataset. By providing a framework for loading, storing, preprocessing and predicting using big data tools and deep learning models, we aim to deepen our understanding of how students learn, ultimately paving the way for more informed educational practices and interventions.

The diagram presented in figure 1 provides an overview of our research. Initially, we collect data by tracking student's behaviors during exams, which is then transmitted for preprocessing through the clickstream framework (refer to 4.1 for more details). Subsequently, the preprocessed data is ready for use in various downstream tasks. In this paper, we concentrate on two tasks: examining user behaviors and a binary classification task to predict if a student will pass or fail based on their clickstream patterns.

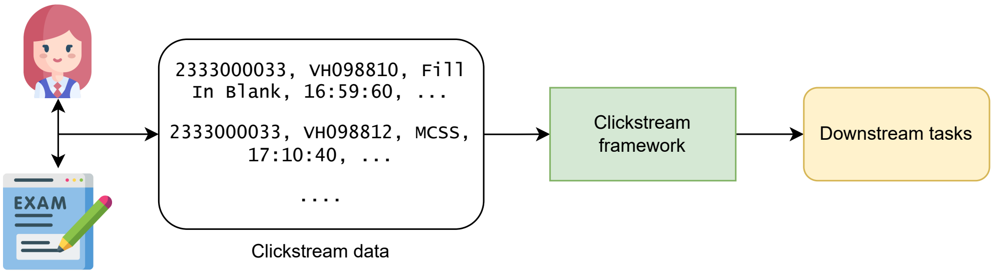
EXAM
2333000033, VH098810, Fill In Blank, 16:59:60, ...
2333000033, VH098812, MCSS, 17:10:40, ...
....
Clickstream data
Clickstream framework
Downstream tasks

Fig. 1: Pipeline

Our contributions can be summarized as follows:

1. We provide an overview of the NAEP Competition dataset by introducing and analyzing it after preprocessing, aiming to reveal valuable insights.

2. We present a framework designed for efficiently loading, storing, preprocessing, and predicting real-time tracking of student behavior during examinations.

3. We perform experiments on classifier models, thoroughly analyze the experimental results, and discuss their implications.

2 Related Works

2.1 Datasets

Educational Process Mining Dataset (EPM) is described in [3]. The dataset has been created by logging performed activities of the students while using an educational simulator. In Harvard Dataverse, the HarvardX Person-Course Academic Year 2013 De-Identified dataset has been described in [4]. The 338,223 logged instances are defined by 20 attributes and were used for understanding the progression of users, Examining access and usage patterns or predicting MOOC performance with week 1 behavior. The EdNet: A Large-Scale Hierarchical Dataset in Education has been described in [5]. The datasets have a different approach offering the logged actions rather than a full set of features describing an instance, and it is more feasible for deep learning algorithms. The Lix Puzzle-game Data Set [6] contains 15.csv file. Each file contains 11 features describing in detail the actions performed during gameplay. Additionally, according to the survey discussed in [7], it was observed that out of the 44 datasets mentioned by the authors, only 6 were related to clickstream data. This indicates a relatively limited availability of clickstream datasets, emphasizing the need for more contributions in this area in the future.

2.2 Existing Approach for Clickstream

The ability to handle and process continuous data streams is becoming an essential part of building a data-driven organization. As a result, numerous studies have proposed Data Stream Processing Systems (DSPS). [8] explored Hadoop limitations and the lambda architecture's potential in data stream processing. They also compared open-source messaging technologies and DSP Engines like Hadoop Online, S4, Storm, Flume, Spark Streaming, Kafka, Scribe, HStreaming, and Impala, considering architectures, use case support, recovery from failures, and license types. [9] compared Storm, Flink, Spark Streaming, and Samza based on language, stream abstraction, latency, throughput, message processing guarantees, and components. [10] compared stream processing solutions (Storm, Spark Streaming, S4, Amazon Kinesis, IBM Streams) based on framework type, implementation language, application development languages, abstraction, data sources, computation model, persistence, reliability, fault tolerance, latency, and vendor.

In time-series tasks, particularly with clickstream data, the increasing popularity of deep learning, including architectures like recurrent neural networks (RNNs) and transformers like BERT, has spurred the development of advanced models in numerous research papers. [11] have developed a novel deep learning-based methodology that predicts student's in-video quiz performance - the likelihood that a student will be Correct on their First Attempt (CFA) based on their clicking behavior. The technique they used is called clustering-guided meta-learning, which guides the neural network to reflect student behavioral clusters during the optimization process. Their evaluation on real-world datasets shows that the behavioral patterns extracted from this process provide useful learning analytics. [12] propose a representation method based on RP [13] to convert the time series to 2D images with a CNN model for time-series classification. In their research, time series are viewed as different recurring patterns like regular cycles and unpredictable variations, which are common in dynamic systems. The core concept behind using the RP method is to identify the points where certain paths revert to a previous state. To classify the images produced by RP, they employ a combination of two convolution stages and two fully connected layers. Pre-trained models are also leveraged for the excellent benefits they bring. [14] proposes a framework for learning representations from educational process data. It includes pre-training with BERT-type objectives on sequential process data and fine-tuning for downstream tasks. Their results show notable improvement over models trained from scratch, scalable and adaptable to various time-series-related domains.

3 NAEP Dataset

3.1 Introduction & Preprocessing

The dataset comprises student actions recorded during a mathematics examination administered by the Educational Testing Service [15] in the 2016-2017 academic year. The examination is structured into two distinct blocks, namely Block A and Block B, each with a time limit of 30 minutes. The primary objective of this dataset is to evaluate the efficiency of time utilization in Block B, predicated on the students' performance and behavior in Block A. The dataset is bifurcated into a training set and a hidden set, with varying degrees of information available for individual students. This dataset serves as a valuable resource for researchers and competition participants interested in investigating the determinants influencing time management and problem-solving abilities in the context of mathematics.

We processed this dataset, referring to the paper [14], and customized it to serve the analysis and training of our deep learning models. For more details, please refer to that paper.

3.2 Data Analysis

The statistic of dataset is shown in table 1. The term ”action” is computed based on the student's time steps, where each time step corresponds to one action.

The distribution of action counts (refer to figure 2) is skewed to the right with a long tail stretching towards higher action counts. Additionally, we examine the distribution of action counts per question (refer to figure 3) for more detailed insights. These findings have helped us analyze the distribution of the number of time steps for each student, providing valuable information for analyzing user's behaviors. For instance, we can examine the actions of students with good results and identify potential anomalies in this box plot, which may represent cases of cheating. Furthermore, these details enable us to optimize the input for time-series models more effectively. Lastly, We also illustrate the distribution of time taken to complete the exam in figure 4.

Regrading to binary classification task to predict whether a student will pass or fail based on their click logs data, we've examined the distribution of labels in the dataset. Figure 5 shows that the ”Pass” label (True) accounts for about 61%, while the ”Fail”

Table 2: NAEP Dataset Statistics

Statistics	
Number of students	1232
Number of questions	24
Question types	10
Action types	42
Min actions in exam	106
Max actions in exam	2609
Average actions in exam	346

label (False) makes up around 39%. This indicates that the label distribution in our dataset is skewed towards the ”Pass” label.

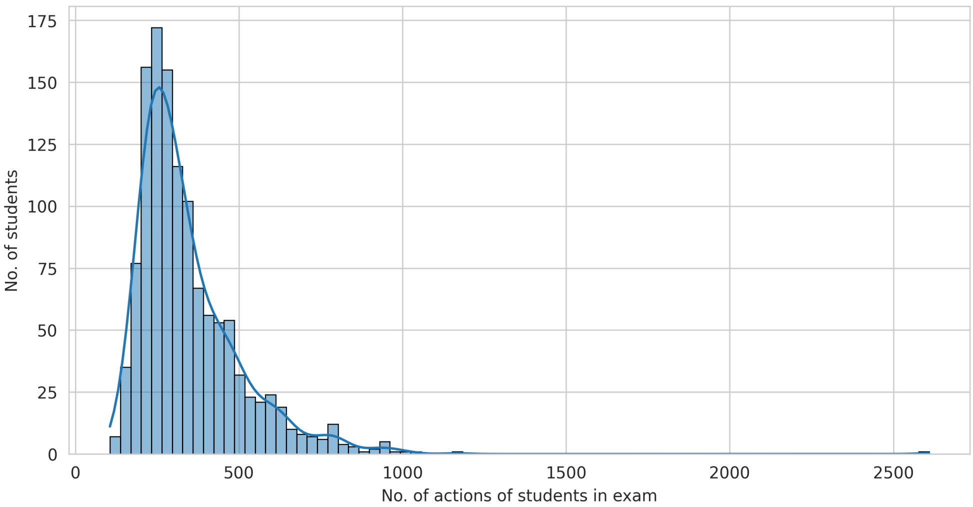

Fig. 2: Distribution of action counts

4 Framework & Models

4.1 Framework

We’ve developed a framework (see figure 6) to handle and analyze clickstream data effectively. It combines strong data management with deep learning techniques for accurate classification.

Firstly, the raw clickstream data gets stored in MongoDB[1] for easy access and long-term storage. We then employ Apache Kafka[2] as a message broker to facilitate

[1] https://www.mongodb.com/
[2] https://kafka.apache.org/

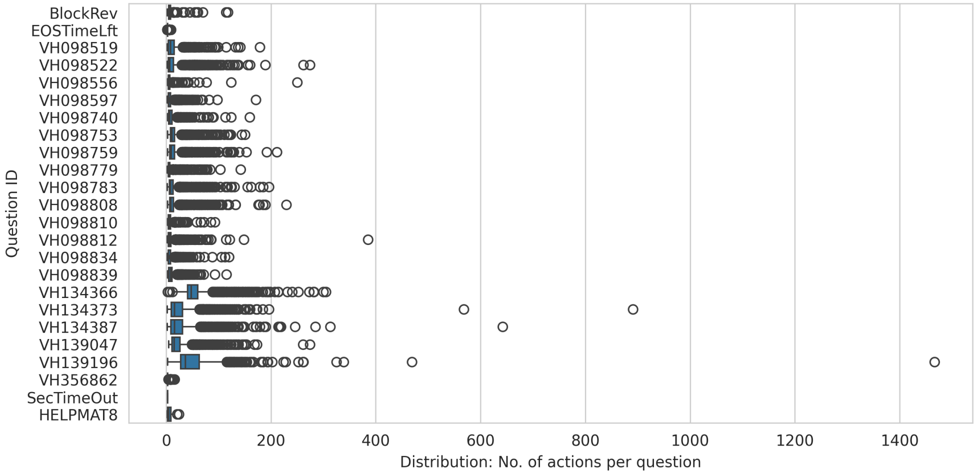

Fig. 3: Distribution of action counts per questions

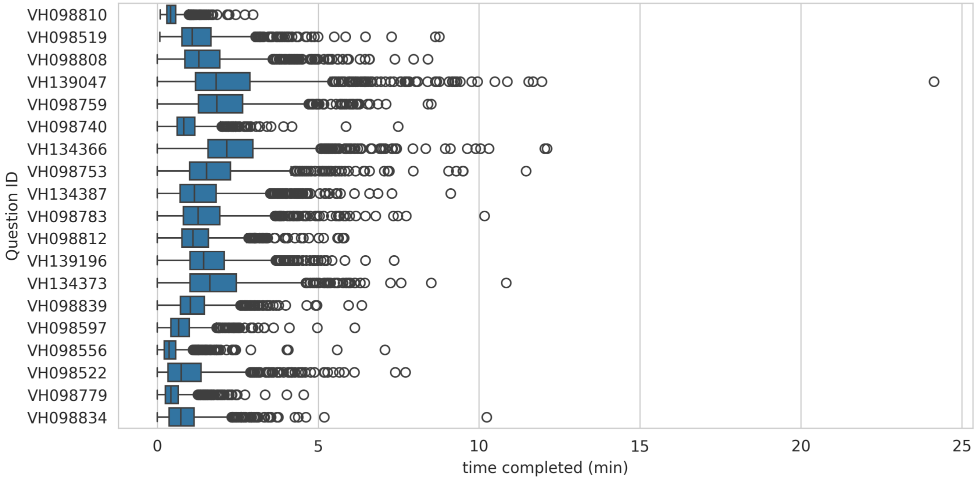

Fig. 4: Duration of exam completion

real-time data distribution and consumption. We also leveraged Apache Spark[3] to streamline the handling of large-scale data efficiently.

Secondly, at the core of our framework, we implemented a deep learning module for classification. This component extracts meaningful insights from the processed clickstream data, enhancing the accuracy of our results.

Finally, to provide a user-friendly interface for exploring and analyzing the outcomes, we have developed a Streamlit[4] dashboard. This interactive dashboard allows

[3] https://spark.apache.org/

[4] https://streamlit.io/

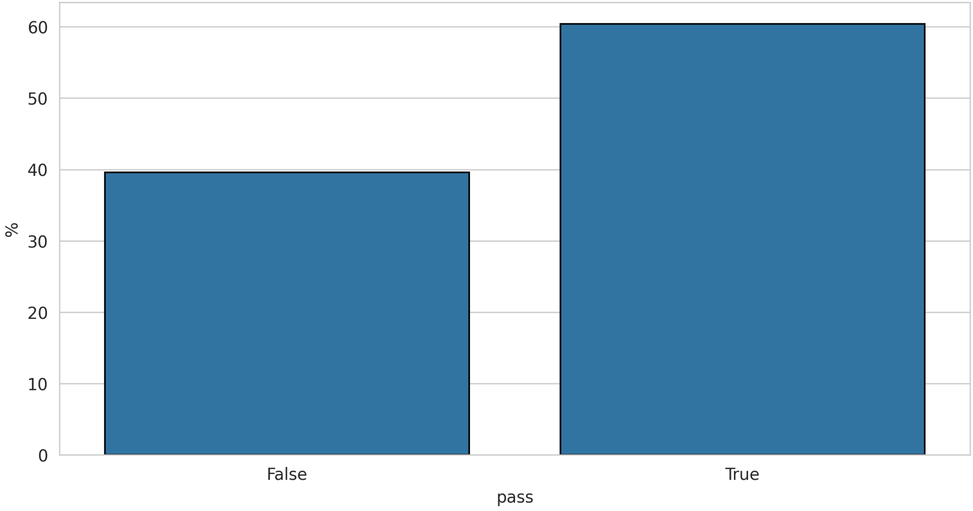

Fig. 5: Distribution of labels in NAEP Dataset

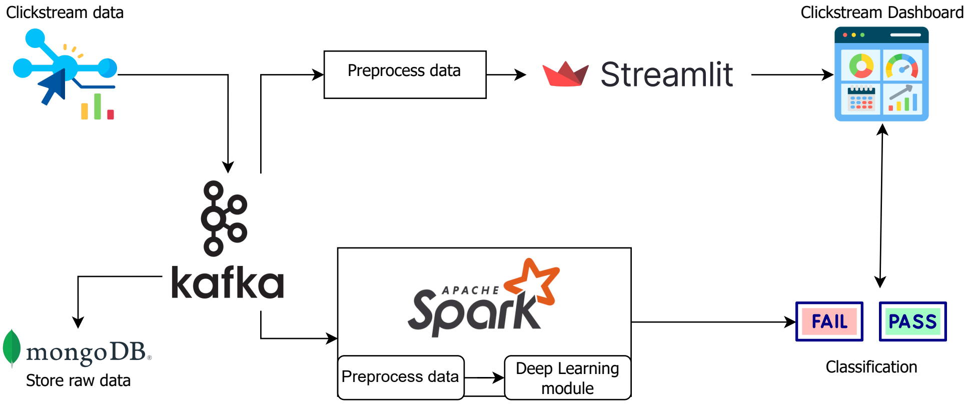

Fig. 6: Framework for loading, storing, preprocesisng, analyzing and predicting clickstream Data

users to visualize the analysis of clickstream data and the results of the classification process, contributing to a more accessible and insightful data exploration experience.

4.2 Models

For binary classification, we have evaluated two variants of Recurrent Neural Networks: Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU), both with and without bidirectionality, as well as the BERT model. Further details provided below:

Long Short-Term Memory (LSTM) [16]: LSTM is a type of recurrent neural network (RNN) architecture designed to address the vanishing gradient problem in traditional RNNs. It introduces memory cells with gating mechanisms to selectively retain and forget information, enabling better learning of long-range dependencies.

Gated Recurrent Unit (GRU) [17]: Similar to LSTM, GRU is another variant of RNN designed to overcome the vanishing gradient problem. It has a simpler architecture with fewer parameters than LSTM, utilizing a gating mechanism to control the flow of information through the network.

Bidirectional Long Short-Term Memory (BiLSTM) [18]: BiLSTM extends LSTM by processing input sequences in both forward and backward directions. This bidirectional approach allows the model to capture contextual information from both past and future inputs, enhancing its ability to understand and predict sequential patterns.

Bidirectional Gated Recurrent Unit (BiGRU) [19]: Similar to BiLSTM, BiGRU processes input sequences in both directions. It combines the advantages of bidirectionality with the simplicity of the GRU architecture, providing an effective solution for capturing contextual information in sequential data.

Bidirectional Encoder Representations from Transformers (BERT) [20]: Unlike traditional RNNs, BERT relies on a transformer-based architecture. It doesn't follow the sequential processing approach of RNNs but employs positional encoding layers to handle time-series data like clickstream events. BERT's bidirectional attention mechanism enables it to capture contextual information efficiently from both past and future events in the clickstream.

5 Experiments and Results

5.1 Evaluation Metrics

To evaluate the performance of classifiers, we chosen accuracy score, macro F1 score and AUC score. More details about these metrics are explained below:

Accuracy score [21]: Measures overall correctness by calculating the ratio of correctly predicted instances to the total number of instances.

$$\text{Accuracy} \;=\frac{(TP+TN)}{(TP+FP+TN+FN)} \tag{1}$$

Macro F1 score [21]: A variant of the F1 score, it combines precision and recall to assess performance across all classes, giving equal weight to each class.

$$\text{F1 score} \;=\frac{\mathrm{TP}}{\mathrm{TP}+\frac{1}{2}(\mathrm{FP}+\mathrm{FN})} \tag{2}$$

$$\text{Macro F1 Score} \;=\frac{\sum_{i=1}^{n}\text{ F1 Score }_{i}}{n} \tag{3}$$

AUC score [21]: stands for "Area under the ROC Curve". AUC quantifies a binary classifier's ability, ranging from 0 to 1, with 1 indicating perfect predictions

Table 4: Classification model performance metrics at a threshold of 0.5

	Accuracy	AUC	Macro F1 score
BiLSTM	**0.7146**	0.6654	**0.6600**
BiGRU	0.7073	**0.6850**	0.6430
LSTM	0.6634	0.6378	0.5762
GRU	0.6341	0.6418	0.4661
BERT	0.6049	0.5716	0.3769

and 0.5 for random predictions. A higher AUC score signifies better discrimination ability across different thresholds.

5.2 Results & Analysis

We report the main results in table 3 and figure 7

We assessed the performance of various recurrent neural network (RNN) variants and the transformer-based model BERT. Our findings indicate that the BiLSTM model achieved the best results with an accuracy score of 0.7146 and a macro F1 score of 0.6600. For the AUC score, BiGRU yielded the highest result of 0.6850.

We also examined the confusion matrix for each model. Overall, BiLSTM emerged as our best-performing model across all three metrics with a threshold of 0.5. While BiGRU showed a high AUC score, its lower F1 score suggests that the 0.5 threshold may not be suitable for all models, highlighting the need for fine-tuning. BERT, a recent and popular model utilizing position embedding instead of layers like RNN, exhibited overfitting on our imbalanced labeled dataset and produced the lowest results, possibly due to the limited dataset size.

6 Conclusion and Future Work

In this paper, we have developed a framework using Streamlit as the front-end to analyze students' click logs in real-time during exams and predict whether they pass or fail. We have also presented valuable insights from the analysis of NAEP data. In binary classification, we found that BiLSTM is our best-performing model, while BERT yields the lowest results, likely due to insufficient data.

Examining the results from the confusion matrix, we observed a bias in our models towards the "Pass" label. This is because we maintained the label distribution of the dataset during training. In the future, we plan to experiment with label balancing methods on the dataset to enhance the predictive capability of the model. Additionally, we identified the potential to leverage embedding layers in the models for representing user behavior. Through clustering algorithms, we can better understand hidden patterns in how users behave, enabling us to create personalized strategies for specific groups of students.

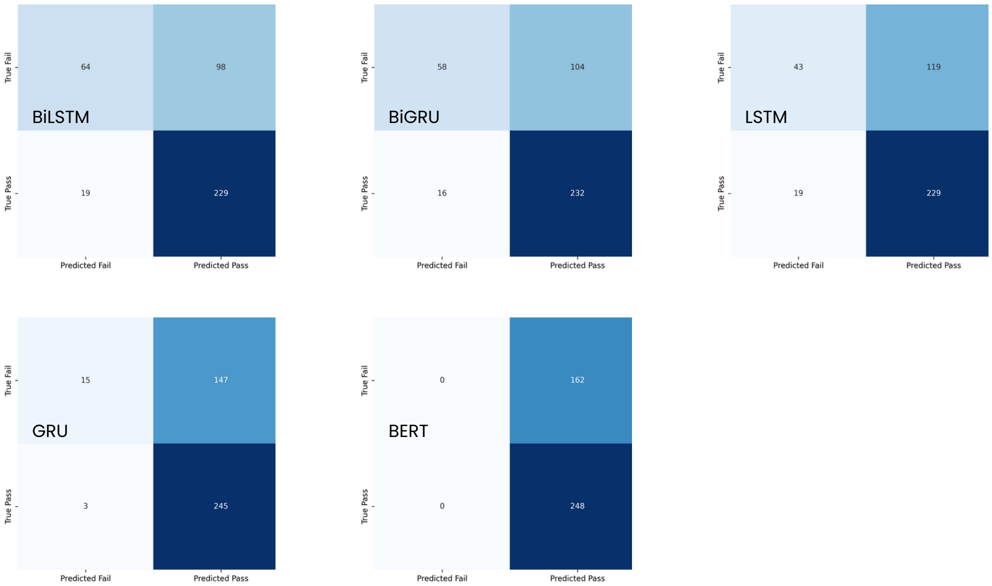

Fig. 7: Confusion matrix

Limitations

Despite our best efforts to enhance our classifier models, it's important to acknowledge certain limitations. One notable concern is the insufficient amount of data. Our dataset is not extensive, whereas deep learning models typically demand a substantial volume of data. Additionally, there's currently a shortage of pre-trained models specifically designed for clickstream tasks, and these models often work best in a narrow domain, mainly in e-commerce. The lack of large datasets and pre-trained models can increase the training costs and affect the accuracy of the model. Overcoming these constraints requires dedicated research time and testing efforts to create better datasets and pre-trained models.

Acknowledgments

Thank you to Ph.D. Do Trong Hop, the direct supervisor and lecturer, who has provided enthusiastic support to the group throughout the learning process and the completion of this thesis.

References

[1] Algarni, A.: Data mining in education. International Journal of Advanced Computer Science and Applications **7** (2016) https://doi.org/10.14569/IJACSA.2016.070659

[2] Almeida, A., Brás, S., Sargento, S., Pinto, F.C.: Time series big data: a survey on data stream frameworks, analysis and algorithms. J. Big Data **10**(1), 83 (2023)

[3] Vahdat, M., Oneto, L., Anguita, D., Funk, M., Rauterberg, M.: A learning analytics approach to correlate the academic achievements of students with interaction data from an educational simulator. In: Conole, G., Klobucar, T., Rensing, C., Konert, J., Lavoue, E. (eds.) Design for Teaching and Learning in a Networked World : 10th European Conference on Technology Enhanced Learning, EC-TEL 2015, Toledo, Spain, September 15–18, 2015 : Proceedings. LNCS, pp. 352–366. Springer, Germany (2015). https://doi.org/10.1007/978-3-319-24258-3_26 . 10th European Conference on Technology Enhanced Learning, EC-TEL 2015, EC-TEL 2015 ; Conference date: 15-09-2015 Through 18-09-2015. http://ectel2015.httc.de/index.php?id=704

[4] HarvardX: HarvardX Person-Course Academic Year 2013 De-Identified Dataset, Version 3.0. https://doi.org/10.7910/DVN/26147 . https://doi.org/10.7910/DVN/26147

[5] Choi, Y., Lee, Y., Shin, D., Cho, J., Park, S., Lee, S., Baek, J., Bae, C., Kim, B., Heo, J.: EdNet: A Large-Scale Hierarchical Dataset in Education (2020)

[6] Li, W., Funk, M., Li, Q., Brombacher, A.C.: Visualizing event sequence game data to understand player's skill growth through behavior complexity. Journal of Visualization (2019) https://doi.org/10.1007/s12650-019-00566-5

[7] Mihaescu, C., Popescu, P.: Review on publicly available datasets for educational data mining. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery **11** (2021) https://doi.org/10.1002/widm.1403

[8] Liu, X., Iftikhar, N., Xie, X.: Survey of real-time processing systems for big data. (2014). https://doi.org/10.1145/2628194.2628251

[9] Hesse, G., Lorenz, M.: Conceptual survey on data stream processing systems, pp. 797–802 (2015). https://doi.org/10.1109/ICPADS.2015.106

[10] Singh, M.P., Hoque, M.A., Tarkoma, S.: A survey of systems for massive stream analytics (2016)

[11] Chu, Y.-W., Tenorio, E., Cruz, L., Douglas, K., Lan, A.S., Brinton, C.G.: Click-Based Student Performance Prediction: A Clustering Guided Meta-Learning Approach (2021)

[12] Debayle, J., Hatami, N., Gavet, Y.: Classification of time-series images using deep convolutional neural networks, p. 23 (2018). https://doi.org/10.1117/12.2309486

[13] Piskun, O., Piskun, S.: Recurrence quantification analysis of financial market crashes and crises (2011)

[14] Scarlatos, A., Brinton, C., Lan, A.: Process-BERT: A framework for representation learning on educational process data. In: Mitrovic, A., Bosch, N. (eds.) Proceedings of the 15th International Conference on Educational Data Mining, pp. 715–719. International Educational Data Mining Society, Durham, United Kingdom (2022). https://doi.org/10.5281/zenodo.6853006

[15] R. Baker B. Woolf I. Kats C. Forsyth J.Ocumpaugh, T.P.N.H.: Nations Report Card Data Mining Competition 2019. https://sites.google.com/view/dataminingcompetition2019/home.

[16] Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997) https://doi.org/10.1162/neco.1997.9.8.1735

[17] Cho, K., Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using RNN encoder–decoder for statistical machine translation. In: Moschitti, A., Pang, B., Daelemans, W. (eds.) Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1724–1734. Association for Computational Linguistics, Doha, Qatar (2014). https://doi.org/10.3115/v1/D14-1179 . https://aclanthology.org/D14-1179

[18] Zhang, S., Zheng, D., Hu, X., Yang, M.: Bidirectional long short-term memory networks for relation classification. In: Zhao, H. (ed.) Proceedings of the 29th Pacific Asia Conference on Language, Information and Computation, Shanghai, China, pp. 73–78 (2015). https://aclanthology.org/Y15-1009

[19] Chung, J., Gülçehre, Ç., Cho, K., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling. CoRR **abs/1412.3555** (2014) 1412.3555

[20] Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. CoRR **abs/1810.04805** (2018) 1810.04805

[21] Sokolova, M., Japkowicz, N., Szpakowicz, S.: Beyond accuracy, f-score and roc: A family of discriminant measures for performance evaluation, vol. Vol. 4304, pp. 1015–1021 (2006). https://doi.org/10.1007/11941439_114