# Using early clickstream data to identify at-risk students in higher education: an LSTM-based approach

**Ricardo Santos** [0000-0003-1007-206X], **Roberto Henriques** [0000-0002-4862-8177]

NOVA Information Management School (NOVA IMS), Campus de Campolide, Universidade Nova de Lisboa, 1070-312 Lisboa, Portugal

**This is the accepted manuscript version of an article published by IATED:**

Santos, R., & Henriques, R. (2024). Using early clickstream data to identify at-risk students in higher education: an LSTM-based approach. In L. G. Chova, C. G. Martínez, & J. Lees (Eds.), 16th International Conference on Education and New Learning Technologies 1-3 July, 2024 Palma, Spain (pp. 6216-6225). (EDULEARN24 Proceedings; No. 2024). IATED Academy. https://doi.org/10.21125/edulearn.2024.1471 Available at the following link: https://library.iated.org/view/SANTOS2024USI

# USING EARLY CLICKSTREAM DATA TO IDENTIFY AT-RISK STUDENTS IN HIGHER EDUCATION: AN LSTM-BASED APPROACH

**R. Santos, R. Henriques**

*NOVA Information Management School (NOVA IMS), Universidade Nova de Lisboa, 1070-312, Lisbon (PORTUGAL)*

## Abstract

Identifying students who require additional support is a challenging task for educators at the higher education level. One of the main reasons for this difficulty is that traditional assessments of student performance, such as a final exam or project, occur at the end of the course when it is often too late to implement corrective measures that could prevent a student from failing. Learning management systems (LMS) help students engage with educational content and provide a continuous flow of data that documents student-content interactions throughout the course. Although the data collected may be incomplete and noisy, it can still provide valuable insights into student behaviour and performance.

This work uses Moodle logs to create course-agnostic Long Short-Term Memory unit (LSTM)-based classifiers for early identification of students at risk of failing a course. Our classifiers used the day-wise sequences of the number of clicks by each student in various activity types as input. By analyzing data collected up to the 50th day of each course, our LSTM-based classifiers achieved an average area under the receiver operating characteristic curve (AUC) of 0.69 while identifying close to 28% of the at-risk students. Furthermore, with minor changes to the model's hyperparameters, we created a classifier that achieved a slightly lower AUC score (0.67) but identified more than 50% of the at-risk students. Moreover, models trained using only the first 25 days of click sequences achieved similar recall scores, even if their overall accuracy and AUC scores were inferior.

These results suggest that our approach can help educators identify struggling students and provide them with timely feedback to prevent avoidable failures. Future research could explore the generalization of this approach to more courses and recognize the contribution of each activity type to the final prediction.

Keywords: Student Performance, Early Prediction, Learning Management Systems, Machine Learning..

## 1 INTRODUCTION

Timely identification of struggling students can be a daunting task for educators in higher education due to the ever-increasing challenges they face in their practice [1]. Issues stemming from increasing class sizes and adopting blended and remote learning solutions were already observable in the 2010s [2], [3]. Still, they became much more evident after the emergency transformations induced by the Covid-19 pandemic [4]. Regardless of the specific cause, educators have a weaker connection to their students, resulting in a shallower understanding of who they are, how they engage with and interpret the materials, and how they perform. Consequently, some students are likely to slip through the cracks.

Learning management systems (LMS) are web applications that support student interactions with educators and educational materials [5]. Students must access the institutional LMS page and navigate through its menus and features to access its contents. They manifest behaviour that may encode relevant information about their motivation and attitude towards the course and learning more broadly. Modern LMS keep timestamped logs of every user-system interaction. The potential of this clickstream data [6] as an early predictor of course performance is an active research interest in learning analytics [7], [8].

Traditional approaches to using clickstream data to identify at-risk students involve extracting potentially meaningful features from the logs and converting them into a flattened multivariate representation [6]. However, the growing popularity of deep learning algorithms, such as recurrent neural networks (RNN) [9] or long short-term memory units (LSTM) [10], [11], has contributed towards their adoption in recent years with promising results [8], [12]. These algorithms are especially suited for handling sequences and capturing temporal information and patterns that would otherwise be lost using the flattened representation. Yet, the field is still in its infancy, with a comparatively low number of works using these approaches. Moreover, to our knowledge, most larger-scale works published on the early identification of at-risk students using RNN or LSTM use the Open University Learning Analytics dataset (OULAD) [13].

In this work, we present our results for an LSTM-based early detector of at-risk students trained on a multivariate temporal representation of the number of clicks performed per day. In doing so, we intend to address the following research questions:

1. Can the sequences of daily clicks in distinct categories be used for course-agnostic early identification of at-risk students?
2. What is the proportion of at-risk students identifiable by the LSTM-based model when predicting each course's 25$^{th}$ day? And on the 50$^{th}$ day?

We analyze data from students enrolled in a European information management school during the 2020/2021 academic year. Courses at this school can be either trimestral or semestral. Trimestral courses, on average, last for seven weeks, with the first exam season occurring 56 to 60 days after the start of the course. Semestral courses, on the other hand, last for an average of 14 weeks, with the first exam season occurring more than one hundred days after the start of the course. Our dataset consists of 927 thousand Moodle logs from 9,296 course enrolments, with 1,590 unique students across 138 distinct courses. For each enrolment, we extract daily click data for each of the first 50 days of the course and use this information to create LSTM-based models. These models predict at-risk students after the course's 25th and 50th days, thus allowing trimestral and semestral courses to be included in the same early-prediction models.

The remainder of this work is organized as follows: the next chapter reviews relevant works that use LSTM-based models in the early identification of at-risk students. The third chapter presents our methodology. The fourth and fifth chapters present our results, analyze their implications, and compare them to expected outcomes. The sixth chapter summarises our main findings and outlines potential avenues for future research.

## 2 RELATED WORK

Any successful implementation of early prediction is contingent on the exclusive use of data collected up to the moment when a prediction is meant to be made [14]. When making predictions from LMS logs, the literature is rich in approaches that vary in the number of courses considered, the specific moment of prediction or the algorithms used [7], [12], [14], [15]. There seems to be a trade-off between the models' predictive performance and the time available to provide feedback and meaningful corrective action to students in need, with some space in the middle for compromises.

In an early attempt using a sample of 300 students from a single course, the authors in [14] achieved an accuracy score equal to 0.972 when predicting at-risk students after the 4th out of 13 weeks. More recent works have attempted to obtain early predictions on more than one course, with [15] achieving an accuracy score of 0.67 with Logistic Regression (LT) at the 3$^{rd}$ out of 10 weeks and [16] using Support Vector Machines (SVM) to achieve an F-Score of 0.83 at the halfway point of a 5-week course. A more generalized institution-wide approach can be found in [7], where the authors use data collected from 699 courses and more than 15,000 students to create different predictive models at different stages of course completion (namely 10%, 25%, 33%, and 50%) and achieve an F-score of 0.89 and an area under the receiver operator characteristic curve (AUC) of 0.95 at the courses' halfway point.

In recent years, there has been an increase in approaches that use deep learning algorithms. In [17], the authors showed that RNN or LSTM trained on sequences of weekly clicks tended to outperform other predictive models, such as SVM, on data collected from 234 students attending an online course at a Taiwanese university. In [18], the authors trained an LSTM-based model using sequences of daily clicks that outperformed conventional classifiers trained on multiple features as early as 28 days after the start of the course when discriminating between the top-performing students and the remaining class members. On a larger scale, two studies extracted weekly sequences of clicks on the LMS on 20 activities and used them to train LSTM-based models that outperformed other conventional machine learning classifiers in identifying at-risk students at multiple stages of course completion on the OULAD dataset, which included 32,593 unique students enrolled in seven courses across 38 weeks [8], [12]. By the 10th week, the classifiers achieved accuracies of 0.85 in [12] and 0.73 and an AUC of 0.74 in [8].

Although these studies have shown promising results, they are often limited to specific courses or smaller sample sizes. This paper introduces a novel approach that uses LSTM-based models trained on daily sequences of clicks in a larger-scale, course-agnostic early warning system.

## 3 METHODOLOGY

Figure 1 presents a summary of the overall experimental design. We divided our work into two main sections. In the first section, we filtered Moodle's activity logs for the initial 50 days of each course and consolidated them into a tensor with 9296 x 14 x 50 dimensions. These dimensions represent the number of student enrolments, the number of clicks per activity type for any given day, and the number of days considered.

In the second stage, we used PyTorch [19] to create and train two distinct LSTM-based classifiers that differed in the number of days they considered as input. The first classifier (Model 1) considers the clicks per activity type for the initial 25 days, while the second classifier (Model 2) considers the entire sequence of 50 days. We provide a more detailed description of each stage in the following subsections.

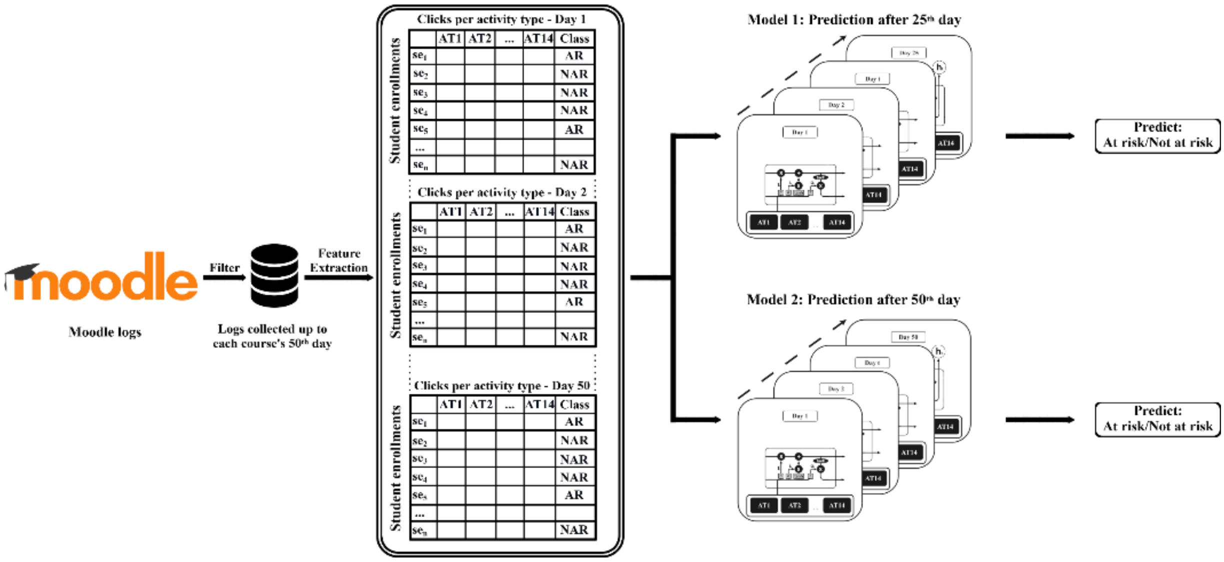


*Figure 1. Sequential overview of the experimental design followed in this work.*

### 3.1 Data description

This work uses the 926,738 Moodle logs recorded by a sample of 1,590 unique students during the first 50 days of 138 courses offered by a European information management school during 2020/2021. The courses from the sample belong to the undergraduate, graduate, and postgraduate levels. Table 1 overviews the number of courses, unique students, enrolments, and at-risk students for each program level. Of the 138 courses, twenty-three were trimestral with an average duration of seven weeks (49 days), while the remaining 115 courses were semestral with an average duration of 14 weeks (98 days). In both types of courses, the final evaluation occurred between 2 to 6 weeks after the end of the course.

To ensure student privacy, all data was anonymized in compliance with the General Data Protection Regulation (GDPR). Moreover, the project was approved by the Ethics Committee and Institutional Review Board with Code DSCI2022-9-227363.

*Table 1. Number of courses, students, student enrollments and average end-of-course performance per undergraduate program*

| Program level | Courses | Students | Student enrolments |
|---|---|---|---|
| Undergraduate | 55 | 409 | 3387 |
| Master's | 62 | 872 | 5013 |
| Postgraduate | 21 | 325 | 896 |

| Total | 138 | 1606 | 9296 (*1872 at-risk*) |
|---|---|---|---|

## 3.2 Data preprocessing

We categorized the clicks from the sample according to fourteen distinct activity types: course, resource, forum, URL, folder, quiz, grades, assignments, groups, user, submission, page, choice, and other. We aggregated the clicks per activity type and day for each student enrolment, thus resulting in a tensor with dimensions 9296 x 14 x 50.

We were given access to each enrolment's end-of-course grade to determine the outcome variable. Enrolments achieving an end-of-course grade equal to or below 11 (out of 20) were labelled at-risk, while the remaining were not. Under this formulation, 1872 enrolments (20% of the total) were considered as *at-risk*.

## 3.3 Data Analysis

### 3.3.1 Overview of LSTM networks

LSTMs are RNN architectures that handle long sequences [10], [11]. As depicted in Figure 2, LSTMs feature an internal cell state C that can be thought of as the cell's long-term memory and a hidden state h that can be considered the cells' short-term memory. At every new timestep $t$, perturbations to $C$ are regulated by three multiplicative gates: a forget gate $(f_t)$ that filters information inherited from previous timesteps, an input gate $(i_t)$ that softens the effects of irrelevant inputs at the current timestep and an output gate $(o_t)$ that shields future units from irrelevant short-term memories of the current cell.

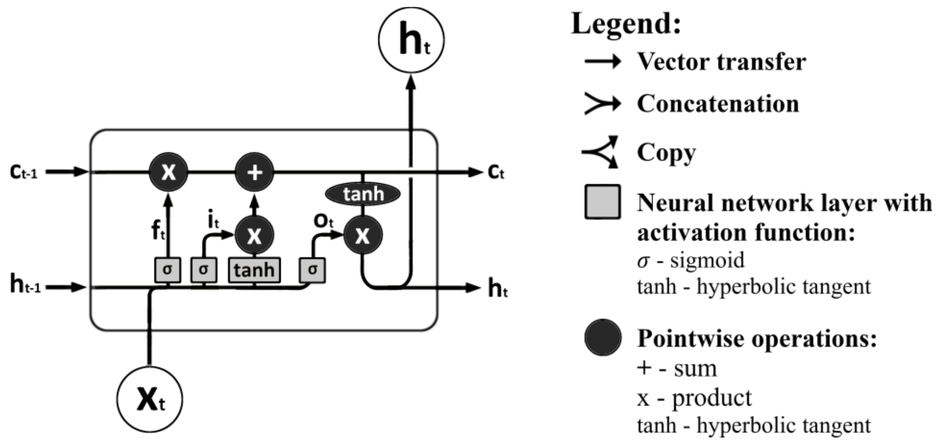


*Figure 2. Schematic representation of the internal architecture of an LSTM*

Equations 1 to 6 describe the inner workings of an LSTM. Let $W_g$ and $b_g$ represent the weights and biases of a gate (where $g$ can either refer to $f$, $i$ or $o$). At timestep $t$, the unit inherits the cell state $C_{t-1}$ and hidden state $h_{t-1}$ from the preceding time step. Moreover, the unit also takes a vector $x_t$ as an input. The forget gate uses Equation (1) to decide what information should be retained and what should be discarded:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \qquad (1)$$

In the second stage, the input gate uses Equation (2) to assign weights to the new information that will be used to update $C_t$. These weights will be multiplied by the vector of candidate values $\xi_t$ described by equation (3):

$$i_t = \sigma(W_i \cdot [ht_{-1}, x_t] + b_i) \qquad (2)$$

$$\xi_t = \tanh(W_\xi \cdot [h_{t-1}, x_t] + b_\xi) \qquad (3)$$

The new cell state $C_t$ is determined by Equation (4), which represents the element-wise sum of the Hadamard products between $f_t$ and preceding cell state $C_{t-1}$, and it with candidate vector $\xi_t$.

$$C_t = f_t C_{t-1} + i_t \xi_t \qquad (4)$$

The final steps involve determining which elements of $C_t$ should be allowed to influence future units. This is determined by the output gate ot described in Equation (5). Finally, the Hamard product between ot and the current cell state, as defined by equation (6), results in the new hidden state ht:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \qquad (5)$$

$$h_t = o_t \tanh(C_t) \qquad (6)$$

$C_t$ and $h_t$ will be used as inputs for the next timestep, and the cycle will be repeated.

### 3.3.2 Modelling approach

In this section, we created LSTM-based models to identify at-risk students from clickstream data. We made several models that took day-wise sequences of clicks for each of the fourteen activity types, as described in the previous sections, for every student enrolment. All models shared a general architecture, as illustrated in Figure 3, with a single LSTM layer. Both the initial hidden ($h_0$) and cell state ($c_0$) were initialized using the Glorot normal initialization [20]. At each day $t$, the models inherited the hidden state ($h_{t-1}$) and cell state ($c_{t-1}$) from the previous day and received as input a vector ($x_t$) featuring the number of clicks per activity type made on that day. The hidden layer from the final day of the sequence was passed through a dropout layer and sent to a dense layer with a sigmoid activation function to generate a prediction $y$ (at-risk or not). During training, the loss was computed using binary cross-entropy. We created two models intended for different prediction moments: Model 1 used sequences of the first 25 days of each course, and Model 2 used sequences of clicks for the first 50 days.

We performed hyper-parameter tuning for each model by measuring the average AUC of each combination across thirty repetitions of 10-fold stratified cross-validation. Hyperparameters were selected through a random search [21] over the following parameters: batch size, size of the hidden state, dropout, number of epochs, the initial learning rate and optimizer. Moreover, due to the imbalanced nature of the dataset, all modelling attempts used data that was oversampled with the synthetic minority oversampling technique (SMOTE) [22].

The combinations of hyperparameters that achieved the highest AUC score on the cross-validation stage were selected to train a set of final models. We evaluated the average performance of the final models as an average across a 30-split repeated holdout that used 75% of the data for training and 25% for testing.

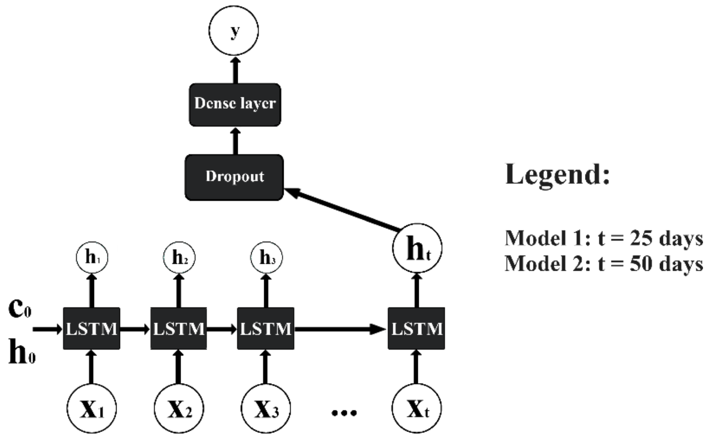


*Figure 3. t-step implementation of the LSTM-based models we created to identify at-risk students*

### 3.3.3 Performance Metrics

We evaluated the selection of the best set of hyperparameters using the AUC score, which ranges between 0 and 1. We also assessed the final models using accuracy and recall scores, defined by equations (7) and (8), respectively.

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \qquad (7)$$

$$Recall = \frac{TP}{TP+FN} \qquad (8)$$

## 4 RESULTS

In this work, we developed and trained two LSTM-based models to predict performance in online courses. Our models used day-wise click sequences in fourteen activity types collected from the first 50 days of each course in our sample. Model 1 predicts performance after the 25$^{th}$ day, while Model 2 uses sequences for the first 50 days. We performed hyper-parameter tuning using a random search and evaluated the results using the average AUC score from thirty repetitions of stratified 10-fold cross-validation.

The results reported herein refer to the ones obtained by the three best combinations of hyperparameters for each model. Table 2 highlights these hyper-parameter combinations, with Model 1A, 1B, and 1C representing the different combinations for Model 1, with a similar naming convention adopted for Model 2. We conducted these experiments on a machine with 6 GB GPU and 32 GB of RAM.

*Table 2. Best combinations of hyperparameters for each LSTM-based model – as measured by AUC during cross-validation*

| **Parameter** | **Model 1A** | **Model 1B** | **Model 1C** | **Model 2A** | **Model 2B** | **Model 2C** |
|---|---|---|---|---|---|---|
| Hidden state size | 128 | 256 | 128 | 128 | 64 | 128 |
| Batch size | 128 | 512 | 32 | 128 | 128 | 32 |
| Number of epochs | 100 | 200 | 200 | 100 | 50 | 200 |
| Optimizer | Adam | Adam | Adagrad | Adam | Adam | Adagrad |
| Dropout | 0.5 | 0.5 | 0.2 | 0.5 | 0.2 | 0.2 |
| Learning rate | 0.001 | 0.01 | 0.01 | 0.001 | 0.005 | 0.01 |

Table 3 presents the average performances of our best combinations of hyperparameters on the repeated holdout stage. The table highlights the best models in bold for each model and performance metric. Multiple scores are highlighted when the differences between the two algorithms are not statistically significant (p-value<0.05 for Wilcoxon signed-rank test).

We found that Model 1 achieved AUC scores between 0.629 (Model 1C) and 0.649 (Model 1A), while Model 2's AUC scores ranged between 0.671 (Model 2C) and 0.692 (Model 2A). Model 1A also achieved the highest accuracy score when predicting after the 25th day, whereas Model 2B had the highest overall accuracy out of all models tested. It is worth noting that despite achieving the lowest AUC and accuracy scores, models 1C and 2C identified, on average, more than 50% of the at-risk students, as shown by the recall scores of 0.581 and 0.537, respectively.

*Table 3. Classification results for identification of at-risk students using AUC, Accuracy and Recall scores. Performances are averaged across a 30-split repeated holdout*

| Moment of prediction | Model | AUC | Accuracy | Recall |
|---|---|---|---|---|
| After 25 days | Model 1A | **0.649 +/- 0.012** | **0.739 +/- 0.019** | 0.276 +/- 0.062 |
| | Model 1B | 0.642 +/- 0.010 | 0.727 +/- 0.013 | 0.276 +/- 0.052 |
| | Model 1C | 0.629 +/- 0.020 | 0.600 +/- 0.078 | **0.581+/- 0.129** |
| After 50 days | Model 2A | **0.692 +/- 0.017** | 0.762 +/- 0.024 | 0.286 +/- 0.089 |
| | Model 2B | 0.680 +/- 0.028 | **0.767 +/- 0.062** | 0.237 +/- 0.110 |
| | Model 2C | 0.671 +/- 0.028 | 0.627 +/- 0.148 | **0.537 +/- 0.251** |

## 5 DISCUSSION

### 5.1 Early identification of at-risk students using LMS click sequences

The ability to distinguish between classes in binary classification is commonly measured using the AUC score, with a score of 0.7 or greater considered a benchmark for acceptable performance [23]. Unfortunately, neither the different combinations of hyperparameters for Model 1 (prediction after the 25$^{th}$ day) nor Model 2 (prediction after the 50$^{th}$ day) achieved this level of performance on average. However, Model 2 came remarkably close to this threshold, as demonstrated by Model 2A's average AUC of 0.694. These results suggest predictive potential in the clickstream data up to the end of the 50$^{th}$ day of each course.

Educators often cannot determine whether a specific student is at risk of failing a course, especially in the initial stages. Considering this, a potentially helpful benchmark for early identification of at-risk students is the expected performance of two hypothetical naïve classifiers: one that labels all students as at-risk and another that labels no students as at-risk. The AUC score for either scenario would be 0.5, lower than the performance achieved by any combination of hyperparameters used for Model 1 or Model 2.

Our performances are also on par with other results on early prediction published in the literature. Regarding LSTM or RNN-based classifiers, all combinations of hyperparameters of Model 1 outperformed the single-course LSTM-based classifier used in [18] at all stages of course completion. Moreover, Model 2's performances were on par with the 0.67 accuracy displayed by the courses' halfway point in [17]. Our models still exhibit reliable performance when compared to more traditional approaches. For example, the average AUC scores obtained by Model 1 and Model 2 are comparable to those obtained by the authors in [8] on non-LSTM models in weeks 1-5 and 1-10, respectively.

Considering the demand for early identification of at-risk students and the results we have presented, we can confidently answer research question 1 (*Can the sequences of daily clicks in distinct categories be used for course-agnostic early identification of at-risk students?*) in the affirmative. Our findings demonstrate that the sequences of clicks on Moodle can be used as predictors of course performance.

### 5.2 Identification of at-risk students by the 25$^{th}$ day and 50$^{th}$ days of a course

Early student performance prediction often relies on a model's predictive ability and the actionability of its predictions. Models perform better when more data is available, but as the course progresses, the time available for educators to provide support and implement corrective measures steadily decreases. Our results follow this trend, as Model 2 (using 50-day-long sequences) outperforms Model 1 in both AUC and accuracy scores.

By the end of the 25$^{th}$ day, our models achieved AUC scores ranging from 0.629 to 0.649. Although Model 1A had the highest AUC score, this metric alone does not indicate how well the models identify at-risk students. Model 1C identified an average of 58% of at-risk students on the test data when considering recall scores, more than double the score obtained by Models 1A or 1B. Similarly, Model 2's lowest-performing combination of hyperparameters in AUC and accuracy (Model 2C) had a better recall score than its counterparts (0.537 recall vs 0.286 in Model 2A and 0.237 in Model 2B).

Based on these results, we can answer research question 2 (*What is the proportion of at-risk students that is identifiable by the LSTM-based model when predicting on each course's 25$^{th}$ day? And on the 50$^{th}$ day?):* Using only sequences of clicks made on the LMS, our LSTM-based models identified, on average, 58% of at-risk students by the end of the 25th day of each course and 53% of at-risk students by the end of each course's 50th day.

Developing an early warning system based on our LSTM-based models opens up a new avenue for supporting struggling students [24]. However, implementing such a system is not just a technical challenge but also an ethical one. The effectiveness of an early warning system can be gauged by its ability to identify as many at-risk students as possible. Still, it can also be measured by the precision of its predictions [7]. The choice between these approaches hinges on the educators' objectives for the early warning system, the resources at their disposal, and the implications of incorrect predictions. Furthermore, the aspiration for real-world deployment necessitates contemplation of aspects often overlooked in retrospective studies such as this one.

The next steps involve careful decision-making once the model flags a student as at-risk. The direct or indirect contact method could influence the student's perception of the intervention and bear privacy implications. The feedback provided should be constructive and supportive, designed to facilitate improvement rather than induce stress or anxiety. The student's potential perception of the feedback is a crucial consideration, with the overarching goal being to offer support without discouraging or stigmatizing the student.

Privacy and consent are paramount, as is the case with any predictive model in the real world. Students should be informed about their data usage and, in compliance with existing regulations, should have the option to opt-out if they so choose. These considerations underscore the complexity of implementing an early warning system in an educational setting. The challenge is not merely to develop a functional model but to create a system that benefits students while respecting their rights and autonomy. While a comprehensive discussion on this topic exceeds the scope of this work, the real-world implementation of such a system would necessitate addressing these and other questions to ensure the system's effectiveness and ethical integrity.

## 6 CONCLUSIONS

Over the years, clickstream data has gathered significant interest from researchers and educators. While it is not the only source of data, nor the most reliable, the logs recorded by Learning Management Systems (LMS) can still offer valuable information on student motivation and performance, especially during the early stages of a course when educators have limited data on their students.

This work collected data from 115 semestral and twenty-three trimestral courses taught at a European information management school during the 2020/2021 academic year. For each course, we gathered the Moodle logs generated during the first 50 days. We used them to create a dataset featuring day-wise sequences of the number of clicks made in the context of each student enrolment on fourteen different activity types. We then fed these sequences into two different LSTM-based model architectures: Model 1 was designed to identify at-risk students using the sequences from the first 25 days of each course, while Model 2 used the complete 50-day sequences.

We used a random search to optimize performance to find the best combinations of hyperparameters for each model. We selected the best-performing combinations for each model based on the average AUC score obtained from thirty repetitions of 10-fold cross-validation. We then tested these models using a 30-split repeated holdout method.

Our results show that the LSTM-based models created using the sequences from the first 25 days (Model 1) tended to underperform those trained on the more extensive 50-day sequences (Model 2), according to the AUC and accuracy scores. However, the ability of the different models to identify at-risk students, as measured by the recall score, did not increase with more extensive sequences. Using only sequences from the first 25 days, Model 1C could identify, on average, 58% of the at-risk students from the repeated-holdout test sets. On the other hand, the proportion of at-risk students identified by Model 2C (using 50-day-long sequences) was only 53%, suggesting that the additional data did not improve the models' ability to identify at-risk students but instead reduced the number of false positives.

However, our work has some limitations. Firstly, this study was retrospective, meaning that no action could be undertaken to assist students identified as at-risk. Secondly, the data was sourced from a single institution, an information management school, during a period of emergency remote learning, and no data beyond clickstreams was used, even though incorporating additional data could enhance model performance. This leads to another limitation: the homogeneity of the institution's student population could limit the findings' generalizability. Lastly, the models do not provide insights into the specific factors putting students at risk, limiting their explainability. Despite these limitations, our findings support that student activity on LMS can predict student performance on a large dataset outside of the OULAD dataset.

Future work could incorporate demographics, grades obtained in previous courses, and other student data as additional model inputs. Implementing these models in a real-world institutional LMS could prove valuable, as it would allow the models to be more actionable in real-time while enabling the exploration and fine-tuning of optimal timing for interventions. From a more long-term perspective, the insights gained from this research could lead to a greater focus on understanding beneficial types of interaction. This could eventually inform course designs to promote these types of interaction with content. For future work, it might be interesting to investigate the ability of the model to make predictions on several types of courses (e.g., math courses vs. programming courses), even within a homogenous student population. Other avenues for future work could explore incorporating extra student data, testing generalizability across institutions and disciplines (e.g., humanities or life sciences), and exploring the potential of attention-based algorithms [25], such as the Temporal Fusion Transformer [26] or the Informer [27]. As deep learning techniques in this domain are still emerging, there remains substantial potential for future research. However, it is important to note that implementing such predictive models in a real-world setting extends beyond technical challenges and involves ethical considerations. Therefore, future research should address these ethical considerations to ensure the system's effectiveness and integrity.

## REFERENCES


[1] L. P. Macfadyen and S. Dawson, 'Mining LMS data to develop an "early warning system" for educators: A proof of concept', *Computers & Education*, vol. 54, no. 2, pp. 588–599, Feb. 2010, doi: 10.1016/j.compedu.2009.09.008.

[2] R. Castro, 'Blended learning in higher education: Trends and capabilities', *Educ Inf Technol*, vol. 24, no. 4, pp. 2523–2546, Jul. 2019, doi: 10.1007/s10639-019-09886-3.

[3] M. Henderson, T. Ryan, and M. Phillips, 'The challenges of feedback in higher education', *Assessment & Evaluation in Higher Education*, vol. 44, no. 8, pp. 1237–1252, Nov. 2019, doi: 10.1080/02602938.2019.1599815.

[4] D. Turnbull, R. Chugh, and J. Luck, 'Transitioning to e-Learning during the COVID-19 pandemic: How have higher education institutions responded to the challenge?', *Educ Inf Technol*, vol. 26, no. 5, pp. 6401–6419, Sep. 2021, doi: 10.1007/s10639-021-10633-w.

[5] H. Coates, R. James, and G. Baldwin, 'A critical examination of the effects of learning management systems on university teaching and learning.', *Tertiary education and management*, vol. 11, pp. 19–36, 2005.

[6] R. Baker *et al.*, 'The benefits and caveats of using clickstream data to understand student self-regulatory behaviors: opening the black box of learning processes', *Int J Educ Technol High Educ*, vol. 17, no. 1, p. 13, Dec. 2020, doi: 10.1186/s41239-020-00187-1.

[7] M. Riestra-González, M. del P. Paule-Ruíz, and F. Ortin, 'Massive LMS log data analysis for the early prediction of course-agnostic student performance', *Computers & Education*, vol. 163, p. 104108, Apr. 2021, doi: 10.1016/j.compedu.2020.104108.

[8] H. Waheed, S.-U. Hassan, R. Nawaz, N. R. Aljohani, G. Chen, and D. Gasevic, 'Early prediction of learners at risk in self-paced education: a neural network approach', *Expert Systems with Applications*, vol. 213, p. 118868, Mar. 2023, doi: 10.1016/j.eswa.2022.118868.

[9] D. E. Rumelhart, G. E. Hintont, and R. J. Williams, 'Learning representations by back-propagating errors', *Nature*, vol. 323, pp. 533–536, 1986.

[10] S. Hochreiter and J. Schmidhuber, 'Long short-term memory', *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[11] F. A. Gers, J. A. Schmidhuber, and F. A. Cummins, 'Learning to forget: continual prediction with LSTM', *Neural Computation*, vol. 12, no. 10, pp. 2451–2471, Oct. 2000, doi: https://doi.org/10.1162/089976600300015015.

[12] N. R. Aljohani, A. Fayoumi, and S.-U. Hassan, 'Predicting at-risk students using clickstream data in the virtual learning environment', *Sustainability*, vol. 11, no. 24, p. 7238, Dec. 2019, doi: 10.3390/su11247238.

[13] J. Kuzilek, M. Hlosta, and Z. Zdrahal, 'Open University learning analytics dataset', *Sci Data*, vol. 4, no. 1, p. 170171, Dec. 2017, doi: 10.1038/sdata.2017.171.

[14] Y.-H. Hu, C.-L. Lo, and S.-P. Shih, 'Developing early warning systems to predict students' online learning performance', *Computers in Human Behavior*, vol. 36, pp. 469–478, Jul. 2014, doi: 10.1016/j.chb.2014.04.002.

[15] R. Conijn, C. Snijders, A. Kleingeld, and U. Matzat, 'Predicting student performance from LMS data: a comparison of 17 blended courses using Moodle LMS', *IEEE Trans. Learning Technol.*, vol. 10, no. 1, pp. 17–29, Jan. 2017, doi: 10.1109/TLT.2016.2616312.

[16] E. B. Costa, B. Fonseca, M. A. Santana, F. F. de Araújo, and J. Rego, 'Evaluating the effectiveness of educational data mining techniques for early prediction of students' academic failure in

introductory programming courses’, *Computers in Human Behavior*, vol. 73, pp. 247–256, Aug. 2017, doi: 10.1016/j.chb.2017.01.047.

[17] C.-C. Yu and Y. (Leon) Wu, ‘Early warning system for online STEM learning—A slimmer approach using recurrent neural networks’, *Sustainability*, vol. 13, no. 22, p. 12461, Nov. 2021, doi: 10.3390/su132212461.

[18] F. Chen and Y. Cui, ‘Utilizing student time series behaviour in learning management systems for early prediction of course performance’, *JLA*, vol. 7, no. 2, pp. 1–17, Sep. 2020, doi: 10.18608/jla.2020.72.1.

[19] A. Paszke *et al.*, ‘PyTorch: an imperative style, high-performance deep learning library’, *33rd Conference on Neural Information Processing Systems*, p. 12, 2019.

[20] X. Glorot and Y. Bengio, ‘Understanding the difficulty of training deep feedforward neural networks’, *Aistats*, vol. 9, pp. 249–256, 2010.

[21] J. Bergstra and Y. Bengio, ‘Random search for hyper-parameter optimization’, *Journal of machine learning research*, vol. 12, no. 3, pp. 281–305, 2012.

[22] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, ‘SMOTE: synthetic minority over-sampling technique’, *jair*, vol. 16, pp. 321–357, Jun. 2002, doi: 10.1613/jair.953.

[23] D. Gašević, S. Dawson, T. Rogers, and D. Gasevic, ‘Learning analytics should not promote one size fits all: The effects of instructional conditions in predicting academic success’, *The Internet and Higher Education*, vol. 28, pp. 68–84, Jan. 2016, doi: 10.1016/j.iheduc.2015.10.002.

[24] M. L. Bernacki, M. M. Chavez, and P. M. Uesbeck, ‘Predicting achievement and providing support before STEM majors begin to fail’, *Computers & Education*, vol. 158, p. 103999, Dec. 2020, doi: 10.1016/j.compedu.2020.103999.

[25] A. Vaswani *et al.*, ‘Attention is all you need’, *arXiv:1706.03762 [cs]*, Dec. 2017, Accessed: May 28, 2021. [Online]. Available: http://arxiv.org/abs/1706.03762

[26] B. Lim, S. O. Arik, N. Loeff, and T. Pfister, ‘Temporal Fusion Transformers for Interpretable Multi-horizon Time Series Forecasting’, *arXiv:1912.09363 [cs, stat]*, Sep. 2020, Accessed: Jun. 02, 2021. [Online]. Available: http://arxiv.org/abs/1912.09363

[27] H. Zhou *et al.*, ‘Informer: beyond efficient Transformer for long sequence time-series forecasting’, *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 12, pp. 11106–11115, Mar. 2021, doi: https://doi.org/10.1609/aaai.v35i12.17325.