Big Data Analysis Project

Report on

SchedWiz

Your AI study Scheduler

By:

PRIYANKA IYER	SECTION 2
GARIMA MATHUR	SECTION 1
RUCHIRA LOKHANDE	SECTION 1
VARUN PRASANNA RAO	SECTION 1
AMIT BALASAHEB GANGANE	SECTION 1

Master's in Science of Business Analytics

University of California, Davis

1 Shields Ave, Davis

CA – 95616

Table of Contents

1. Business Objective

1.1. Top Business Priority:

The top business priority of this project is to significantly improve student's academic performance and enhance their holistic educational experience by providing personalized and flexible scheduling options. This project tackles these issues by using an intelligent, data-driven scheduling system, by recognizing the increasing complexity and stress students experience in juggling daily responsibilities and exam preparation. Our methodology systematically approaches study time in different subjects by addressing the unique requirements of different individuals, past academic history, ongoing levels of difficulty, and personalizes study plans through advanced technologies and analytical tools. Apart from instilling orderly study habits and daily consistency in academic work, this integrated platform further strives to ease academic tension and stress. Finally, the program strives to make student participation seamless, promote rational study timings, and enable measurable improvements in learning achievements, thus boosting the success, well being, and happiness levels of the students.

2. Key Actionable Business Initiative

2.1. List of Potential Business Initiatives

As we work on this project, there are a few key initiatives we could focus on that are aimed at helping students manage their time better and, as a result, do better academically.

- One of the main things we're focusing on is a smart scheduling tool powered by AI. The goal is to create custom study plans for students based on how well they're doing in their subjects, how much time they have, and which subjects need more attention.
- We could also build a simple, interactive interface where students can plug in their class schedules, upcoming exams, and personal preferences. The system will then generate a daily study plan that fits their life and not the other way around.
- To make this more accessible, we're planning to integrate the tool into university Learning Management Systems (LMS) so students can use it without needing to go to a separate platform.
- One feature we could integrate is automatic schedule updates. So if a student logs that they've finished a task or if their performance shifts, their study plan changes on the fly to reflect that.
- Before rolling it out broadly, we could do a test run with a small group of students to see how useful and intuitive it really is, and to make improvements based on their feedback.

2.2. Most Impactful Initiative

The AI-based scheduling engine is the focal point of all the possible initiatives mentioned above. It tackles a prevalent problem: students frequently struggle to study effectively because they lack a strategy that takes into account both their strengths and weaknesses. This

tool alters that by assisting students in maintaining time flexibility and concentrating more on their areas of difficulty. This is something traditional approaches are unable to provide.

2.3. Actionability and Execution Plan

This initiative is both specific and actionable. We plan to execute it in clear, practical steps:

- Train predictive models using the OULAD dataset to predict student performance and classify students into risk categories like Fail, Pass, or Distinction.
- Based on those predictions, the scheduler will recommend study hours, prioritizing subjects where the student needs more support.
- We'll design the interface in Streamlit, making it user-friendly so students can input subjects, availability, and preferences with minimal effort.
- The schedule will keep evolving in real time as students check things off or enter new updates.

The ultimate goal is to help students feel more in control of their time and improve their academic results without the stress of rigid planning.

3. Metrics of Success

3.1. Key Success Metrics

Based on the training pipeline and evaluation methods implemented in the code, the following metrics are used to assess the success of the SchedWiz initiative:

- **Test Accuracy**:

  A test accuracy of 73.3% is achieved, which indicates strong generalization performance in terms of predicting the academic outcomes.

- **Training Accuracy**:

  A training accuracy of 74.98% indicates consistent learning and minimal signs of overfitting across the training process.

- **F1-Score for Each Class (Fail, Pass, Distinction)**:

  The classification report in the code calculates F1-scores. It shows that the model has the best performance in predicting "Pass" (F1 ≈ 0.81), while it has a tough time predicting "Distinction" (F1 ≈ 0.45). This means that there is class imbalance and prediction difficulty too

- **Precision and Recall per Class**:

  Along with F1-score, Precision and Recall are also reported. They provide information about the model's sensitivity (recall) and reliability (precision) for each performance category.

- **Schedule Allocation Efficiency**:

The schedule generated finally is DataFrame which includes the subject, the available hours for that subject, and the days left for the exam for that subject. Apart from this, success can also be measured on whether weaker subjects get more study hours or not, which aligns well with the project's objective of adaptive scheduling.

3.2. Metric Prioritization

Among the key metrics, the two most critical ones are:

- **Test Accuracy**: Test accuracy is the primary measure of how well the system predicts student outcomes on test data, or data that is unseen. It is crucial to measure how reliable the whole scheduling process is.
- **F1-Score for "Pass" and "Distinction" Classes**: These metrics reflect the model's nuanced performance across important academic thresholds. While "Pass" is predicted well, improving performance on "Distinction" is essential for future iterations.

3.3. Hypothesis and Expected Impact

The hypothesis is that by integrating real-time exam timelines and predicted academic risks, this personalized scheduling system will create smarter, and more relevant study schedules. This approach is expected to improve the following:

- Prediction Accuracy is expected to improve by 5–10% over baseline models because deep learning and PySpark preprocessing is used.
- Engagement with Study Plans also are expected to become better as students will find the schedules to be much more relevant towards their immediate needs, with a focus on weaker areas.
- Time Allocation Efficiency is also something that is anticipated to improve qualitatively, since schedules are dynamically customized to prioritize those subjects where the predicted performance is weaker and days left to the exam are fewer. This should improve study efficiency and lessen any last minute hasty studying.

4. Role of Analytics

4.1. Enabling the Business Initiative

Analytics enables the business initiative by fueling the main functionality which is to predict academic outcomes and generate custom study schedules for students. By using PySpark for large-scale preprocessing and a PyTorch neural network model for classification, analytics transforms raw data for student engagement and performance into actionable predictions. After these predictions are made, they guide the learner and scheduler agents to allocate optimal study hours based on the urgency of the subject as well as the predicted performance in that subject.

4.2. Refining the Initiative Through Ideation

Analytics helps refine the business initiative by making use of predicted class labels like "Fail", "Pass", and "Distinction" to drive how study hours are prioritized in the schedule. As

per the code implementation, the learner agent takes the model's predicted class and dynamically adjusts how hours are allocated. It gives extra weight to subjects that are predicted as weak. This feedback loop ensures the study schedule is not static but constantly improves as per student performance and this in turn enables continuous refinement of how the system supports its users.

4.3. Evaluating Success Using Analytics

Analytics provides clear evaluation metrics like training and test accuracy, precision, recall, and F1-scores for every performance class. They are calculated through the classification report which is generated in the model evaluation step, helping determine how well the model can differentiate between students who are most likely to fail, pass, or achieve distinction. This evaluation feedback loop is needed to monitor the model performance, to improve its prediction quality, and to also ensure that the study schedules created stay efficient, effective and relevant.

5. Thinking Through the Analytics

5.1. Data Strategy

5.1.1. Data Source (Designed vs. Existing)

Existing Data Source: Open University Learning Analytics Dataset (OULAD).

The dataset employed in this study is the Open University Learning Analytics Dataset (OULAD), which offers a rich and diverse set of educational records. It captures the longitudinal, observational behavior of over 32,000 students enrolled in multiple modules at The Open University. The data has different dimensions:

- Student demographics (Student_Info)
- Performance Assessment (Assessment)
- Student Virtual Learning Environment (VLE) engagement (Student VLE)
- Course/module structure (Courses)

Most importantly, this dataset is based on the real-world student activity with any synthetic intervention or experimental control, which makes it reliable and suitable for observational analysis with some limitations in establishing causality.

5.1.2. Outcome and Explanatory Variables

- **Outcome Variables:**

Categorical Outcome:

- Our primary target variable is the final result, which has three categories of classification labels: ***Fail, Pass, and Distinction.***

- These labels are used for classification tasks to predict student outcomes/final results.

Mapped Numeric Outcome (for regression-based prioritization):

For regression models, the categorical outcomes are mapped numerically as follows:

- Fail = 0
- Pass = 60 (passing criteria)
- Distinction = 85

This formulation supports prioritization algorithms in learning agents.

- **Explanatory Variables:**

We selected a diverse set of input features to provide a comprehensive picture of student behavior pattern, performance, and background. These include:

For assessment performance metrics we choose:

- avg_score: Average score across student assessments
- std_score: Standard deviation of student scores
- last_score: Score of the most recent student assessment
- weighted_score: Student Score weighted by assessment importance or submission order

For engagement metrics we choose:

- total_clicks: Total number of interactions with the VLE
- click_variability: Variance in daily or weekly click patterns
- active_days: Number of days with VLE activity

For student behavioral trends pattern we choose:

- score_trend: Change in student score over time showing direction and magnitude of score changes
- clicks_first_14_days: Student engagement during the initial two weeks
- clicks_last_7_days: Student engagement during the last week before assessment or dropout

For student demographics we choose:

- age_band: Categorical age group
- imd_band: Index of Multiple Deprivation

- highest_education: Highest level of prior education of a student.

5.1.3. Variation in Data

The dataset shows multiple variation, which we observe both structural and behavioral:

Engagement Variation:

- We observe assessments being submitted on different dates and schedules.
- We observe student engagement evolves across weeks and modules, making the engagement data suitable for time-series feature engineering.

Behavioral Variation:

- We observe VLE interaction patterns vary across students, indicating different learning styles, habits, and levels of engagement.

Academic Variation:

- We observe performance and participation differ across module types, presentation formats, and assessment structures, introducing academic heterogeneity into the data.

5.2. Type of Analytics

5.2.1. Exploratory Analytics

Objective: To identify hidden initial patterns, correlations, and outliers that may inform further modeling or hypotheses.

- Key Analyses:
  - We analysed distribution of assessment scores across demographic groups.
  - We also observed weekly and cumulative engagement patterns by module and student type.
  - We dropout trends associated with early engagement or demographic profiles.
- Tools & Techniques:
  - Included python's Pandas and PySpark for scalable data aggregation using groupby, join, filter etc.
  - We observed correlation heatmaps, box plots, histograms to visualize variable distributions and relationships.

5.2.2. Predictive Analytics

Objective: To showcase student outcomes using machine learning models, enabling proactive intervention and adaptive scheduling.

Models Used:

- Multilayer Perceptron (MLP): We implemented MLP using PyTorch, trained on normalized feature sets.
- Gradient Boosted Trees (GBT): For more advanced structured, tabular data, we trained using Spark MLlib pipelines.

Pipeline:

- Initially we did feature preprocessing and scaling i.e. min-max, z-score.
- Then we split the data on train-test on temporal holdout or stratified sampling.
- Then we performed model evaluation using accuracy, F1-score, and ROC-AUC.
- Additionally we performed feature importance extraction for insight generation.

Application:

- We observed predictive outputs which helps us to identify the Learner Agent system to prioritize modules/topics for students most at risk of failing or underperforming.

5.2.3. Causal Analytics

- Current Use:
  - We did not directly apply causal analytics due to the observational nature of the dataset.
- Potential Future Application:
  - **Quasi-Experimental Designs:**
    - We implement propensity score matching to estimate the causal effect of engagement i.e high or low engagement on student outcomes.
    - We explored difference-in-differences (DiD) or instrumental variables (IV).
  - **Confounder Identification and Mitigation:**
    - We included key covariates (e.g., age_band, highest_education, total_clicks) to reduce omitted variable bias in any causal analysis.

- We did segment analysis to control for module-level and demographic-level heterogeneity.

5.3. Impediments and Mitigation Plans

1. Incomplete or Missing Data

   Impediment: Model dependency can be lowered by incompetent student demographic data, missing assessment results, or gaps in engagement logs.

   Mitigation: We can use data methods such as mean/mode fill and remove records that have too many missing values. Further we can verify imputed values while testing the model.

2. Bias in Observational Data

   Impediment: Due to the lack of experimental control over the dataset, bias and confounding variables may exist.

   Mitigation: To lessen the bias caused by omitted variables, use a wide range of explanatory features, such as performance, engagement, and demographics. For upcoming causal research, think about using quasi-experimental techniques.

3. Variability Across Modules and Time

   Impediment: Model generalisation is usually impacted by the differences in student conduct and performance across modules and academic sessions.

   Mitigation: Add the student ID and module as features. Make use of time-based cross-validation. If it is necessary, we can think about creating distinct models for each module.

4. Model Complexity and Interpretability

   Impediment: Since these models can be challenging to understand, therefore MLP and GBT may not be widely adopted by stakeholders or educators.

   Mitigation: To explain key drivers, we used feature importance and for comparison, we add more straightforward models (like logistic regression).

5. Scalability of Training and Inference

   Impediment: Complex models and large datasets may require more time and resources during training.

   Mitigation: Preprocessing using PySpark and distributed computing and by improving training pipeline caching, batch processing, and model size.

6. Ethical and Privacy Concerns

   Impediment: Care must be taken when using behavioural and personal data to prevent bias or privacy violations.

   Mitigation: Encrypt data, limit access, and adhere to applicable regulations or the GDPR. Verify the model's outputs for fairness across demographic groups.

6. Executing the Analytics

6.1. Roles and Responsibilities

Ruchira, Varun, Amit, Garima, and Priyanka make up our team of five. Assessing our unique strengths, we each took responsibility for various project components.

- Amit was in charge of the preliminary preparation and data gathering.
- Varun concentrated on developing the models and incorporating the backend logic.
- Ruchira used Streamlit to work on the front-end implementation.
- Garima oversaw the model's performance evaluation and made sure it produced the desired results.
- Priyanka worked to link the technical work with user needs and assisted in defining success metrics from the viewpoints of educators and students.

The entire team worked together on documentation, testing, and feedback even though we had designated areas.

- **Data Collection**

  Amit was in charge of gathering and preparing the data for our project. In order to get the Open University Learning Analytics Dataset (OULAD) ready for modeling, he cleaned it, dealt with missing values, and carried out the required feature engineering.

- **Model Development**

  Varun was in charge of training and fine-tuning the classification models, which included Gradient Boosted Trees and MLP. Additionally, he was in charge of creating the prioritization logic that converted the model's predictions into subject-level priority scores that were then entered into the scheduling system.

- **Implementation and Evaluation**

  Using Streamlit, Ruchira created the scheduling interface, which allows students to engage with the model outputs in an intuitive application. Garima used performance metrics like F1-score and ROC-AUC to assess the model's relevance and accuracy. She also assisted in testing how well the generated schedules aligned with predicted needs. Together, they worked to ensure the tool was both functional and meaningful in real-world use.

6.2. Stakeholder Involvement in Defining Metrics

Priyanka took the lead in creating success metrics that take into account the opinions of academic stakeholders as well as students. We concluded through group discussions that success should include subject focus for underperforming students, schedule usefulness, and adaptability in addition to high model accuracy. Both interface design and model tuning were

influenced by these metrics. At this stage, each team member offered suggestions to make sure our analytics plan was thorough and well-rounded.

7. Implementation Plan

7.1. Decision-Making Influenced by Analytics

The analytics developed in our project, **SchedWiz**, an AI-powered personalized study scheduler, directly influence how students allocate their daily study hours based on predicted academic risk levels. By classifying students as *Fail*, *Pass*, or *Distinction*, the system prioritizes weaker subjects, thereby enabling students to focus where it matters most.

What we'll do differently:
Instead of relying on fixed, one-size-fits-all study plans, institutions and learners can now make **data-driven, adaptive decisions**. For example, a student predicted to be at risk of failing in Computer Science will now automatically receive more study hours allocated to that subject. This prioritization wasn't possible with traditional manual scheduling.

7.2. Workflow Integration and Adoption Strategy

The workflow looks like:

- **Learner Agent:** The Learner Agent is responsible for extracting insights from historical student performance data. It utilizes PySpark for preprocessing and is designed to predict academic outcomes categorized as Fail, Pass, or Distinction.
- **Model Training:** This stage involves training the machine learning model. Spark DataFrames are converted into tensors, and the model is trained using CrossEntropyLoss to handle multi-class classification. The model is then evaluated using accuracy metrics and a detailed classification report. The outcome is a trained neural network that can predict student outcomes based on patterns in academic activity.
- **Scheduler Agent:** The Scheduler Agent allocates study hours for each student by leveraging the predicted academic performance. It takes into account the predicted class, exam dates, and the number of daily available hours to develop a personalized study plan. Priority is given to subjects in which the student is predicted to perform poorly.
- **Output:** The final output is an actionable time-management plan in the form of a structured DataFrame. This includes information on the subject, the number of days left until the exam, and the number of study hours assigned. The system supports effective time allocation by helping students focus more on weaker subjects and facilitates decision-making using AI-generated predictions.

As students update progress or assessment scores, the schedule dynamically adapts. This feedback loop ensures **ongoing optimization** and reinforces engagement with minimal manual intervention.

7.3. Results

7.3.1. Model Summary:

The model performs well in an overall sense, with a training accuracy of 74.98% and a test accuracy of 73.3%, which suggests that it generalizes well without overfitting. It performs best at predicting "Pass" with an F1-score of 0.81. The F1-score at "Distinction" however is 0.45 which indicates that the model struggles to flag students who have achieved distinction. So the model will need a bit of more tuning. The detailed classification report is given in the image below:

```
Train Accuracy: 74.98%
Test Accuracy: 73.3%

 Test Set Classification Report:
              precision    recall  f1-score   support

        Fail       0.71      0.65      0.68       807
        Pass       0.74      0.89      0.81      2475
 Distinction       0.75      0.32      0.45       756

    accuracy                           0.73      4038
   macro avg       0.73      0.62      0.64      4038
weighted avg       0.73      0.73      0.71      4038
```

7.3.2. Final Schedule:

- **Input:** Students enter subject codes, exam dates, assessment marks, and available study hours through a simple input prompt. This setup personalizes the schedule generation based on urgency and recent academic performance.
- **Prediction:** The model re-evaluates predicted performance.
- **Schedule Generation:** A study plan is built by allocating daily study hours based on exam proximity and performance.
- **Feedback Loop:** As students update progress or assessment scores, the schedule dynamically adapts.

Given in the images below are the Input and Output respectively:

```
 Available subjects: AAA, BBB, CCC, GGG, DDD, EEE, FFF
How many subjects do you want to schedule for? 2
Enter subject code (e.g., AAA): EEE
Enter exam date for EEE (YYYY-MM-DD): 2025-06-01
Enter latest assessment mark for EEE: 50
Enter subject code (e.g., AAA): GGG
Enter exam date for GGG (YYYY-MM-DD): 2025-06-03
Enter latest assessment mark for GGG: 80
Enter number of hours you can study daily: 4
```

```
 Personalized Study Schedule:
                       Subject  Days Until Exam  Total Hours Assigned
0                    Humanities                4                   8.0
1  Engineering / Applied Physics                2                   4.0
```

8. Scaling the Initiative

8.1. Organizational Challenges and plans to Address Challenges

Scaling this analytics-driven initiative across larger institutions or multiple semesters presents key **organizational challenges**:

- **Data:**
  - *Challenge:* Take extra measure to access real time information though there were no missing values and student performance data across courses.
  - *Solution:* Integrate with institutional LMS and enforce standardized digital gradebooks and activity tracking.
- **People:**
  - *Challenge:* Instructors and advisors may resist adopting AI-driven tools or question prediction fairness.
  - *Solution:* Conduct training sessions and transparently communicate model limitations and benefits. Include staff in early pilot phases to increase trust.
- **Systems:**
  - *Challenge:* Compatibility with existing academic infrastructure.
  - *Solution:* Leverage platforms like Streamlit for LMS compatibility.
- **Culture:**
  - *Challenge:* Shifting from intuition-based academic planning to analytics-guided decisions.
  - *Solution:* Promote analytics as a *complement* to, not a replacement for, human decision-making.

8.2. Continuous Improvement Strategy

This is not a one-shot initiative. Future improvements include:

- Balancing the training dataset to improve “Distinction” prediction.
- Incorporating subjective features like motivation levels or topic difficulty.
- Adding real-time schedule reshuffling based on continuous feedback.

- **Link to the Google Drive:**

https://drive.google.com/drive/folders/1ya9FkThCS16e1jCQFRajFEvIakQOnoM2?usp=sharing

---