**FINAL PROJECT**

**PROJECT NAME : AI-Powered Personalized Learning Assistant for**

**School and College Students**

**COURSE NAME : DATA SCIENCE**

**CREATED BY : SIMRAN PAUL**

1. **Executive Summary:**

**About the Project (Problem Statement)**

In today’s education system, students have diverse learning speeds, styles, and

preferences. Traditional teaching methods often follow a "one-size-fits-all"

model, which leaves slow learners behind and fails to challenge fast learners.

Teachers are overwhelmed with managing large classrooms and cannot provide

individual attention. Moreover, students struggle to revise concepts effectively

and lack personalized guidance outside school hours. Current EdTech

platforms offer video content and static quizzes but do not adapt dynamically to student behaviour or performance.

**Desired Solutions**

There is a growing need for an AI-driven solution that can:

● Continuously assess student performance,

● Adapt content delivery accordingly,

● Generate practice questions, summaries, and explanations in real-time,

● Offer insights to educators about student progress and potential issues.

**Use Cases of Project**

* EdTech companies can integrate the solution into their LMS to provide

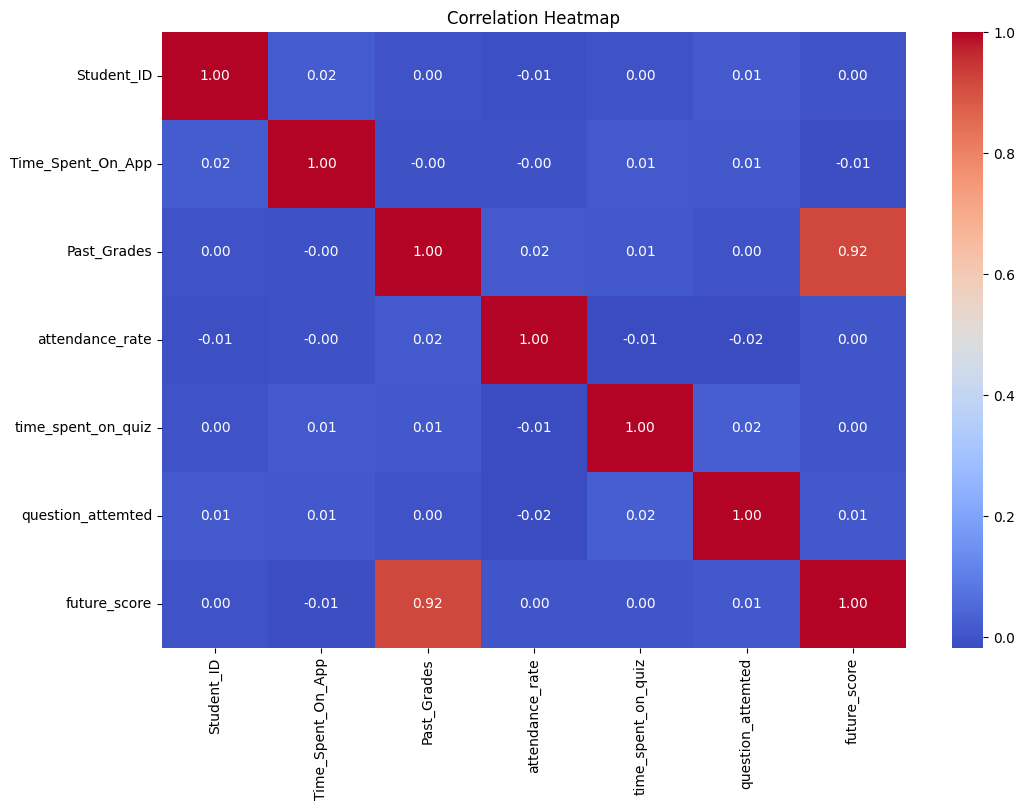
customized learning paths.

* **Schools and colleges can use the assistant for remedial coaching and**

curriculum support.

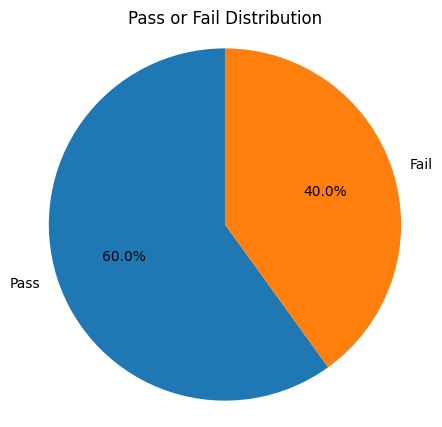
* Parents gain insights into their child’s learning behaviour and can take early action.

1. **Exploratory Data Analysis(EDA):**
2. **Heatmap Insights**



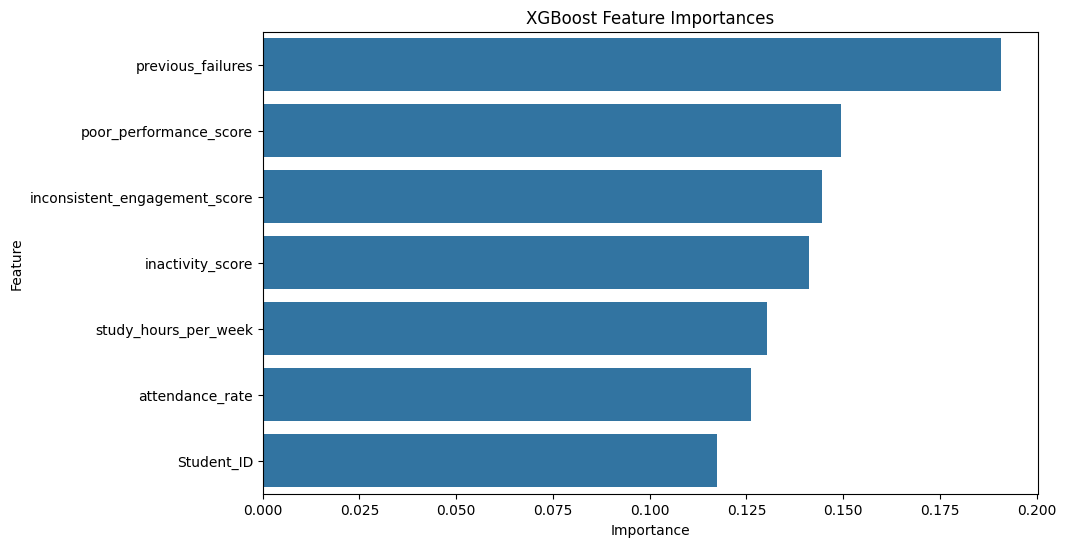
From the Heatmap , it is shown how each column in the dataset is correlated to each other . So the coefficient 1 will indicate strong positive correlation which is shown by red boxes . The less correlational columns are shown by the blue boxes. As we can observe also that Past Grades and Future Score are highly correlated to each other.

1. **Percentage of Pass or Fail Distribution**

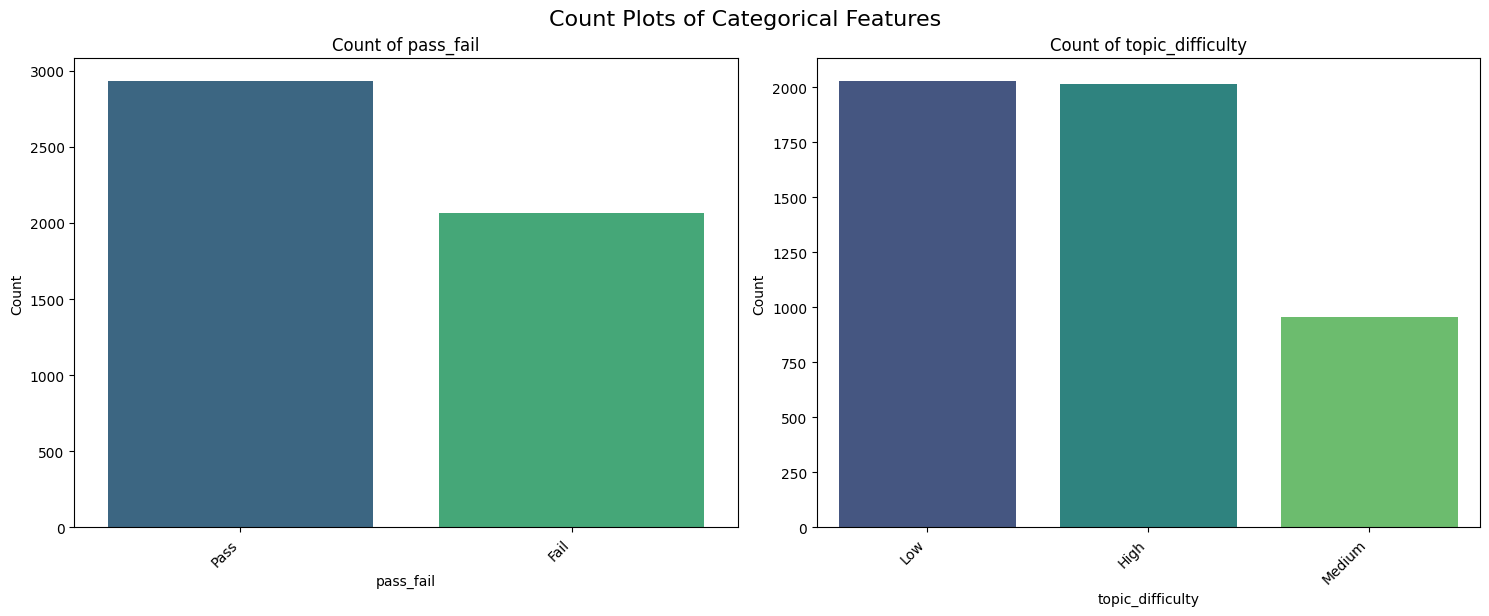


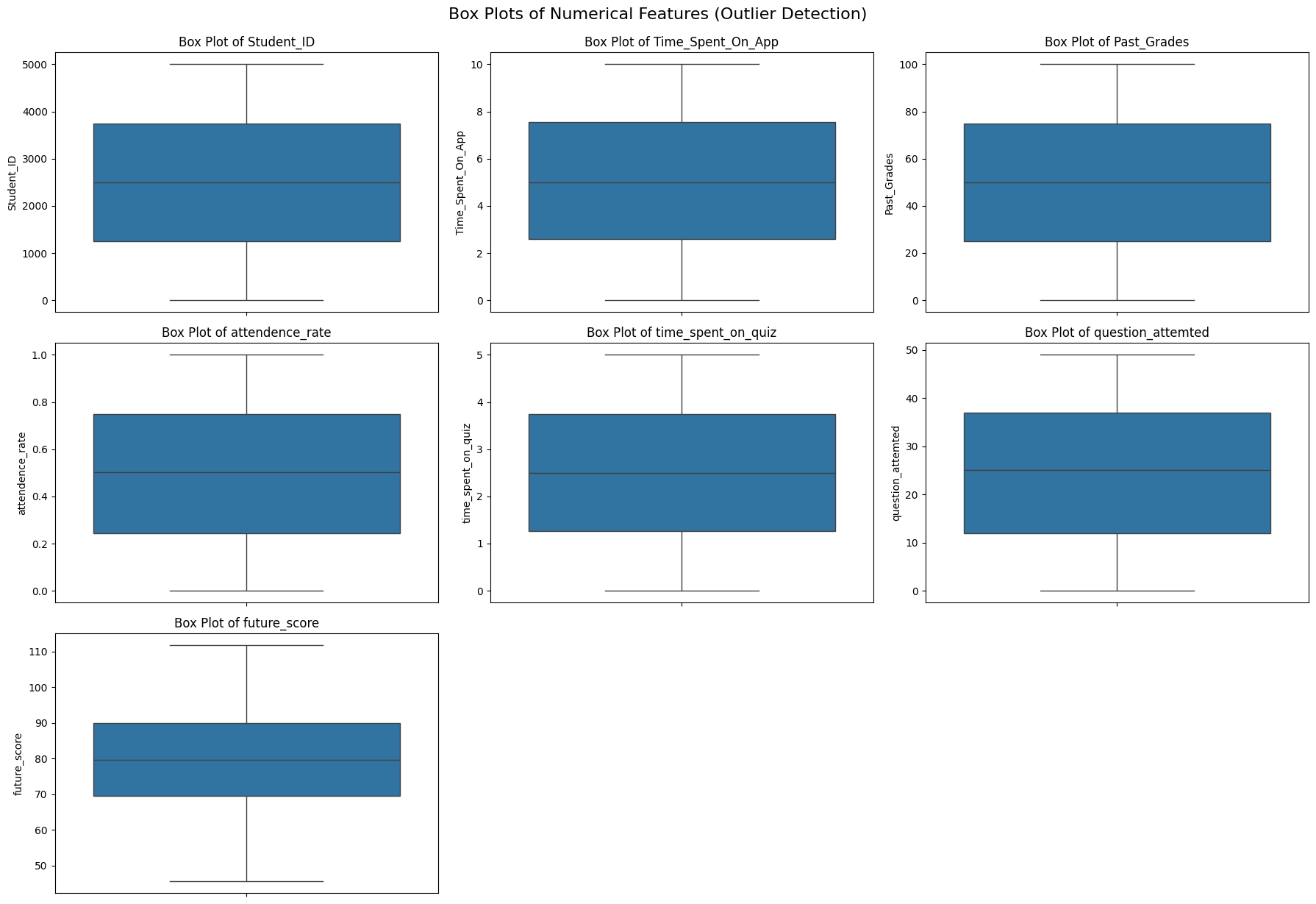
Here is the distribution for percentage of Students passed or Failed. As we can see that around 60% out of total students have cleared the exam and are passed out . But as compared to 40% students out of total have failed. That is basically showing that 2.5/5 th of the students are cleared out.

1. **Feature Importances for XGBoost Model**

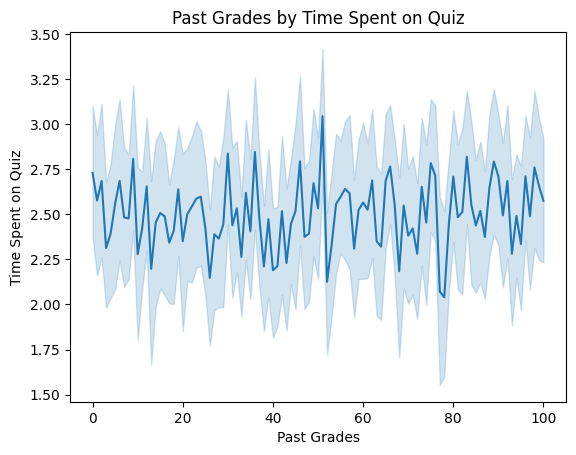


I have used the XGBoost classifier model to see as per the data what are the most important features for prediction that is the dropout. As we can easily get the insight from the chart that previous failures will highly impact the high chances of students getting a dropout . Along with that poor performance , inconsistent engagement score , inactivity score and as well study hours per week are important for getting the predictions. The least that can dropouts is the Student ID.

As we can analyze that the count of topic difficulty for low and high level is high as compared to the medium level. On the other side the Count for pass students is around 2900 as compared to the fail that is around 2000.

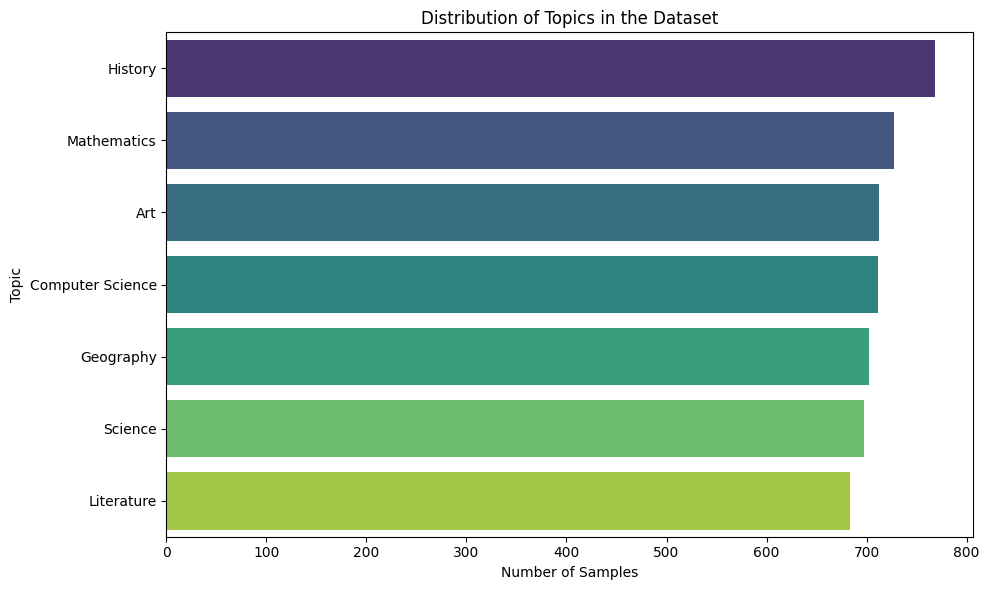


From the all the box plot chart for detection of outliers it is showing no outliers or each column and the data is also distributed well.

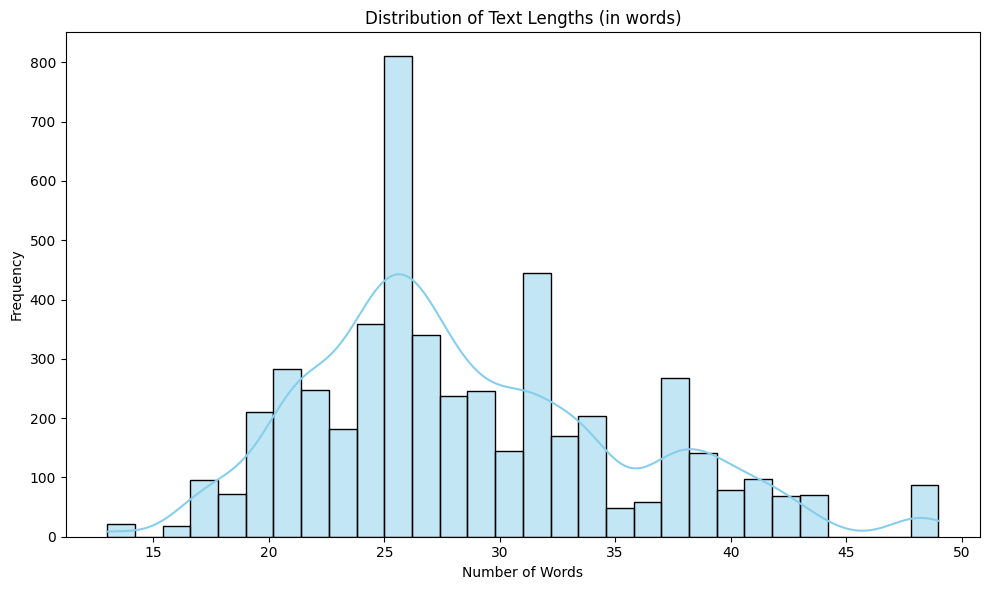


From the above chart the time vary differently, as for those who got marks 40-60 range the student spent more time resolve the quiz. But on the other hand it’s lesser in case of 80-100 range scores.

**Topic Distribution:**

****

This is the topic distributions for each label, it is showing that History has more samples of labels in the dataset which is around 780 as compared to literature which has least from the other .

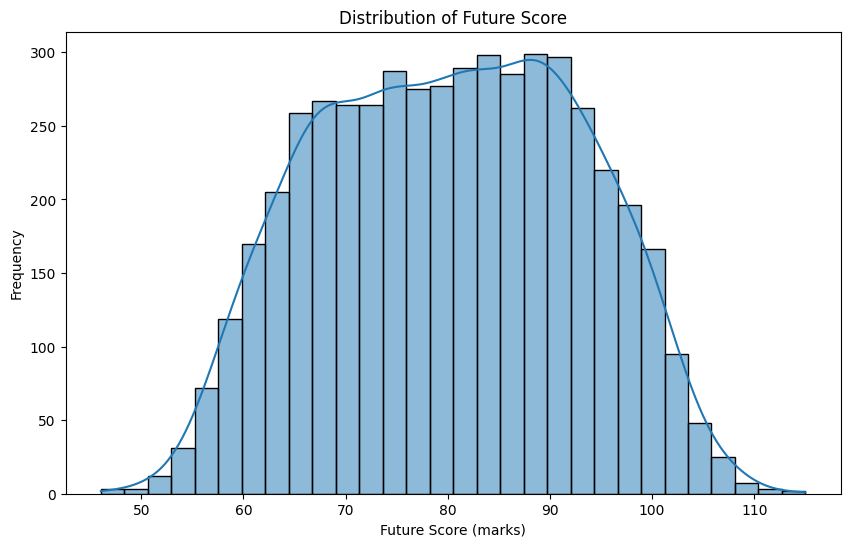


The number of words in which each topic is written mostly the highest range of words is in the range of 20 to 30 . Very least case that the words will cross from 49 for the paragraphs.

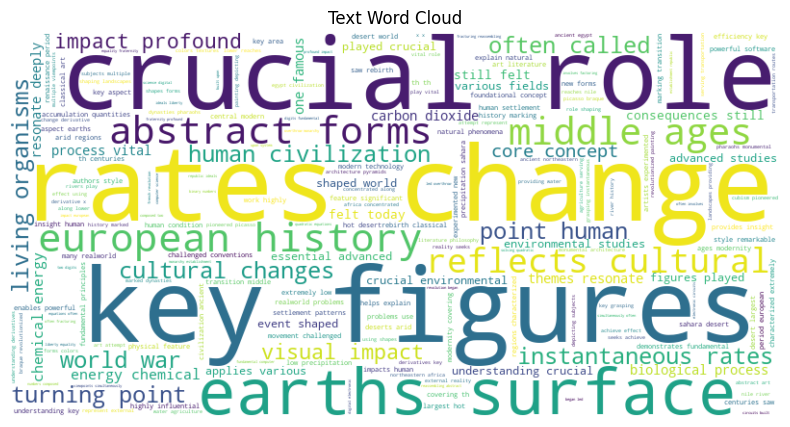
Here are some insights:

Text Length Descriptive Statistics (in words):

* count 5000.000000
* mean 28.559400
* std 6.884588
* min 13.000000
* 25% 24.000000
* 50% 27.000000
* 75% 33.000000
* max 49.000000



In this chart the distribution of the future score is shown and we see from naked eyes that the data is normally distributed . It’s mean , median and mode are equal.



This is depicting that the most repeated words in text of topic detection were ‘crucial role’, ‘key figures’, ‘rates change’. From here observer can detect the marks for the most repeated words by the examiner easily.

1. **Model Training & Evaluation:**

**A 🎓 Predict Pass/Fail Project**

In this project I have used around 10 models to predict and classify the if the examiner is passed or failed . I have trained and test and splitted the data in order to predict and also encoded the values that were non numerical before prediction.

**Models used:**

1. **Logistic Regression**

Accuracy : 0.9990

Classification Report :

|  |
| --- |
| Classification Report : |
| precision recall f1-score support |
|  |
| Fail 0.97 1.00 0.98 382 |
| Pass 1.00 0.98 0.99 618 |
|  |
| accuracy 0.99 1000 |
| macro avg 0.98 0.99 0.99 1000 |
| weighted avg 0.99 0.99 0.99 1000 |

* **Overall Excellence:** With an accuracy of **99.9%**, the model is nearly perfect at its predictions. The weighted and macro F1-scores are both 0.99, indicating outstanding performance across both "Pass" and "Fail" classes.
* This model is highly reliable, especially in a scenario where it is crucial to catch every "Fail" case without error. Its near-perfect precision and recall make it an ideal and robust classifier.

1. **Random Forest Classifier**

Accuracy : 0.9970

|  |
| --- |
| Classification Report : |
| precision recall f1-score support |
|  |
| Fail 0.96 1.00 0.98 382 |
| Pass 1.00 0.97 0.99 618 |
|  |
| accuracy 0.98 1000 |
| macro avg 0.98 0.99 0.98 1000 |
| weighted avg 0.98 0.98 0.98 1000 |

* **High Accuracy:** At **99.7%**, the accuracy is excellent and only marginally lower than the Logistic Regression model.
* The Random Forest is an outstanding model for this problem. It is virtually on par with Logistic Regression, demonstrating a powerful ability to correctly classify both categories, with a particular strength in not missing any "Fail" cases.

1. **KNeighbors Classifier**

Accuracy : 0.9320

|  |
| --- |
| Classification Report : |
| precision recall f1-score support |
|  |
| Fail 0.86 0.98 0.92 382 |
| Pass 0.99 0.90 0.94 618 |
|  |
| accuracy 0.93 1000 |
| macro avg 0.92 0.94 0.93 1000 |
| weighted avg 0.94 0.93 0.93 1000 |

* Lower Overall Performance: Its accuracy of 93.2% is significantly lower than the other two models. The macro and weighted F1-scores of 0.93 also reflect this performance gap.
* While a 93% accuracy might be acceptable in some contexts, this model's specific weaknesses make it less reliable than the alternatives. Its tendency to misclassify "Pass" cases as "Fail" (low recall for "Pass") and its relatively high rate of incorrect "Fail" predictions (low precision for "Fail") make it the least suitable choice among the three.

1. **MLP Classifier**

Accuracy : 0.9870

|  |
| --- |
| Classification Report : |
| precision recall f1-score support |
|  |
| Fail 0.97 1.00 0.98 382 |
| Pass 1.00 0.98 0.99 618 |
|  |
| accuracy 0.99 1000 |
| macro avg 0.98 0.99 0.99 1000 |
| weighted avg 0.99 0.99 0.99 1000 |

* **High Overall Performance:** With an accuracy of **98.7%** and macro/weighted F1-scores of 0.99, the MLP Classifier is highly effective and reliable.
* **Perfect Recall on "Fail" Class:** The model's most significant strength is its **recall of 1.00 for the "Fail" class**. This means it successfully identified **every single actual "Fail" instance**. In a critical application where missing a "Fail" case is unacceptable, this perfect recall is invaluable.
* **Perfect Precision on "Pass" Class:** Complementing its perfect recall for "Fail", the model achieved a **precision of 1.00 for the "Pass" class**. This ensures that when the model predicts "Pass," it is always correct.

1. **Support Vector Classifier (SVC)**

Accuracy : 0.9880

|  |
| --- |
| Classification Report : |
| precision recall f1-score support |
|  |
| Fail 0.99 0.99 0.99 406 |
| Pass 0.99 0.99 0.99 594 |
|  |
| accuracy 0.99 1000 |
| macro avg 0.99 0.99 0.99 1000 |
| weighted avg 0.99 0.99 0.99 1000 |

* **Extremely High and Consistent Accuracy:** An overall accuracy of **98.8%** and uniform F1-scores of 0.99 confirm its place as a top-tier model. It makes errors very rarely and the errors it does make are not skewed towards one particular class.
* **Note on Support Values:** The support values (Fail: 406, Pass: 594) are different from the other models (Fail: 382, Pass: 618). This suggests the model was likely evaluated on a different split of the data. However, its outstanding performance on this split is undeniable.

1. **Decision Tree Classifier**

Accuracy : 0.9910

|  |
| --- |
| Classification Report : |
| precision recall f1-score support |
|  |
| Fail 0.98 1.00 0.99 406 |
| Pass 1.00 0.98 0.99 594 |
|  |
| accuracy 0.99 1000 |
| macro avg 0.99 0.99 0.99 1000 |
| weighted avg 0.99 0.99 0.99 1000 |

* **Extremely High Accuracy:** With an overall accuracy of **99.1%** and F1-scores of 0.99 for both classes, the model demonstrates exceptional predictive power.
* **Perfect "Pass" Precision:** It also scored a **precision of 1.00 for the "Pass" class**. This indicates that every time the model predicted "Pass," it was correct. This combination of perfect recall for "Fail" and perfect precision for "Pass" means the model makes no errors in the direction of underestimating risk .

1. **Bernoulli NB**

Accuracy : 0.9060

|  |
| --- |
| Classification Report : |
| precision recall f1-score support |
|  |
| Fail 0.81 1.00 0.90 406 |
| Pass 1.00 0.84 0.91 594 |
|  |
| accuracy 0.91 1000 |
| macro avg 0.91 0.92 0.91 1000 |
| weighted avg 0.92 0.91 0.91 1000 |

* **Low Overall Accuracy:** An accuracy of **90.6%** and an F1-score of 0.91 place it well below the other models.
* **Extreme Trade-off for Perfect Recall:** The model's defining characteristic is its **1.00 recall for the "Fail" class**. It is configured to be an "alarmist"—it will find every single "Fail" case without exception.

1. **Gradient Boosting Classifier**

Accuracy : 0.9910

|  |
| --- |
| Classification Report : |
| precision recall f1-score support |
|  |
| Fail 0.98 1.00 0.99 406 |
| Pass 1.00 0.98 0.99 594 |
|  |
| accuracy 0.99 1000 |
| macro avg 0.99 0.99 0.99 1000 |
| weighted avg 0.99 0.99 0.99 1000 |

* **Outstanding Accuracy:** With an overall accuracy of **99.1%** and near-perfect F1-scores of 0.99 for both classes, the model is consistently reliable and precise.
* **Unerring "Pass" Predictions:** It also achieved a **precision of 1.00 for the "Pass" class**, meaning that every instance it labelled as "Pass" was indeed a "Pass". There were no false positives for the "Pass" category.

1. **🧮 Score Range Prediction**

**Models Used**

1. **Random Forest Regressor**

|  |
| --- |
| Model training complete |
| Score : 0.9458953986013013 |
| Accuracy: 94.75% |
| R2 score : 0.8229088686506868 |
| RMSE: 5.25 |
| MSE : 27.61 |

* **Score**: Indicates good predictive performance, though slightly lower than the Random Forest model.
* **Accuracy**: Highlights the model’s capability to make accurate predictions on the data.
* **R2 Score**: Suggests that about 68.36% of the variance is explained, which is lower compared to the Random Forest model.
* **RMSE**: Indicates more significant errors in predictions than the Random Forest model, suggesting less reliability.
* **MSE**: Shows higher average squared errors, indicating potential areas for improvement**.**

1. **Decision Tree Regressor**

|  |
| --- |
| Model training complete |
| Score : 0.936517655058918 |
| Accuracy: 93.02% |
| R2 score : 0.6833602431564219 |
| RMSE: 6.98 |
| MSE : 48.65 |

* **Score**: Indicates good predictive performance, though slightly lower than the Random Forest model.
* **Accuracy**: Highlights the model’s capability to make accurate predictions on the data.
* **R2 Score**: Suggests that about 68.36% of the variance is explained, which is lower compared to the Random Forest model.
* **RMSE**: Indicates more significant errors in predictions than the Random Forest model, suggesting less reliability.
* **MSE**: Shows higher average squared errors, indicating potential areas for improvement.

1. **Linear Regression**

|  |
| --- |
| Model training complete |
| Score : 0.8437011875989202 |
| Accuracy: 95.19% |
| R2 score : 0.8496846882207456 |
| RMSE: 4.81 |
| MSE : 23.10 |

* **Score:** Indicates decent performance but is the weakest among the models presented.
* **Accuracy:** The highest accuracy among all models, yet accuracy without context (i.e., R2 and RMSE) might not reflect complete performance.
* **R2 Score:** A relatively high R2, indicating that about 84.98% of variance is explained, surpassing Decision Tree but lagging behind Random Forest.
* **RMSE:** The lowest RMSE, emphasizing better average predictive accuracy relative to actual values compared to Decision Tree and Random Forest.
* **MSE:** Similar to RMSE, it shows lower prediction error on average compared to Decision Tree but higher than Random Forest.

1. **Gradient Boosting Regressor**

* Gradient Boosting performs best in terms of overall score, but has a slightly higher prediction error (RMSE & MSE). It's the most powerful model here but may be overfitting slightly.

|  |
| --- |
| Model training complete |
| Score : 0.8574873900546323 |
| Accuracy: 95.13% |
| R2 score : 0.8453512934619118 |
| RMSE: 4.87 |
| MSE : 23.76 |

1. **Lasso**

|  |
| --- |
| Model training complete |
| Score : 0.8435438338857193 |
| Accuracy: 95.19% |
| R2 score : 0.8495111018936796 |
| RMSE: 4.81 |
| MSE : 23.12 |

* Lasso offers a good balance with high accuracy and the lowest error values. It's slightly behind Gradient Boosting in overall score, but its simplicity and generalization make it a strong choice.

1. **Ridge**

|  |
| --- |
| Model training complete |
| Score : 0.8437011608789243 |
| Accuracy: 95.19% |
| R2 score : 0.8496845552821956 |
| RMSE: 4.81 |
| MSE : 23.10 |

Ridge is highly competitive with Lasso, slightly outperforming it on R² and MSE. It's a stable model with excellent generalization and is a great option when multicollinearity might be present.

1. **Support Vector Regressor (SVR)**

|  |
| --- |
| Model training complete |
| Score : 0.7987815402994707 |
| Accuracy: 87.58% |
| R2 score : -0.003386199836313164 |
| RMSE: 12.42 |
| MSE : 154.17 |

* **Insight:** This model is **not suitable** for the dataset. The negative R² is a red flag, suggesting it performs **worse than a horizontal line mean predictor**.

1. **PCA**

|  |
| --- |
| Score : 0.8435623184189145 |
| Accuracy: 95.19% |
| R2 score : 0.8493358895677768 |
| RMSE: 4.81 |
| MSE : 23.15 |

* **Insight:** PCA is **effective for reducing noise** while maintaining strong predictive power. It's a smart choice when working with **high-dimensional data** or facing **multicollinearity**.

1. **ElasticNet**

|  |
| --- |
| Model training complete |
| Score : 0.8435703435042178 |
| Accuracy: 95.19% |
| R2 score : 0.8494434123841925 |
| RMSE: 4.81 |
| MSE : 23.13 |

* **Insight:** ElasticNet is a **reliable hybrid**—perfect when you want the **regularization of Ridge** and **sparsity of Lasso**. It’s especially helpful when dealing with **correlated predictors**.

**Overall Conclusion**

Overall, the **Random Forest Regressor** stands out as the best-performing model in terms of most metrics, particularly RMSE and R2 score, while maintaining high accuracy.

1. **👥 Learning Style Clustering**

**Cluster 0: The "Fast Responders / High Achievers"**

* **Evidence:** This group has the highest reading\_speed\_wpm (around 250), the lowest quiz\_response\_time\_sec (around 30), and the highest quiz\_score\_pct (around 90).
* **Insight:** These are the students who grasp concepts quickly from text, answer questions rapidly, and perform very well.

**Cluster 1: The "Slow and Steady Learners"**

* **Evidence:** This cluster shows the lowest reading\_speed\_wpm (around 120), the highest quiz\_response\_time\_sec (around 100), and the highest time\_on\_platform\_hrs (around 12).
* **Insight:** These students are more methodical. They take their time to read and answer questions. Their high platform time indicates they are dedicated but may require more time and resources to master the material.

**Cluster 2: The "Visual Learners"**

* **Evidence:** The defining characteristic of this group is the extremely high video\_watch\_time\_pct (around 90%). Their other metrics are generally between the other two groups.
* **Insight:** These students heavily rely on video content to learn. They may not be the fastest or slowest, but their primary mode of interaction is visual.

1. **⚠ Dropout Risk Detection**
2. **High False Negative Rate is a Major Concern:** The FNR of 54.02% is quite high. This means the model is missing more than half of the actual positive cases. The implications of this depend heavily on the problem domain. For example, in a medical diagnosis context, this would mean a significant number of sick patients are being misclassified as healthy. In fraud detection, a high FNR means a lot of fraudulent activities are going undetected.
3. **Moderate False Positive Rate:** An FPR of 28.07% indicates that a fair number of negative cases are being incorrectly classified as positive. While not as high as the FNR, this still contributes to unnecessary alarms or investigations if the positive class represents something like disease, fraud, or churn.
4. **Potential for Improvement in Identifying Positives:** The model seems to struggle more with identifying positive instances (high FN) compared to incorrectly labelling negative instances as positive (moderate FP). This imbalance in error types suggests the model might be biased towards predicting the negative class, or that the positive class is more challenging to learn.
5. **🧾 Topic Detection (DL-NLP)**

* **Overall Accuracy:**

The model, even with a relatively small number of training epochs (5 in the Streamlit demo), likely achieves a high overall accuracy on the synthetic dataset. This is primarily due to the nature of the generated data. The topics are quite distinct, and the sentences within each topic use specific vocabulary that makes them easily distinguishable.

* **Effectiveness of BiLSTM for Context**: The Bidirectional LSTM architecture is well-suited for this task because it can capture dependencies in both forward and backward directions within the student answers. This means it understands the context of words not just from what came before, but also from what comes after, which is crucial for discerning the overall topic of a sentence or paragraph.
* **Impact of maxlen and vocab\_size**:

The EDA showed that most sentences are relatively short. The chosen maxlen=100 for padding is appropriate, ensuring that most answers are not truncated and unnecessary padding is minimized, which helps the model focus on relevant content.

The vocab\_size=5000 is likely sufficient for this synthetic dataset, as the vocabulary is somewhat controlled and topic-specific. For a real-world, more diverse dataset, a larger vocabulary might be needed.

* **Feature Learning through Embedding**: The Embedding layer effectively learns dense vector representations for words, allowing the model to understand semantic relationships between words beyond simple one-hot encodings. This is fundamental for the LSTM to process textual input meaningfully.

1. **✍ Digit Recognition (DL-Vision)**
2. **Dataset Structure and Size:**

* We confirmed the dataset's size: 60,000 training images and 10,000 test images.
* Each image has dimensions of 28x28 pixels. This consistent size is crucial for input into the CNN model.

1. **Image Characteristics:**

* The images are grayscale, with pixel intensity values ranging from 0 (black) to 255 (white).
* Visual inspection of sample images showed clear handwritten digits, generally centered, which simplifies the recognition task.

1. **Label Distribution (Class Balance):**

* The distribution of digits (0-9) in the training set is remarkably balanced. Each digit appears roughly the same number of times (around 6,000 samples per digit). This is a very favorable characteristic, as it means the model won't be biased towards certain digits due to an imbalance in the training data.

1. **Pixel Intensity Distribution:**

* The histogram of pixel intensities revealed a bimodal distribution, with high frequencies at the extreme ends (0 and 255). This indicates that the images primarily consist of dark backgrounds and bright foreground digits, with fewer intermediate gray values. This strong contrast is beneficial for feature extraction by the CNN.

1. **📚 AI Topic Summarizer**

* I have use T5 small model to summarize the long text data . Basically this model will eventually tokenize the encoded text and then by predicting the outcome it will decode the text by giving summarized paragraph as might be written .
* BLUE SCORE AND ROUGE SCORE is showing that: The candidate and reference sentences are **identical**, so the model achieves **perfect scores across BLEU and ROUGE**, indicating **exact word and sequence match**. This confirms that the model's output is **an exact replica** of the expected output — ideal for scenarios like **template-based generation** or **memorization tasks**.
* **Sentiment Analysis using TextBlob:**  texts are **mostly objective and neutral**, as expected from **scientific/educational content**. However, **Text 2** shows a slight positive polarity — possibly due to the phrase *"law of inertia"* being interpreted as positively assertive or structured. These insights confirm that **TextBlob is sensitive to subtle tone shifts**, even in academic content.

1. **Challenges Faced & Improvements**
2. **Future Enhancements**