**A**

***SYNOPSIS REPORT***

*On*

**Personalized Reading Recommendations with Summarization**

*Submitted in partial fulfilment of the requirements of the degree of*

**BACHELOR OF TECHNOLOGY**

**Computer Science & Engineering (DS)**



**Submitted by**

Adarsh Singh (FET/BCE/2021-25/031)

Shrestha Kumar Agarwal (FET/BCE/2021-25/040)

**Guided by**

Dr. Deepak Sahoo

**FACULTY OF ENGINEERING & TECHNOLOGY**

**SRI SRI UNIVERSITY**

**Ward No.3, Sandhapur, Godisahi, Odisha 754006**

**AUGUST 2024**

**Content**

1. Name (Group Name) and Title of the Project
2. Problem of Statement
3. Objective and scope of the project
4. Methodology
5. Hardware & Software to be used
6. Application and Future Scope of the Project
7. Project Timeline (Gantt chart/ PERT chart)
8. References/Bibliography.

**Group Name:**

Group 2

* Members:
  1. Shrestha Kumar Agarwal (FET/BCE/2021-25/040)

No.: 9193911757

Email ID: [shrestha.a2021btcseds@srisriuniversity.edu.in](mailto:shrestha.a2021btcseds@srisriuniversity.edu.in)

Semester 7

* 1. Adarsh Singh (FET/BCE/2021-25/031)

No.: 9667516780

Email ID: [adarsh.s2021btcseds@srisriuniversity.edu.in](mailto:adarsh.s2021btcseds@srisriuniversity.edu.in)

Semester 7

**Title of the Project:** Personalized Reading Recommendations with Summarization

**Problem of Statement:**

In today's information-rich digital environment, users often struggle to find relevant reading material that aligns with their interests. Existing recommendation systems may offer suggestions based on user behaviour, but users are frequently overwhelmed by the volume of content and may lack the time to explore each recommendation in depth. Therefore, developing accurate and efficient mechanisms is crucial. Thus, the problem is to create systems that can reliably identify people’s interests based on the history and recommend suitable options and help them summarize the articles as well.

**Objective and Scope:**

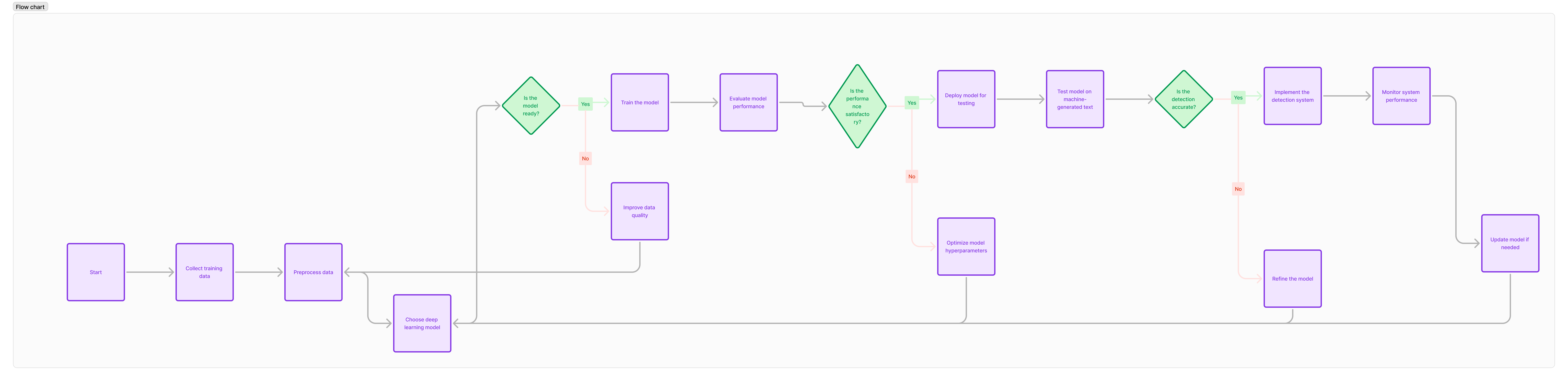
The objective is to develop a system that provides personalized reading recommendations to users based on their reading history and preferences, while also generating concise summaries of recommended articles or books. The goal is to enhance user engagement by not only suggesting relevant content but also giving users a quick overview of what to expect from each recommendation.

**Methodology:**

The project will follow these key steps:

* Data Collection: Collect a diverse set of articles, books, or other reading materials. Gather data on user reading history, preferences, and interactions. This may include clicks, reading times, ratings, and feedback.
* Data Preprocessing: Preprocess the collected data to remove noise, handle missing values, and normalize text formats. This step ensures that the data is clean and ready for feature extraction.
* Feature Extraction: Extract important features from the preprocessed data. These features may include text length, word frequency, grammar patterns, and other linguistic characteristics that help differentiate between human and machine-generated content.
* Model Training: Train a machine learning model using the extracted features. The model will learn to classify text as either human-written or machine-generated based on the patterns identified during feature extraction.
* Model Optimization: Optimize the trained model to improve its performance. This may involve tuning hyperparameters, using cross-validation techniques, and applying regularization methods to prevent overfitting.
* Model Validation: Validate the optimized model using a separate validation dataset to assess its accuracy, precision, recall, and other performance metrics. This step ensures that the model generalizes well to unseen data and is reliable for real-time detection.

This methodology ensures a systematic approach to developing a robust system for detecting machine-generated text in real-time communication platforms.



**Hardware & Software:**

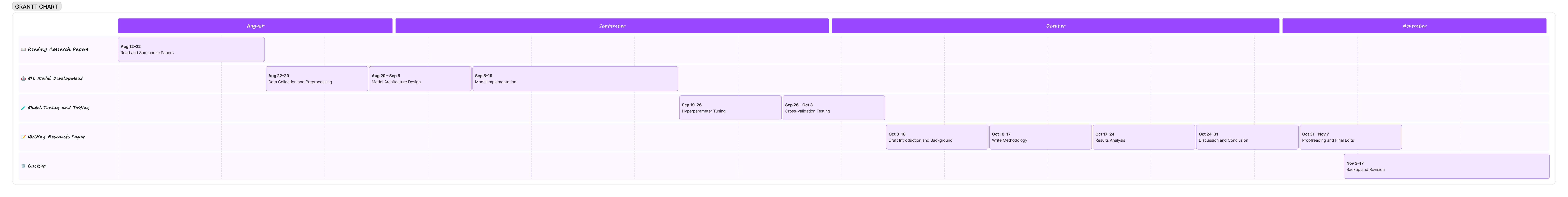
1. Hardware:
   1. Standard computing systems for development and testing/Google Colab.
   2. Servers for hosting the system (if applicable).
2. Software:
   1. Programming Languages: Python
   2. Machine Learning Libraries: Pandas, Numpy, Seaborn, NLTK , TensorFlow, PyTorch, scikit-learn
   3. Development Tools: IDEs like Visual Studio Code, Jupyter Notebook
   4. Version Control: Git, GitHub
   5. Design Tools: Figma
   6. Project Management: Figma

**Application and Future Scope:**

* Application:
  + News Aggregators (e.g., Flipboard, Google News): Personalized recommendations and summaries of news articles can help users stay informed on topics of interest without being overwhelmed by the volume of information.
  + Academic Research Tools (e.g., Mendeley, Google Scholar): In academic settings, personalized recommendations and concise summaries of research papers can assist students and researchers in discovering relevant studies more effectively.
  + Content Subscription Services (e.g., Medium, Substack): These platforms can use personalized recommendations to suggest articles, while summarization helps users quickly gauge whether the content is worth reading.
* Future Scope:
  + Integration with Voice Assistants (e.g., Alexa, Google Assistant): Personalized reading recommendations and summaries can be delivered via voice, allowing users to discover and explore content hands-free, enhancing accessibility.
  + Cross-Platform Personalization: Developing systems that sync across various devices and platforms (e.g., smartphones, tablets, e-readers) to provide consistent and personalized reading experiences, no matter where users are accessing content.
  + Enhanced Summarization with AI-driven Creativity: Future advancements in AI could enable the creation of not just concise summaries, but also engaging and creative summaries, such as interactive story previews, visual summaries, or even audio summaries, making content more accessible and appealing across different media formats.

**Timeline:**

Grantt Chart of Timeline:



**References:**

Few Papers to mention are:

* <https://arxiv.org/pdf/2104.10083v3>
* <https://journal.esrgroups.org/jes/article/view/3053/2387>
* <https://ar5iv.labs.arxiv.org/html/2106.08934>