**Group B14 Interim Report**

**1 Introduction**

Our group project aims to develop a comprehensive web application that combines various computer science techniques to build a deep learning-based stock price prediction distributed system.

Throughout the project, we acquired fundamental skills in several areas of computer science that allowed us to make significant progress.

This interim report highlights the technologies we have adopted, as well as the technologies we have been learning and will utilize in the following development process.

**2 Learning**

**2.1 Front End**

To create an engaging and user-friendly interface, we have become proficient in front-end development using the following techniques and technologies:

- Vue.js: We have gained expertise in building dynamic web applications using Vue.js framework.

- JavaScript (JS): We have extensively used JavaScript for client-side scripting to enhance interactivity and responsiveness.

- Node.js: Our team has utilized Node.js for server-side scripting and developing scalable network applications.

- ElementUI: We have employed ElementUI, a popular component library, to enhance the design and functionality of our application.

- H5: We have implemented HTML5 (H5) features to ensure compatibility and utilize modern web capabilities.

- CSS: We have applied Cascading Style Sheets (CSS) to style and format our web pages effectively.

**2.2 Algorithm**

To provide advanced data processing and analysis capabilities, we have acquired expertise in the following algorithmic techniques:

- PyTorch: We have extensively utilized the PyTorch library for deep learning tasks, enabling us to develop and train complex neural networks.

- RNN (Recurrent Neural Network): We have implemented RNNs to process sequential data and extract meaningful patterns.

- LSTM (Long Short-Term Memory): Our team has employed LSTM networks for modeling long-term dependencies in sequential data.

- Specific Model: We will review and analyze stock prediction research papers to identify and implement suitable models that align with our project requirements. By comparing and evaluating different models, we aim to develop a customized approach for accurate stock price prediction.

**2.3 Back End**

To ensure efficient data processing and seamless communication between the front end and databases, we have learned the following back-end techniques:

- Spring Cloud: We have utilized the Spring Cloud framework to develop scalable and resilient microservices architecture.

- Flask: Our team has employed Flask, a lightweight Python web framework, to build RESTful APIs and handle backend logic.

- Nginx: We have configured Nginx as a reverse proxy server, improving the performance and scalability of our application by efficiently distributing incoming network traffic.

**2.4 Data**

To store, manage, and process large volumes of data, we have acquired proficiency in the following data-related techniques:

- MySQL: We have utilized MySQL as our relational database management system (RDBMS), ensuring structured storage and efficient querying.

- RabbitMQ: Our team has employed RabbitMQ as a message broker to enable asynchronous communication between components of our application.

- Redis: We have used Redis as a fast in-memory data store to cache frequently accessed data, enhancing performance.

- Dataset: We will collect a dataset of historical stock prices from reputable financial sources to train our deep learning models for stock price prediction.

**2.5 Server**

To deploy our application and ensure its availability, we have gained expertise in the following server-related techniques:

- Linux: We have utilized Linux as the operating system for our server, leveraging its stability, security, and flexibility.

**2.6 Collaboration**

To collaborate effectively as a team and manage our project codebase, we have adopted the following collaboration techniques and tools:

- Git: We have utilized Git as a version control system to track changes, manage branches, and collaborate seamlessly.

- GitHub: Our team has leveraged GitHub as a remote repository, enabling us to collaborate, review code, and manage issues efficiently.

**3 Next Step**

We have completed the initial process of building the project, with learning lots of related knowledge and technologies. This gives us a good foundation for subsequent development.

In the coming weeks, we aim to continue refining these skills, focusing on integrating the different components of our project. Regular progress updates will be provided to ensure transparency and to facilitate timely feedback.

We will try to keep following the scheduled milestone plan as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| ***Tasks*** | | ***Estimated completion time*** | ***Estimated number of***  ***learning hours*** |
| 1 | Data Collection and Preprocessing | 1 week | 15 |
| 2 | Feature Selection | 1 week | 20 |
| 3 | Model Development | 2 weeks | 80 |
| 4 | Model Evaluation | 1 week | 20 |
| 5 | Graphical Representation | 1 weeks | 35 |
| 6 | User Interface Development | 2 weeks | 35 |
| 7 | Real-time Data Integration | 1 week | 40 |
| 8 | System Testing | 1 week | 15 |
| 9 | Deployment & Maintenance | 1 week | 20 |
| 10 | Feature Selection | 1 week | 20 |

**4 Conclusion**

This interim report showcases the computer science techniques and technologies we have learned and applied to our final year group project. By acquiring expertise in front-end development, algorithmic techniques, back-end development, data management, server deployment, and collaborative tools, we have made significant progress towards completing our project successfully. The skills we have acquired will not only contribute to our current project but also serve as a foundation for our future careers in the field of computer science.