Non-Binary Market Sentiment Prediction Based on the Dynamic Analysis of the News

System Requirements

The following is the set of functional and non-functional requirements for our project:

1. The system should be divided into two parts: the frontend and the backend which operate independently of each other.
2. The frontend displays information in a readable manner across desktop, tablets, and phones.
3. The information displayed to the user is clear, concise, and easy to understand.
4. The User should be able to see information about the company
5. The User should be able to filter companies out by 1 single field, displaying only relevant information
6. For each company, the User should be able to input a value to see predicted according to our model.
7. The information should be displayed to the User in under 5 seconds of accessing the website
8. The information processed in the backend should be updated each day
9. Users should not be able to manipulate data in the backend
10. Financial news should be divided field, sentiment, and source reliability
11. A continuous variable should be returned from the NLP analysis in the backend
12. The backend should hold information pertaining to each company and send it to the frontend
13. The NLP model used should be accurate at least 75% of the time to ensure quality in our system
14. The scraping process should run with 0 errors, and return accurate categories for the backend
15. The User should get information related to predictions to the nearest hundredth

The system requirements were gathered and elicited by interviewing each other while we assumed the roles of potential user and high-level project manage. By putting ourselves in our users' shoes, we were able to think of requirements that would make it easiest for the user to

navigate our system. For instance, we believe that users want autonomy over how they view the information as well as clarity, so having a filtering tool and responsive design would help with this. When we looked from a high-level perspective, we saw that we wanted our frontend and backend to work independently as they accomplish two different goals: the frontend is to display data and the backend is to process data. Seeing this, we put a lot of emphasis on creating specific requirements that helped divide these processes for their respective goals. This can be seen above, as we have no overlap between the two different subsystems' requirements.