LetsEat Technical Summary

Technical Innovation

There are multiple technical components that go into creating LetsEat. There is the frontend portion made in REACT, the RestAPI made in FastAPI, the MySQL database, the web scraper and YelpAPI connection, and finally the machine learning model with Scikit Learn/PyTorch. These components come together to create a full-stack web application that scrapes and stores information from Yelp and processes that information along with user information in the machine learning model to create a recommendation for the user. The most novel part of LetsEat is its ability to create a recommendation for a user based on their preferences and how they answer questions about where they want to eat. With our project, the emphasis is on the recommendation generated as that is something that has not been done before. Comparing this to websites that simply act as databases for restaurants, we are trying to make it easier to find a website that matches what your needs are at the moment with the least complications possible.

Key Objectives

The key objectives of the project is to create a full-stack web application that has a simple, well-designed frontend that is easy to navigate, a functional backend that bridges the components and the data flowing between them, a database that stores user information to be used in the machine learning model, a machine learning model that is able to give good recommendations to the user based on their preferences, and have an application that is secure. Some questions to answer are:

- How can we scale the project once we have the components working?
- How can we best secure the data we are processing?
- How can we better the machine learning model as we get more data?
- How can we move data between the components quickly and accurately?
- How can we attract/market the application to gain a large user-base?

Technical Feasibility

The tools we are using to build the project are: REACT, FastAPI, Scikit Learn/PyTorch, MySQL, Yelp API, and Selenium Web scraper. Each of these technologies have a lot of documentation and support for them online and have been put together by other projects, so we know that they can be used in conjunction with one another.

Costs,RIsk and Risk Mitigation

Our project only relies on software and not hardware. We anticipate having a working machine learning model that gives low-confidence scores by the end of December with a fully working model a month or two after that. In total, the project should have 700+ lines of code with the backend and machine learning taking up about 300+ lines of code.