Progress Report: Shark Tank Assistant

Version 1.1 approved

Leann Blanc, Colin Chenoweth, Aditi Mukkara, Sahil Patel, & Mehlam Saifudeen

October 20, 2023

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Initial version	9/5/23		1.0
Version 1.1	10/20/23	Progress Report 1st half	1.1

1. Background

1.1 Purpose and Audience

This project proposal explains the various requirements for the first release of the Shark Tank Assistant Application. The user base for this application is wide, primarily targeting sharks and applicants on Shark Tank. However, the user base can also include users of other shows such as Dragon's Den, a spinoff of Shark Tank in Canada and Poland, Money Tigers in Japan, and Tu Oportunidad (Your Chance) in Spain. The document was created to guide the development of the application with the goal being to develop each of the listed requirements. For the users, this document provides information on the functionality and usability of the application as well as progress thus far.

1.2 Product Scope

With entrepreneurial spirit rising around the world, fostering creativity and encouraging individuals in their aspirations is more important than ever. One such way is Shark Tank which provides seed funding for individual companies. Our application will allow for users to tailor their Shark Tank application based on their specific company and product details including but not limited to industry, size, and previous sales. Benefits of this app include allowing users to better obtain funding and valuation deals, instead of potentially less meaningful applications. In addition, our application can be used by sharks who are investing funds into the companies to see how their previous investments match up to new products being pitched.

2. Technical Project Requirements

2.1 Product Perspective and Goal

- Create a program that provides SharkTank contestants with feedback on their proposal and product, including but not limited to potentially revealing whether/which a shark will accept their offer, or whether their product will succeed in the market.
- Utilize datasets and past episode transcripts to train the model.
- Have a user interface that is a pseudo Shark Tank application to get information on the product and proposal, and return useful feedback.
- Measure "usefulness" by comparing feedback to past proposals/product outcomes.

2.2 Product Functions

2.2.1 Front End

The first page will include graphs, tables, and other interactive visualizations to provide the user with general information regarding previous Shark Tank contestants. For example, the user is able to select filters such as the industry they are interested in or the investor they are targeting.

Accordingly, the page will display information such as the average investment amount per investor per industry. This page will also include interactive elements such as pie charts and bar graphs.

The second page will include an application for the user to fill out. This application will include specific aspects of the actual 17 page Shark Tank application. Once the user submits the application, the program will tell them if their pitch will be successful or not on the Shark Tank show. Based on our progress, we hope to expand this so that the user can also receive feedback on which shark is most likely to invest in their idea or whether their original valuation is too high or too low.

2.2.2 Back End

Feature Extraction:
The goal is to create an application-like feature set. It will be easy to expand from simple facts about the proposal/product, to eventually include more complex data like the transcript of an episode or the description of a product/demonstration. We can also use NLP to extract useful features from the transcript or application text. We can either try to create our own metrics for relevance (such as highest frequency words or frequency of chosen buzz words) or use NLP libraries to help gather semantic meaning and transform that into a usable feature.

Machine Learning Model(s):
Either use one multi-class classification model or multiple single-class classification models. We can use something complex like support vector machines or neural networks (which could reduce the amount of the feature transformation required) or more simple classification models such as logistic regression. Depending on the complexity and difficulty of creating a single accurate/precise model, it may be feasible to also compare how well different models work for this problem.

2.3 Dataset

For general statistics regarding past Shark Tank investments we will use this open source dataset providing details on episodes. This dataset includes information from the contestant's application such as their original amount ask, the original equity ask, their gender, and their industry. It also includes information regarding how successful the contestant was on the show using measures such as whether or not they got the deal, how mean each investor invested, and the deal valuation.

Shark Tank US Dataset

For further information regarding rhetoric or negotiations after our developments with this data, we will look into other data sources such as Shark Tank show transcripts or data from other similar shows across the world.

2.4 Code

https://github.com/ColinChenoweth/Shark-Tank-Assistant

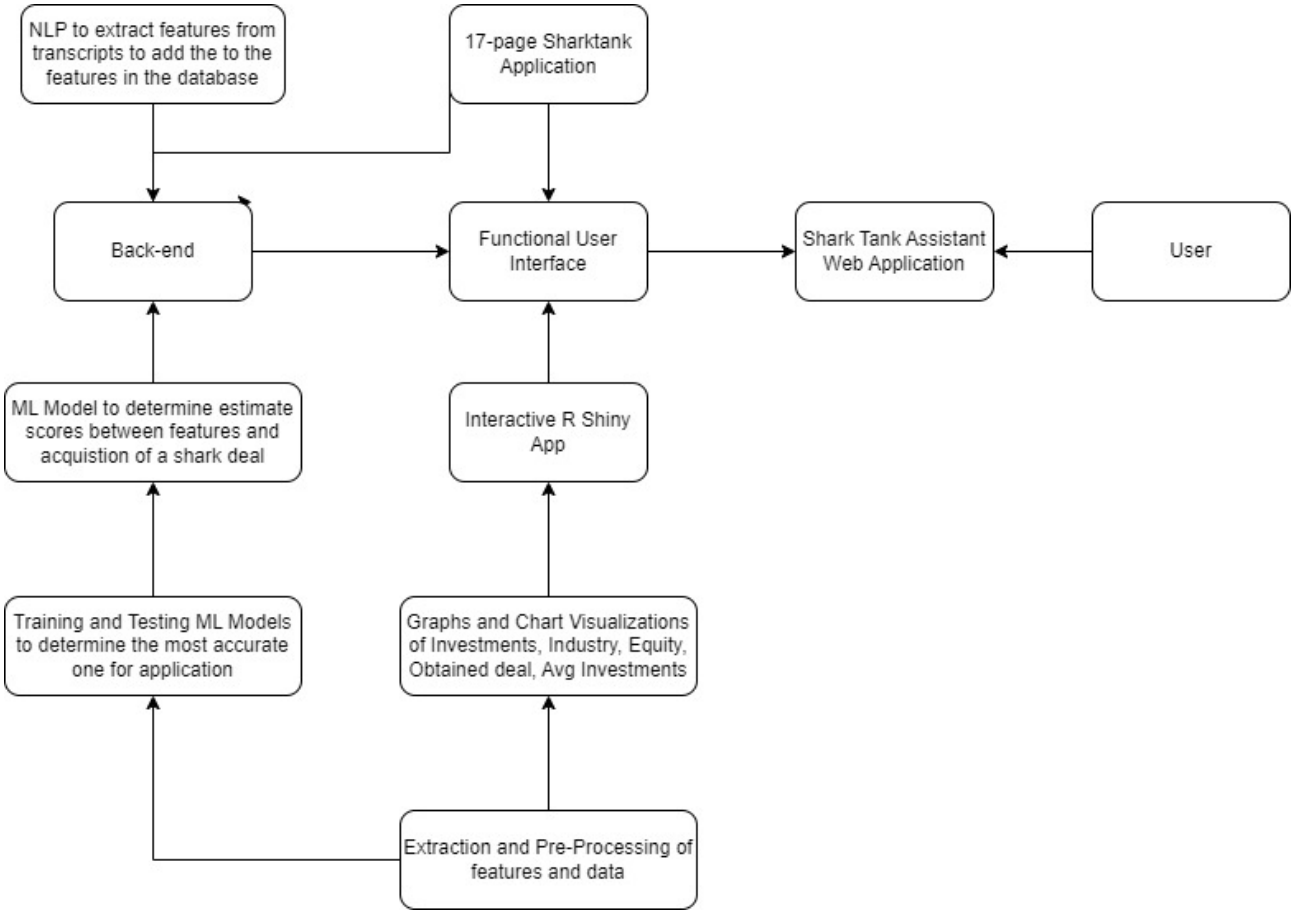
3. Management Plan

3.1 Completed Work

To start off, we were able to obtain a dataset that included product information, valuation, final deal, which sharks invested, among other variables which we used to train the model and create a variety of visualizations. We created a variety of visualizations that a user can use to guide them if they do not actually want to run the model. To start off, we have an interactive pie chart that allows the user to see a product breakdown by industry of all pitches and one of pitches that sharks invested in. We created a bar chart showing the Original Offered Equity vs. the Total Deal Equity Per Industry on Average as well as a bar chart to show the breakdown of the pitchers' gender per industry. Additionally, we created an interactive shiny interface that allows users to select an industry they are interested in and choose whether they are interested in seeing a bar chart of original ask amount, valuation requested, total deal amount, or deal valuation for products in that industry from past episodes. This can give users an overall idea of what they can expect before filling out an application for Shark Tank or using our model to determine a valuation for their company.

In addition to this, we obtained the transcripts of close to 240 episodes from Hulu which can be used to build an NLP that can help us expand on the current dataset, allowing for more training data to be used to make the model more precise and accurate. We also extracted the most important features that can help both us and the user determine whether their business idea would obtain an investment from the shark or not. These features included, industry, pitchers average age, gender, state, city, original asked amount, and original amount offered and equity offered. Using the data we used 7 different Machine Learning models including Random Forest, Naive Bayes, Gradient Boost, SVM, amongst others. Combining the data in the features and the ML models we were able to obtain estimate scores of whether a SharkTank deal was obtained or not based on the features mentioned. The accuracy, precision, and F1 score of the models were also included to see which one would perform best.

A design of the front-end of the application was developed which incorporated all the main features that are intended to be included in the front-end of the web application. The front end will interact with the backend model that will be used to provide feedback on a user's pitch idea using the 17-page long SharkTank application. The front-end will also contain the charts and graphs developed in the R shiny app which will be interactive and provide a user-friendly environment.



3.2 Work To Be Done

For our final project report and submission we plan to have selected the best model amongst the ones that we mentioned above, along with selecting metrics in its function to make it as accurate as possible. In addition to this, we plan to complete a full-fledged front-end design of the application that will present a variety of graphs and charts highlighting features such as industry wise investments, success rate by industry, as well as information for how the sharks invest by industry. This front-end will allow users to get a broad idea of how products do in the industry without needing to run the model. This will also allow users to gain insight prior to creating their own pitch. If they want more specific insight regarding the product they have come up with, they can run our model using the 17-page Shark Tank application. The application would give the user an idea of what to expect in terms of investment amount, valuation, and individual shark interest based on their current sales, product market and net profits. There will also be suggestions for users to either reduce their ask or whether their valuation is lower than what the product market or any previous similar Shark Tank investments suggests. We plan to incorporate two different forms of machine learning models/techniques in order to implement the above.

3.3 Timeline

- [x] ~~Week 1: Set up environment~~
- [x] ~~Week 2: Submit Project Proposal~~
- [x] ~~Week 3: Find dataset and episode transcripts~~
- [x] ~~Week 4: Research and decide on best models~~
- [x] ~~Week 5: Train/debug various models for basic classification~~
- [x] ~~Week 7: Plan user interface and complete the implementation of first look of UI~~
- [x] ~~Week 8: Submit Progress Report and Demo Video~~
- [ ] Week 9: Select best Model in backend for Individualized Pitch Feedback
- [ ] Week 10: Expand more on the Model to Improve Accuracy
- [ ] Week 11: Train/debug model for all feedback
- [ ] Week 12: Combine model and user interface
- [ ] Week 13: Debug and combine all the code and make poster for intersection
- [ ] Week 14: Practice presentation
- [ ] Week 15: Submit Final Project Code and Report

3.4 Individual Contributions

A. Leann
    a. R charts and Visualizations (Interactive Pie Charts by Industry)
    b. Shiny App Front-end and Visualizations (Allows user to select an industry and specific value they want visualize)
B. Aditi
    a. R charts and Visualizations
        i. Original Offered Equity vs. Total Deal Equity
        ii. Pitcher's Gender by Industry
C. Sahil
    a. Building Machine Learning Models
        i. Logistic Regression
        ii. Decision Tree
        iii. Random Forest
        iv. Support-Vector Machine
    b. Website Front-end Design
D. Colin
    a. Building Machine Learning Models
        i. Naive-Bayes
        ii. Neural Network
    b. Testing Accuracy
    c. Preprocessing of transcripts
E. Mehlam
    a. Building Machine Learning Models

    i. Gradient Boost
b. Features Data Extraction
c. Software Design Schema