## Overview

This project tried to predict whether applicants will be successful if funded by the Alphabet Soup Co. Alphabet Soup Co. wants to provide funding to companies, but it needs to know in advance whether it will be successful or not. Money is on the line here and we would not want it to be wasted.

For this I will create a neural network by using Data Manipulation, creating training and testing sets, and finally analysing my models that I have created.

## Results

- \*\*Data Processing\*\*

- To clean the data I removed the EIN column since it has no value to the model.

- The variables being considered for my model are as follows: ‘NAME’, 'STATUS', 'ASK\_AMT', 'IS\_SUCCESSFUL', 'APPLICATION\_TYPE', 'CLASSIFICATION', 'USE\_CASE', 'ORGANIZATION', 'INCOME\_AMT'.

- I used cut off for the ‘Name’ , 'APPLICATION\_TYPE', ‘CLASSIFICATION' and group those as ‘others’ for each columns.

- Also used get dummies for the categorical variables.

- Removed outliers from the ‘ASK\_AMT’ and replace with Upper /Lower bound values.

- My Dependent variable is "IS\_SUCCESFUL" since we want to try to predict this with high accuracy.

- \*\*Compiling, Training, and Evaluating the Model\*\*

\*\*Attempt #1\*\*

- 2 Hidden Layers

- 10 neurons (Layer1), 20 neurons (Layer2)

A screenshot of a computer

Description automatically generated

I tried to change my models in order to achieve more than 75% accuracy rate that is ~79%. I changed my features, Hidden Layers, and the number of neurons in order to achieve this.

## Summary

On Average my models kept around 79% accuracy score which is decent considering it was an improvement. My recommendation to improve this model would be to find better features to help explain what determines "IS\_SUCCESFUL" such as more indepth knowledge of the other associates/ firms being funded. At the end of the day, knowledge is power and if we had more in-depth data between all these applications, we can create a better model.