**Report on Neural Network Model for Alphabet Soup**

**Overview of the Analysis**

The purpose of this analysis was to develop a deep learning model capable of predicting whether organizations funded by Alphabet Soup would be successful based on various features derived from their application data. The goal was to achieve a predictive accuracy of over 75%, optimizing the model to enhance its performance.

**Results**

**Data Preprocessing**

* **Target Variable(s):**
  + IS\_SUCCESSFUL: This is the target variable for the model, indicating whether the funding was successful (1) or not (0).
* **Feature Variable(s):**
  + The features selected for the model include the processed and encoded categorical variables such as:
    - APPLICATION\_TYPE
    - AFFILIATION
    - CLASSIFICATION
    - USE\_CASE
    - ORGANIZATION
    - STATUS
    - INCOME\_AMT
    - SPECIAL\_CONSIDERATIONS
    - The ASK\_AMT column was also used as a feature.
* **Removed Variable(s):**
  + EIN: This unique identifier was removed because it does not provide predictive value.
  + NAME: This variable was also removed since it’s not relevant for predicting success and could lead to overfitting due to its high cardinality.

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

* **Neurons, Layers, and Activation Functions:**
  + **Input Layer:**
    - The input layer had the same number of neurons as there were features in the dataset.
  + **First Hidden Layer:**
    - Number of neurons: 80
    - Activation function: ReLU (Rectified Linear Unit) was chosen due to its efficiency in handling non-linear relationships.
  + **Second Hidden Layer:**
    - Number of neurons: 30
    - Activation function: ReLU
  + **Output Layer:**
    - Number of neurons: 1
    - Activation function: Sigmoid was used since it’s appropriate for binary classification.
* **Model Performance:**
  + The initial model did not achieve the target accuracy of 75%.
* **Steps to Increase Performance:**
  + **Increased Neurons and Layers:** Additional hidden layers and neurons were added to provide the model with more complexity.
  + **Activation Functions:** Different activation functions (such as tanh) were experimented with to determine if they offered better performance.
  + **Data Adjustments:** Binning rare categories and scaling the data were techniques used to improve the model’s ability to generalize.
  + **Epochs:** The number of epochs was adjusted to ensure the model had enough time to learn, while also preventing overfitting.

**Summary**

* **Overall Results:**
  + The final model achieved an accuracy of approximately 72% after several optimization efforts. Despite multiple attempts to optimize the model, achieving the target accuracy remained challenging.
* **Recommendation:**
  + Given the current results, an alternative model such as a Random Forest classifier or Gradient Boosting Machine might be more effective for this classification problem. These models can handle a large number of features and can often achieve better performance with less need for complex hyperparameter tuning.