Part 1: Research Question

1. Can positive or negative sentiment be predicted for future customers using a neural network and natural language processing (NLP)?

2. The goal of this analysis is to build a neural network that can predict positive or negative sentiment of future customers using previous customer reviews. This will lead to greater understanding of which products customers prefer and can help with future marketing decisions.

3. A recurrent neural network (RNN) will be used for this analysis. RNN considers that the next word depends on the previous words in the review. This is a many to one RNN. There are many inputs that lead to a single output, in this analysis, if the sentiment is positive or negative. (Saeed, 2021) The model will be built using a sequential neural network utilizing the TensorFlow and Keras libraries. A sequential model is used when the data is not 'independently and identically distributed' (Lendave, 2021). The data is not independent because the order of the words matter. If the order was changed, the meaning changes.

Part 2: Data Preparation

The proposed word embedding length was found using the formula: 4th root of the number of categories. (Introducting TensorFlow Feature Columns, 2017) In this analysis, every word in the vocabulary is its own category. Based on this formula, a word embedding length of 8 dimension will be used.

2. Tokenization is the process of separating text into smaller units known as tokens. For this analysis, the token size is word. These tokens are used to create the vocabulary, the set of unique tokens. The vocabulary is then as the features when vectorizing the reviews. (Aravindpai, 2020) This process is necessary because Keras cannot process text.

3. Neural networks requires that all inputs be the same size and shapes. This is accomplished by adding zeros to the beginning or ending of the sequence. (Caner, 2020) Here, zeroes are added to the end of the sequence.

4. There are two categories of sentiment: 1 – positive, 2 – negative. The sigmoid activation function will be used in the final dense layer because the outputs are binary.

5. The steps used to prepare the data are as follows:

- Import packages
- Load datasets
- Inspect datasets size
- Merge the datasets into one DataFrame
- Remove special characters and digits

- Remove stop words
- Create a vocabulary
- Find the embedding size
- Split the data into training and test sets using a 70/30 split
- Tokenize the DataFrame
- Pad the reviews to standardize the lengths

PART III: Network Architecture

2. The first layer is the Embedding layer. This layer converts each word to a fixed length vector of a stated size. The parameters used in the layer are the vocabulary size, embedding dimension, and the maximum length of each review. The second layer is GlobalAveragePooling1D() which flattens the vector from three dimensions to two dimension. The third layer is a Dense layer with 6 nodes and the activation function relu. The final layer is a Dense layer with 1 node and activation function sigmoid. The sigmoid activation function was chosen because the output is binary.

Activation Functions: relu – chosen because it only returns positive numbers or 0. Sigmoid – chosen because the outcome is binary.
Number of Nodes per Layer – The nodes of the first layer are equal to the number of words in the vocabulary. The second node of 10 is a number that is between the first layer and the last layer. The third node of 5 was chosen because it reduces the number of nodes from the previous layer but is more than the final layer. The last layer has 1 node since the output of the model is binary.
Loss Function – The loss function used in the model was Binary Crossentropy. It was chosen because there are two possible outputs.
Optimizer – the Adaptive Moment Estimation, Adam, optimizer was chosen because it is efficient and requires less memory. The Adam optimizer combines the strength of gradient descent with momentum and root means square propagation. (Intuition of Adam Optimizer, 2020)
Stopping Criteria – Early stopping is used during the fitting of the model. The model will stop based on validation loss. Validation loss shows how well the model fits new data.
Evaluation Metric – Accuracy is the percent of correct predictions to the set of targets. It was chosen because of its ease of interpretation.

Stopping criteria is used to prevent overfitting. It does this by stopping the training of the model as soon as the validation error reaches a minimum. The patience parameter is used to ensure that the training does not stop at a local minimum but reaches the true minimum.

The model accuracy improves over the first 6 epochs. After this point the model starts to overfit the training data. While having a large accuracy percentage may seem good, it can be a sign that the model will not handle new data well. To reduce the chance of overfitting, early stopping was used.

The predictive accuracy of the model is 75.67%. The model can predict customer sentiment about 75% of the time. This is high accuracy.

This model is very functional. It has a high predictive accuracy. One recommendation would be to try the model with more layers. This might improve the accuracy.

A recommend course of action is to use the model to reach out to customers who are predicted to have poor sentiment towards the product. The company can use this opportunity to correct any problems with the product or gather feedback on how to improve. This will help to improve sentiments towards the company if not the product.

Works Cited

Aravindpai. (2020, May 26). *What is Tokenization in NLP? Here's All You Need To Know*. Retrieved March 2022, from Analytics Vidhya: https://www.analyticsvidhya.com/blog/2020/05/what-is-tokenization-nlp/

Caner. (2020, April 2). *Padding for NPL*. Retrieved March 2022, from Medium: https://medium.com/@canerkilinc/padding-for-nlp-7dd8598c916a

Introducting TensorFlow Feature Columns. (2017, November 20). Retrieved March 2022, from Google Developers: https://developers.googleblog.com/2017/11/introducing-tensorflow-feature-columns.html

Intuition of Adam Optimizer. (2020, October 24). Retrieved March 2022, from Geeks for Geeks: https://www.geeksforgeeks.org/intuition-of-adam-optimizer/

Lendave, V. (2021, November 17). *A Tutorial on Sequential achine Learning*. Retrieved March 2022, from Analytics India Magizine: https://analyticsindiamag.com/a-tutorial-on-sequential-machine-learning/

Saeed, M. (2021, September 24). *An Introduction to Recurrent Neural Networks and the Math That Powers Them*. Retrieved March 2022, from Machine Learning Mastery: https://machinelearningmastery.com/an-introduction-to-recurrent-neural-networks-and-the-math-that-powers-them/