

Summary

The objective of this project was to implement a neural network for classifying customer reviews of microwave products from Amazon as either positive or negative. The model utilized a simple recurrent neural network (RNN) with specific modifications, which are discussed in detail in the Network Implementation section. Overall, the project was successful and achieved its intended functionality.

Dataset

The dataset used is repurposed from the Spring 2024 semester and contains microwave reviews scraped from Amazon. The dataset contains 7000 entries. Each entry contained two features: the review rating, ranging from one to five stars, and the text portion of the review. The preprocessing phase of the project consisted of two steps. First, all the three-star reviews were removed from the dataset, as they were neutral and counterproductive to the goal of classifying as either positive or negative. Second, the remaining reviews were reclassified into two categories: positive and negative. Reviews with four or five stars were categorized as positive, while those with one or two stars were labeled as negative. The final data processing stage converted the text portion of the reviews to embedded word2vec matrices and split into the following datasets; 60% for training, 20% for cross-validation, and the final 20% for testing.

Network Implementation

A simple recurrent neural network was used for this project, unrolling the network into layers and using outputs from previous layers as inputs for upcoming layers. The structure of each unrolled layer consists of an input layer, the size of which is determined by embedded matrix, one hidden layer consisting of 128 neurons, and 1 output layer with 2 neurons, each representing the positive or negative

outcome. This model used the Adam optimizer and Categorical Cross Entropy to calculate loss. Outputs from all unrolled layers are fed into a separate feed-forward neural network for final classification. The system implements early stopping if cross-validation entropy loss doesn't improve within a predefined number of epochs, which is known as the Patience of the system.

Results

After extensive testing, the final testing accuracy averages to near 80%, with cross-validation at 83% and training at 79%. While these results don't necessarily improve upon last semester's results, the model does a reasonably good job of identifying positive and negative product reviews and generally speaking is successful at its defined task.

Requirements

- Clean Review Dataset
- Python - Version 3.9.12
- Python Libraries

Library Name	Version
Numpy	1.22.4
Pandas	1.4.2
Tensorflow	2.10.0
Gensim	4.4.3
Scipy	3.1