NLP ANALYSIS OF AMAZON VIDEO GAME REVIEWS

Chuck Tucker

The Problem

- There are benefits to quickly classifying text as positive or negative
  - Identifying unsatisfied/satisfied customers to influence decisions
  - Generate predictions based on the satisfaction level for certain products
  - Create action plans for improving products based on reviews

The Data

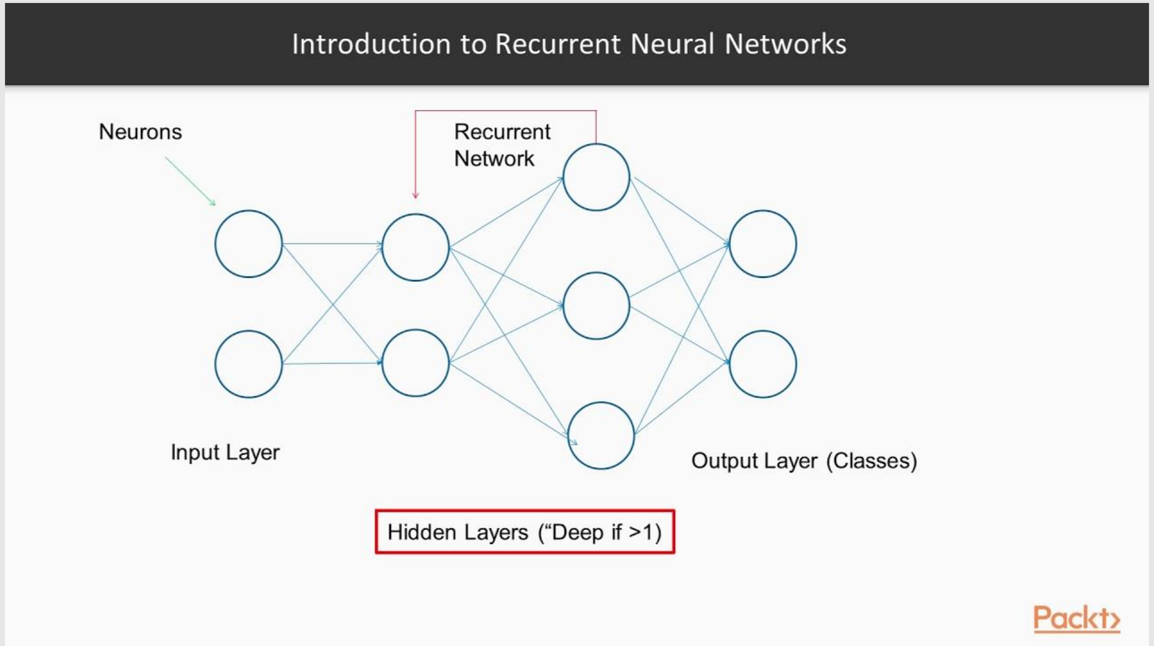
- The data came from an NLP study and are available online
- The selected subset of the Amazon data were video game reviews, filtered to only include games for which there were at least 5 reviews

- Citation:
  - Justifying recommendations using distantly-labeled reviews and fined-grained aspects Jianmo Ni, Jiacheng Li, Julian McAuley Empirical Methods in Natural Language Processing (EMNLP), 2019 https://nijianmo.github.io/amazon/index.html#files

Approach

- Exploratory Analysis
  - Read in the data file
  - Explore the data structure
  - Remove any missing values
  - Visualize summary information like word counts,

- Classification Model
  - Try Flair NLP pre-trained model (trained on IMDB reviews)
  - If Flair pre-trained model is inadequate, train a model using a recurrent neural network



Exploratory Results

- Initial analysis of the data found many emojis and odd characters among the text data
- This was dealt with by creating a function to remove all special characters
  - It would not be practical to include these in the data, as variations and customizations meant most of the emojis only appeared once in the data

Examples

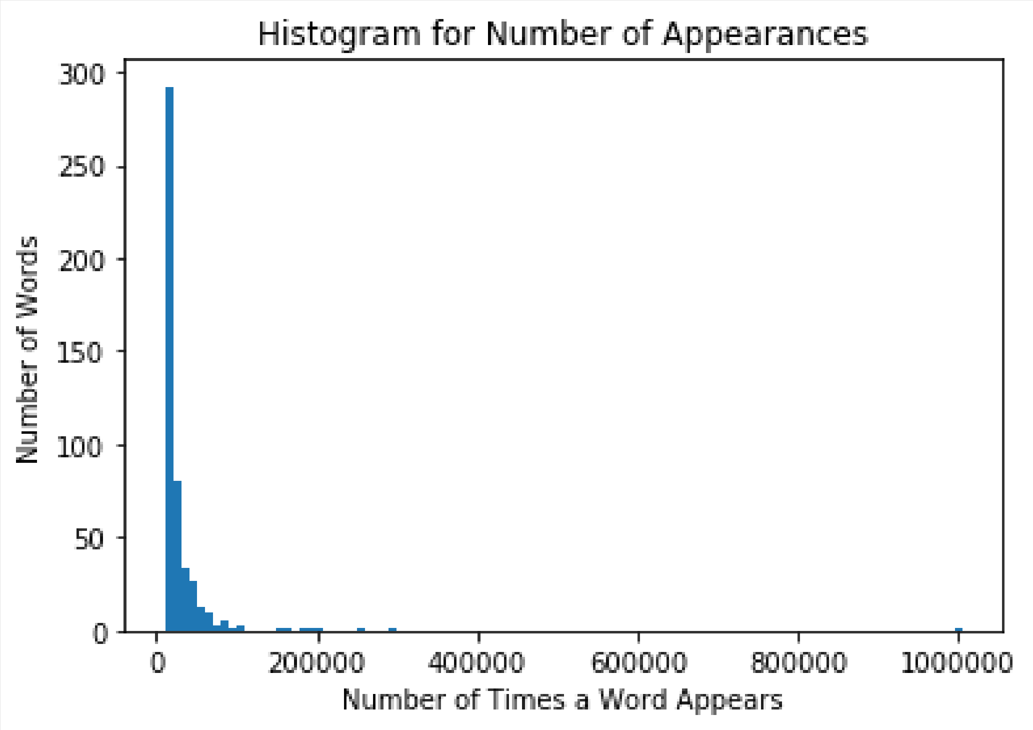
\m/(>_<)\m/

\(^0^)/

\(^u^)/

Word Counts

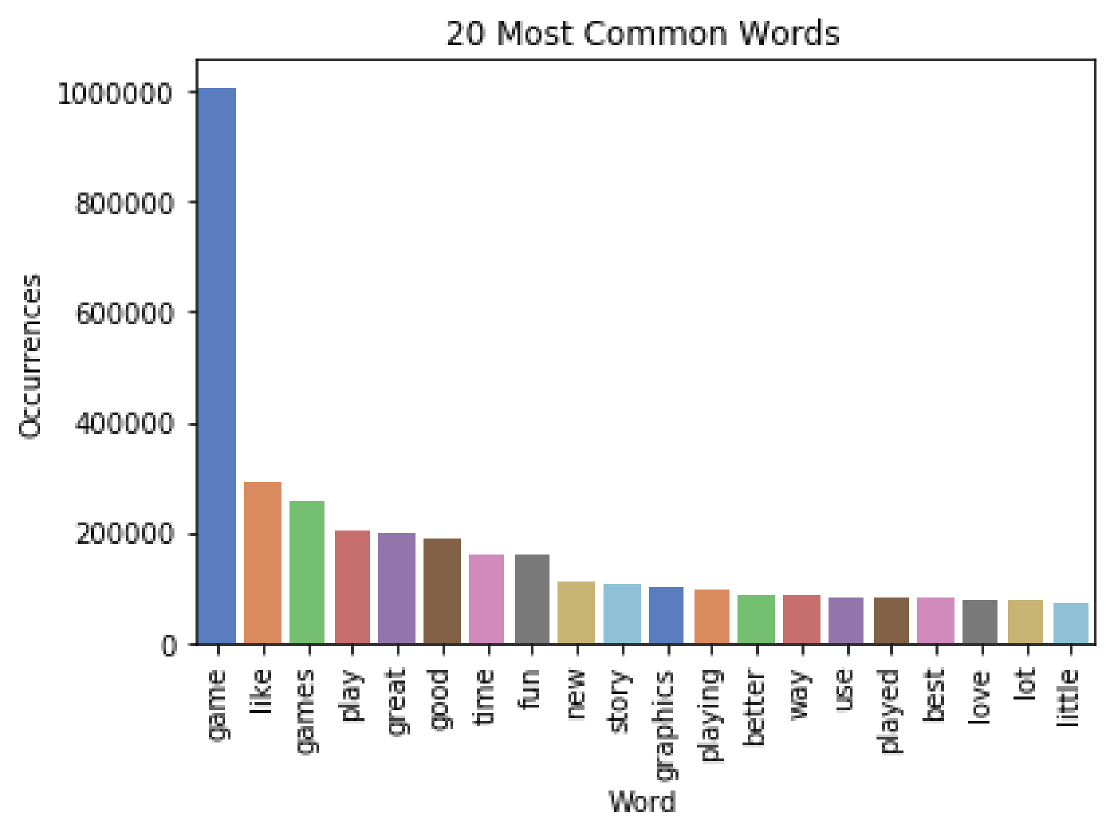
- Analysis found a total of 26,737,141 clean words in the reviews
- There were only 262,853 unique words
- Further analysis showed that about 60% of words were only used once, meaning
- This means that only about 105,141 words accounted for 99% of the total words appearing in the reviews



Most words appeared infrequently

Most Common Words

- The word ‘game’ accounted for nearly 4% of all the words in the reviews, and this was by far the most encountered word



Word Cloud – top words/phrases

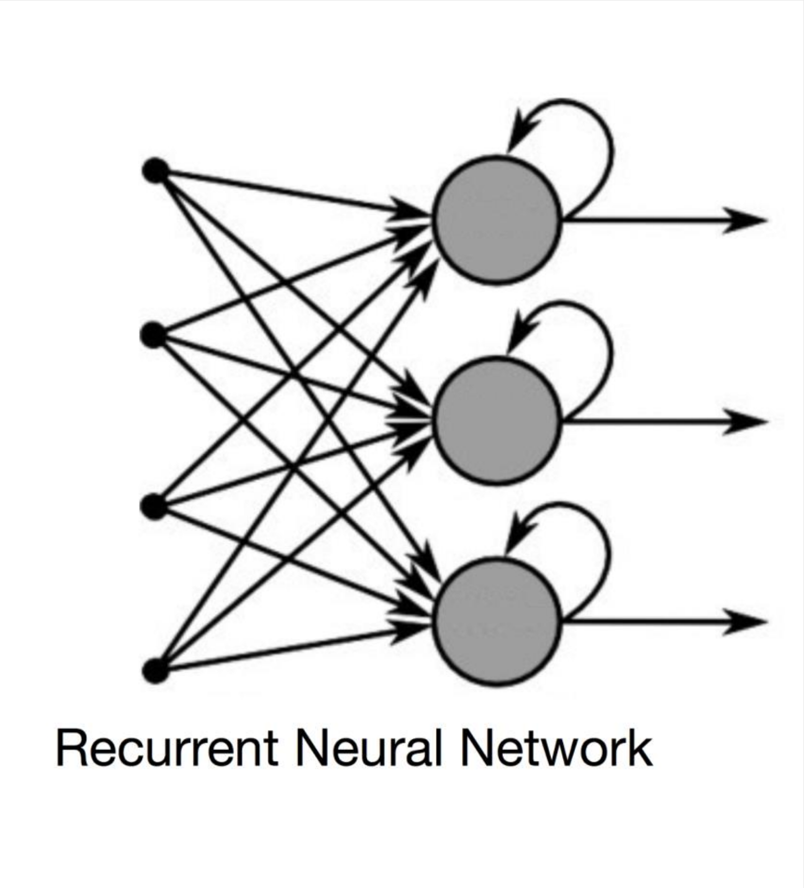
Sentiment Analysis

- Step 1: Flair Pre-Trained Model
  - Poor performance:
    - Accuracy 76% (not bad)
    - Negative precision: 28% (horrible)
    - Negative F1 score: 41% (really bad)
    - Positive precision: 96% (great)
    - Positive F1 score: 85% (really good)

- Ok performance on predicting positive results, but very poor performance predicting negative results
- This is not good if you want to identify unsatisfied customers
- Why did this happen?

Interpreting Pre-Trained Results

- The Flair model was trained on movie reviews
- The data come from different sources, and the vocabularies used could be very different
- A custom model was trained using the video games data to try and achieve better results using a recurrent neural network

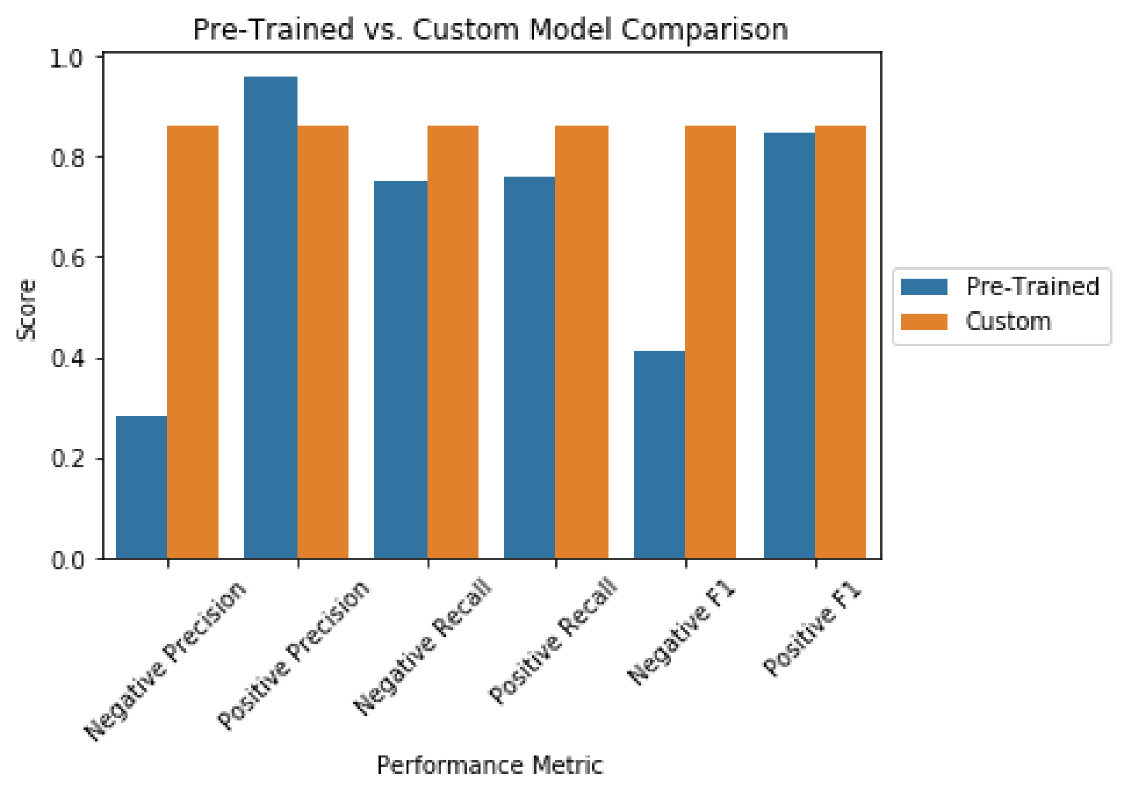


Custom Model Methods

- The custom model was created using LSTM to process the sequence of the text data
- 512 hidden layers were used with standard document embeddings (GloVe, news-forward-fast, news-backward-fast)
- A maximum of 10 epochs was chosen due to the length of time and computational resources necessary to train this model
- The model was fed the cleaned text with the special characters removed
- The data was downsampled to balance the classes prior to training
- An 80%, 10%, 10% split was used for training data, validation data, and test data
- The model ran for a total of about 13.5 hours on paid Google Colab Pro using a high RAM and GPU runtime

Custom Model Results

- The final best model produced F1 scores of about 85% for both classes, greatly surpassing the pre-trained model performance of 41% for the negative class
- Accuracy was about same at 76%, but the vast improvement on the negative class indicates this was a superior model
- Precision and recall were both also right at 86% for both classes, surpassing the pretrained model



Recommendations: NLP and Usage

- This analysis revealed that it is important to evaluate the performance of any pre-trained model prior to deploying
- It also demonstrated the importance of tailoring NLP models to the specific use case as much as possible
  - Yes it would be possible to overfit, but with text information, it is important to train on a set of features that is meaningful
- This model can be used to quickly assess a customer's sentiment regarding video games
  - This information could be used to help with support cases or create recommendations for other products based on sentiment