Predicting sentiment on Amazon review data

https://github.com/spitko/amazon-sentiment-analysis

About

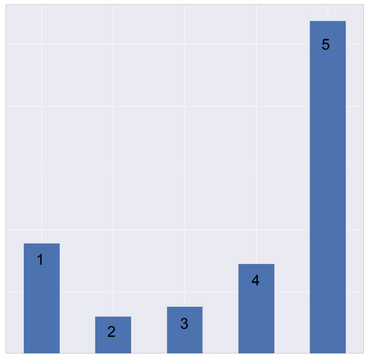
Sentiment analysis is contextual mining of text which identifies and extracts subjective information in source material. One common use case for this is classifying opinions as either positive or negative. Our goal with this project was to predict how a reviewer felt about their purchase based on their Amazon review by developing a model capable of classifying given text's polarity.

Data

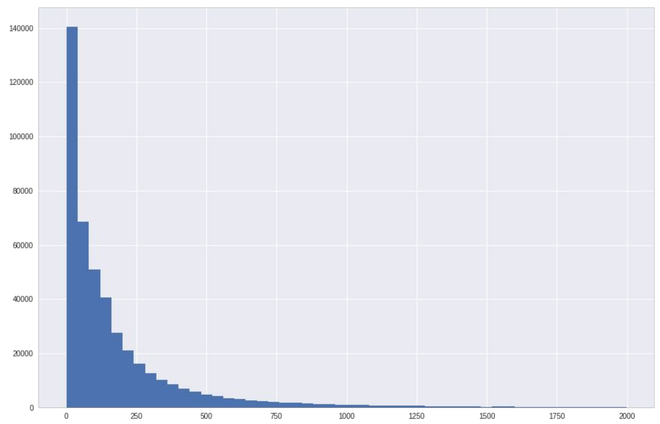
Our data consisted of two separate datasets found on Kaggle, which when combined contained data about over 480k different reviews left on different Amazon mobile phone product listings.

After filtering out reviews with over 2000 characters and neutral (3 star) reviews, we were left with 454k rows of data of which 74% were positive (4 or 5 stars).

Median number of words in a review was 18.

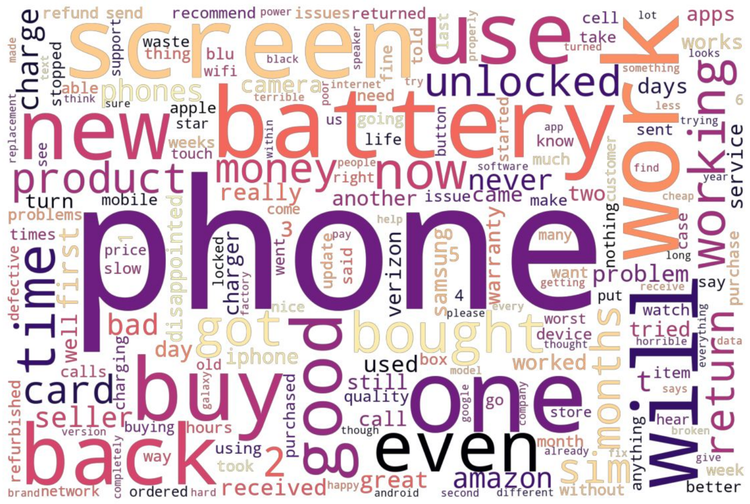


Amount of reviews by star rating

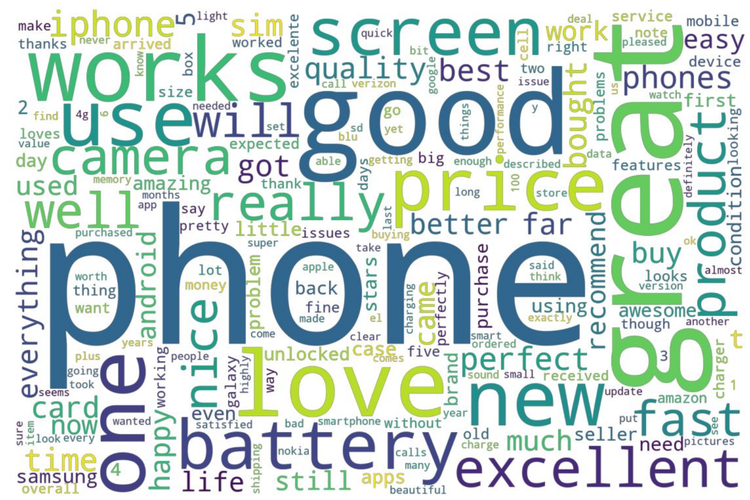


Character count of reviews

Most common words in negative reviews



Most common words in positive reviews



Preprocessing for ML

As the remaining data was fairly unbalanced with only 26% negative reviews, the first step was to downsample the positive reviews, leaving us with an equal amount of positive and negative samples.

From the remaining data a vocabulary was created containing 20000 most common words and assigning each a numeric index. The review texts were then replaced with sequences (lists) of word indexes.

Model

For machine learning, a convolutional neural network was utilised in TensorFlow using Keras as the frontend.

Given the large amount of data available, no pre-trained word embeddings were used and all embeddings were learned from scratch.

Model was trained on a Kaggle kernel, that included an Nvidia P100 GPU.

Conclusions

Our model achieved 96.8% accuracy on validation data. Considering our set goal to reach 85% accuracy, we can safely consider this project a success.

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding (Embedding)        (None, 200, 250)          5000000
_________________________________________________________________
dropout (Dropout)            (None, 200, 250)          0
_________________________________________________________________
conv1d (Conv1D)              (None, 196, 64)           80064
_________________________________________________________________
global_max_pooling1d (Global (None, 64)                0
_________________________________________________________________
dense (Dense)                (None, 1)                 65
=================================================================
Total params: 5,080,129
Trainable params: 5,080,129
Non-trainable params: 0
_________________________________________________________________
Train on 176061 samples, validate on 58687 samples
Epoch 1/3
176061/176061 - 118s - loss: 0.1794 - acc: 0.9319 - val_loss: 0.1225 - val_acc: 0.9578
Epoch 2/3
176061/176061 - 113s - loss: 0.0959 - acc: 0.9685 - val_loss: 0.1037 - val_acc: 0.9661
Epoch 3/3
176061/176061 - 108s - loss: 0.0635 - acc: 0.9810 - val_loss: 0.1002 - val_acc: 0.9681
Validation accuracy: 0.9680508375167847, loss: 0.10018643492896162
```

Model summary and training

Contact information

Samuel Pitko https://github.com/spitko
Helen Pung https://github.com/helenpung