ECE324: Rater Machine

Charles Kim
Andy Xie

Word Count: 1947
Permission to post video: yes
Permission to post final report: yes
Permission to post source code: yes

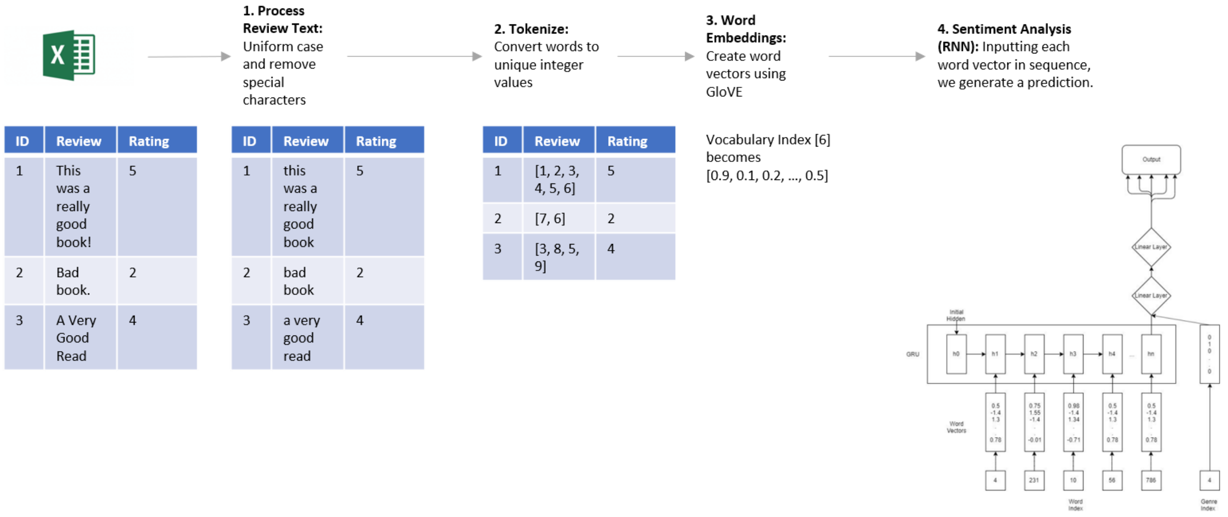
Introduction

The purpose of this project was to predict the rating of a book on a scale of 1-5 based on its reviews. Deep Learning mimics the neural pathways in a human brain; a key element of which is feature recognition. As such, a neural network with the capability to distinguish feature in book reviews which contribute to either negative or positive score is applicable for this task.

The rating scores which a rater leaves provides no indication to what exactly was good or bad about the book, whereas reviews contain the sentiment features of the reviewer. However, the rating scores still serve as a suitable representation of whether the reader enjoyed the book.

An immediate application of this neural network would be a recommendation system. Another use would be literature analysis, from both a financial and educational perspective. Authors and publishers can use the feedback from this neural network to act as a criteria for book releases. An analysis of the features which the neural network observes can also act as an indicator of what makes a book good which may foster new growth in the way authors write.

Illustration/Figure



1. Process Review Text: Uniform case and remove special characters

2. Tokenize: Convert words to unique integer values

3. Word Embeddings: Create word vectors using GloVE

4. Sentiment Analysis (RNN): Inputting each word vector in sequence, we generate a prediction.

ID	Review	Rating
1	This was a really good book!	5
2	Bad book.	2
3	A Very Good Read	4

ID	Review	Rating
1	this was a really good book	5
2	bad book	2
3	a very good read	4

ID	Review	Rating
1	[1, 2, 3, 4, 5, 6]	5
2	[7, 6]	2
3	[3, 8, 5, 9]	4

Vocabulary Index [6] becomes [0.9, 0.1, 0.2, ..., 0.5]

Background and Related Work

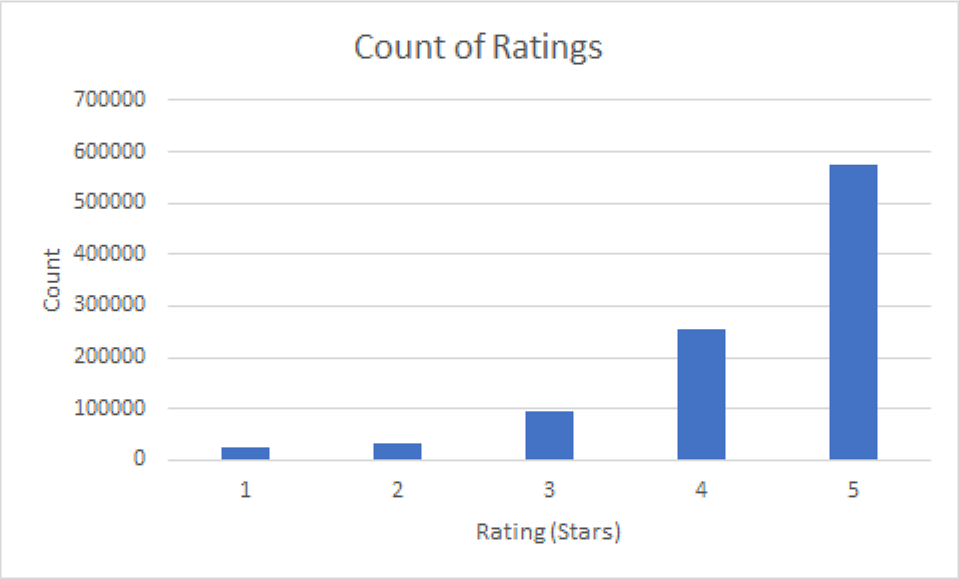
Natural language processing has been an evolving form of computation and Machine Learning has become the forefront methodology in this area. Our project is a type of sentiment analysis, which is defined as a process which identifies and categorizes opinions expressed in text. An example of a prior investigation of sentiment analysis is “Sentiment Analysis of Twitter Data Using Machine Learning Approaches and Semantic Analysis” by G.Gautam and D.Yadav. In their study, they were able to achieve accuracy rates of 80%-85% when categorizing a Twitter tweet as negative or positive.

Data Processing

The original dataset was obtained from Kaggle[1][2],. The size of the data set was 669.37MB and consisted of data as follows:

The dataset consisted of 982,267 unique reviews for 61,934 unique books.

The data used was the review text and the rating. The review text was processed so that reviews that consisted of irregular characters were removed, as they were found to usually be unusable/spam. Furthermore, reviews were limited to a character size of 2000 and a minimum size of 10. The review text was also tokenized into word indices, and converted to the necessary word vectors of size 100, using the Spacy and Torchtext libraries.

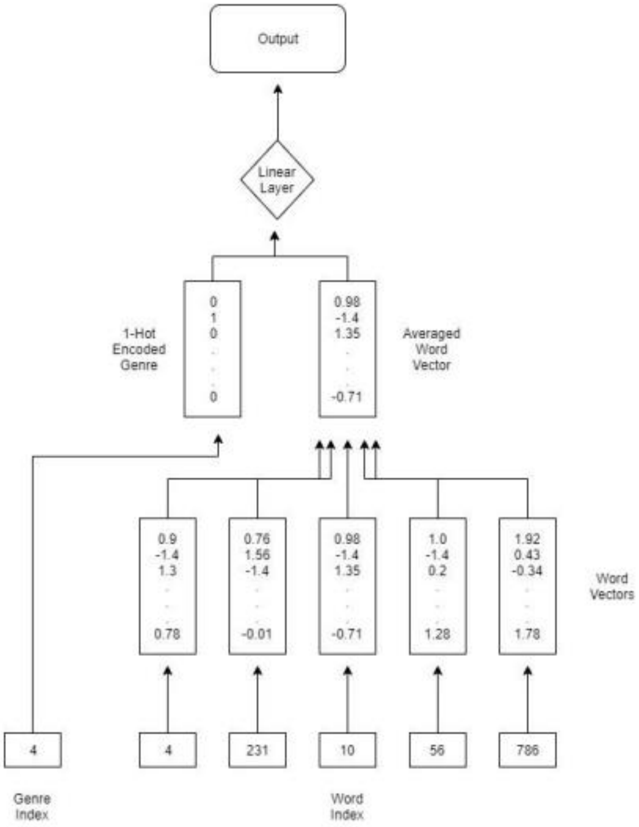


The ratings were converted from their original score to a either a 0 or 1 label, representing negative (1-2 stars) or a positive (3-5 stars) rating. This was because there was a smaller representation of ratings below 3 stars. Another dataset with labels of -1, 0, and 1 were created, representing negative (1-2 stars), neutral (3 stars), and positive (4-5 stars). A third dataset was also created in which, labels were also one-hot encoded to represent each of the 5 ratings.

Genres were attempted to be obtained using the Amazon Lookup API, however, the number of access calls was heavily limited. Therefore, a pre-existing dataset consisting of an Amazon book's metadata[2] was also used. These genres were then one-hot encoded, with books being classified as one of 52 different genres. An Alteryx workflow was created to perform the parsing of the metadata.

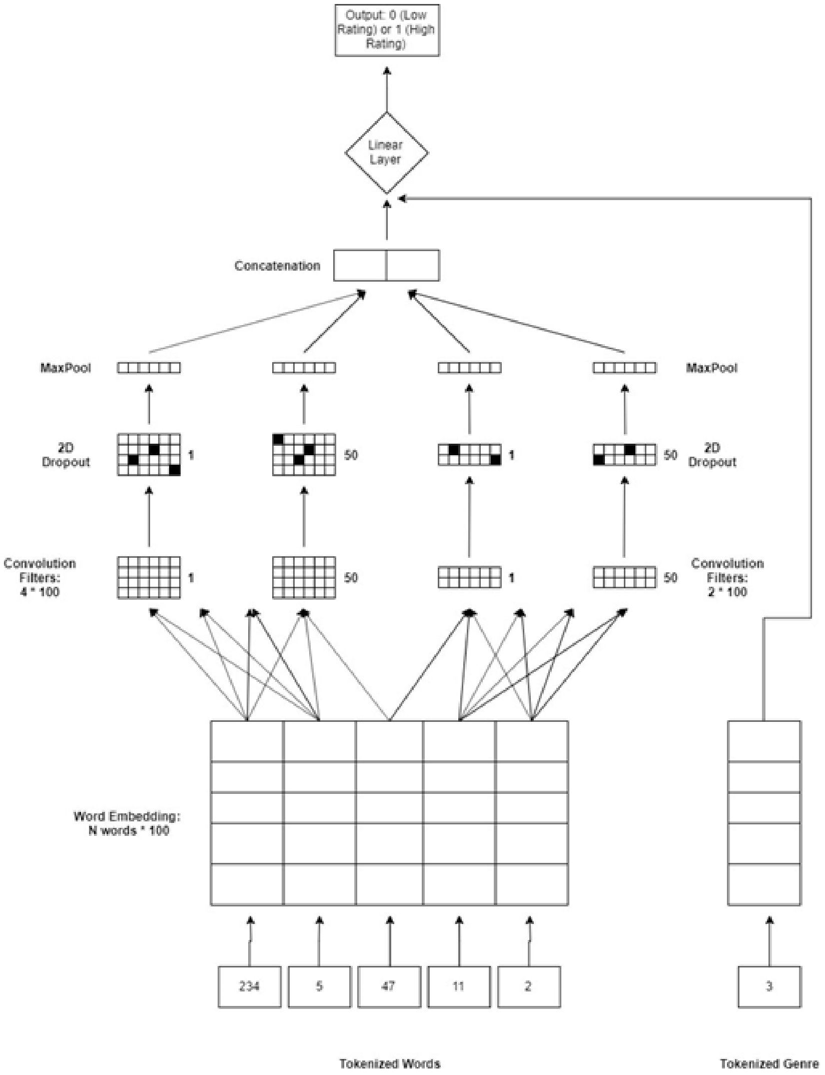
Baseline Model

The baseline model that we used is based off the baseline model used for a natural language processing assignment. The model tokenizes and encodes each word into a 100 by 1 word vector. The words in the review are then averaged into a single 100 by 1 word vector and fed through a fully connected layer to get the final result.



CNN Model

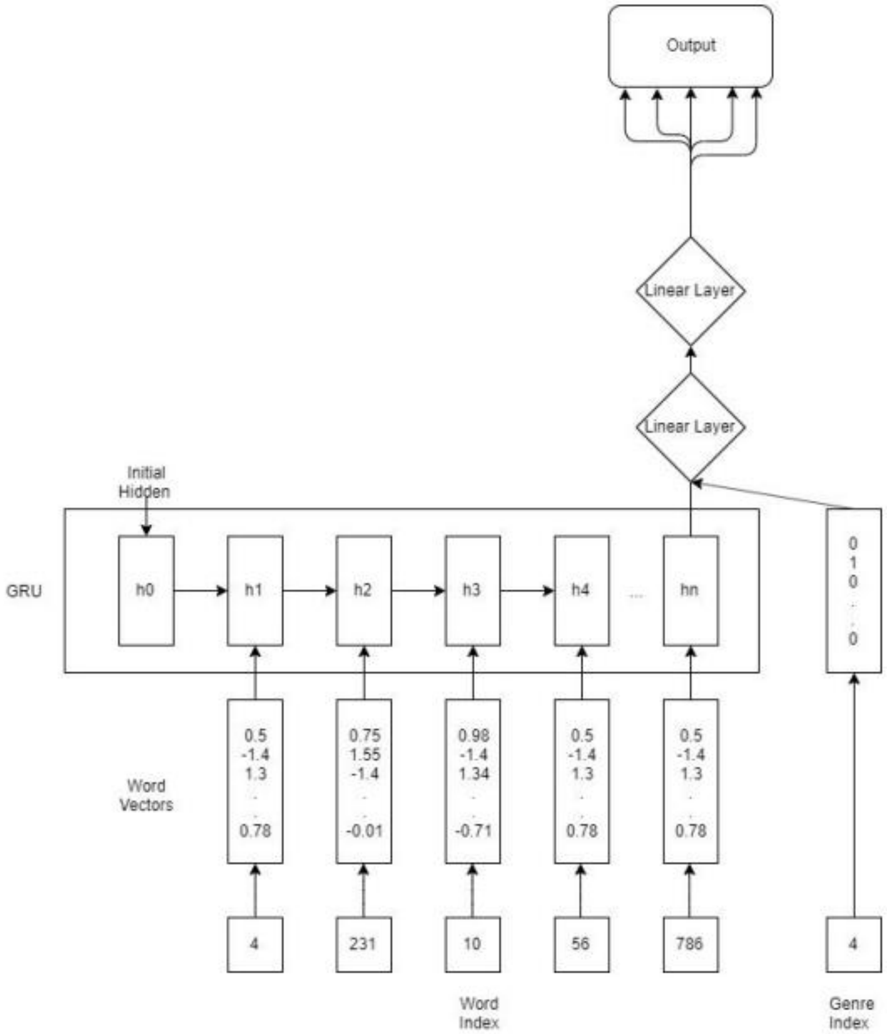
The CNN consists of two paths of convolutions. Each path has a convolution layer, a ReLu activation layer, and a dropout layer. The two convolution paths help the model identify critical word patterns in the reviews for sentiment analysis. Each convolutional path has 50 filters. The first path has a 4 by 100 kernel, and the second path has a 2 by 100 kernel. Each are followed by their accommodating max-pooling layer. Since the size of the kernel affects the scope with the NN learns, this ensures that the features learned are general enough to apply to reviews outside of the training set. The results of the paths are then concatenated together with the genre vector, and fed through a fully connected layer to produce the output.



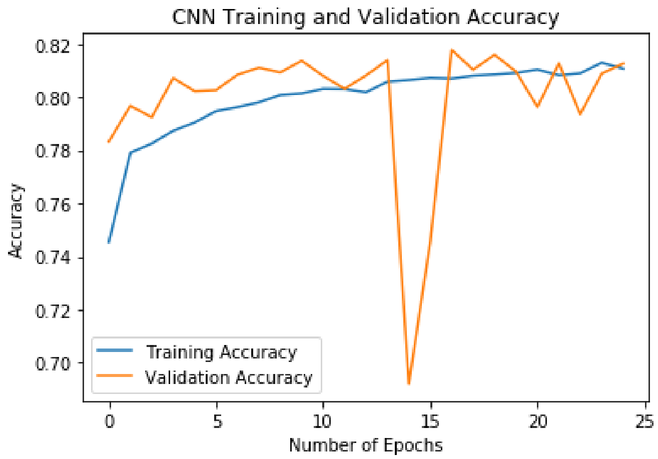
Output: 0 (Low Rating) or 1 (High Rating)
Linear Layer
Concatenation
MaxPool
MaxPool
2D Dropout
1
50
1
50
2D Dropout
Convolution Filters: 4 * 100
1
50
1
50
Convolution Filters: 2 * 100
Word Embedding: N words * 100
234
5
47
11
2
3
Tokenized Words
Tokenized Genre

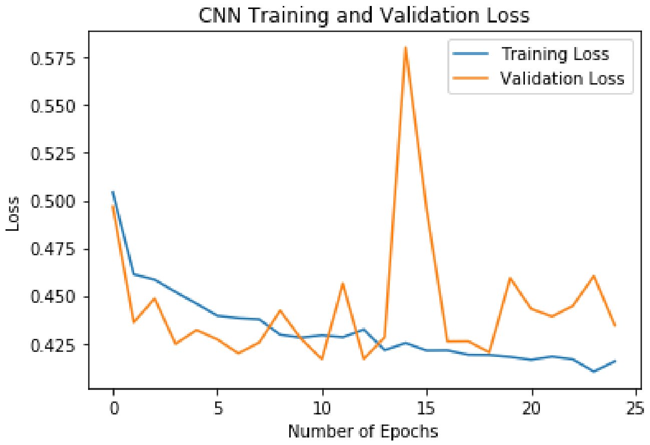
RNN Model

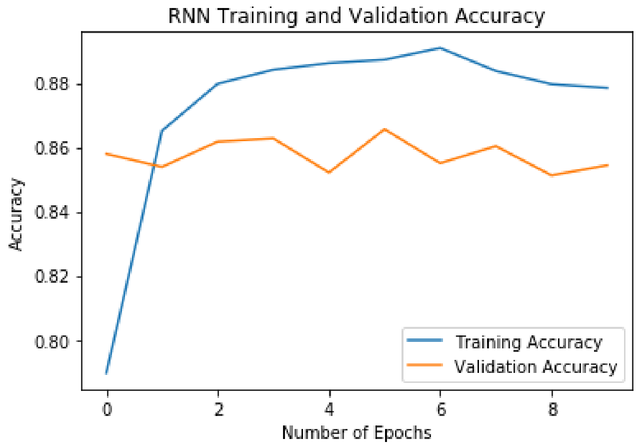
The RNN used is the Gated Recurrent Unit (GRU), which feeds the output into the following layer to be used in the training process. The batch size which provided the best results was 128, with a learning rate of 0.01. Using the same tokenization and word embeddings, the first 100-element word vector is fed to the GRU, which was initialized with an initial hidden layer. The output of this first layer is then sent to the next layer, alongside the next word vector in the sentence. This repeats until a final layer is obtained, the output. If the genre is included in the network, it is joined with this final output of the GRU, and then sent to the linear layers for output.

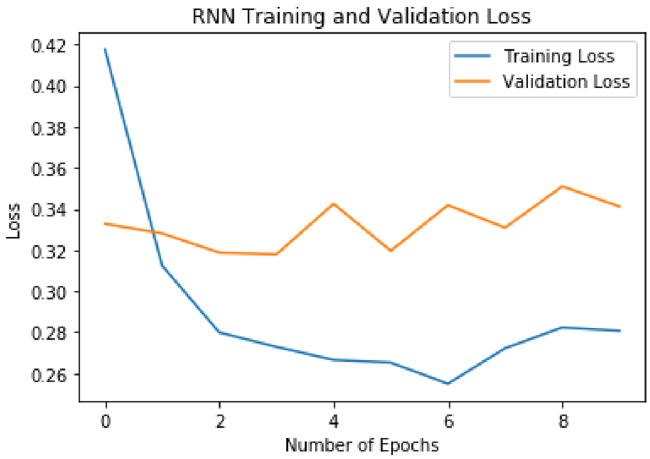


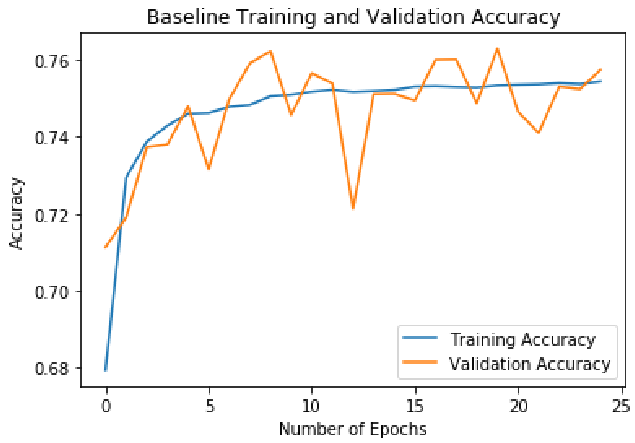
Quantitative Results

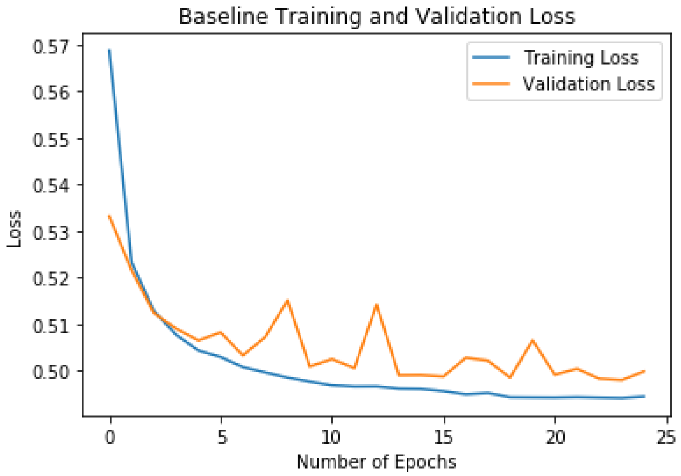

CNN Training and Validation Accuracy


CNN Training and Validation Loss


RNN Training and Validation Accuracy

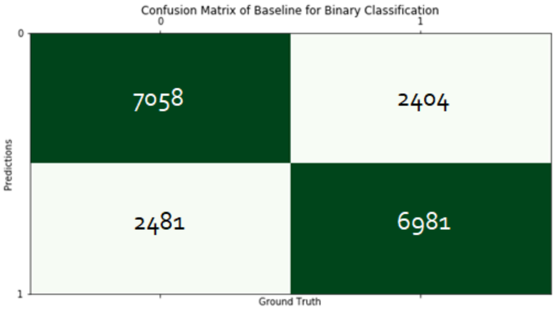

RNN Training and Validation Loss

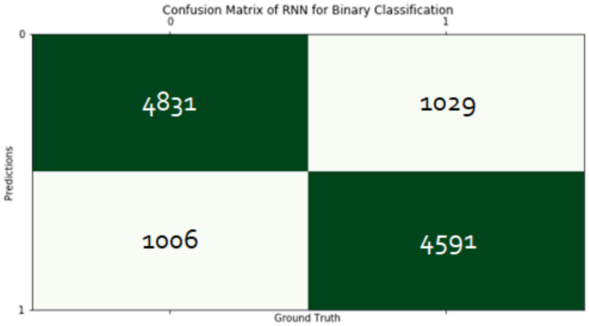

Baseline Training and Validation Accuracy

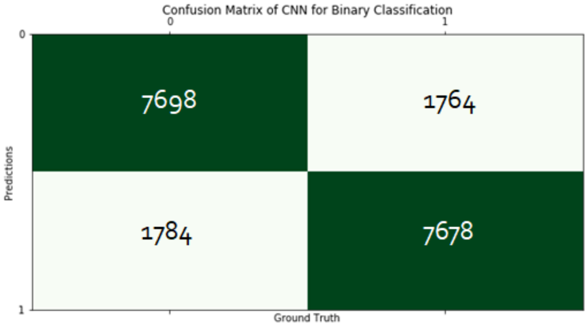

Baseline Training and Validation Loss

Qualitative Results

The output of the neural networks are binary labels of 0 and 1. ‘0’ represents the 1- and 2-star reviews, and ‘1’ represents 4- and 5-star reviews. The accuracy for both the training and validation were very similar for both labels. This is also reflected in the testing set.







One observation would be that the negative reviews always got slightly higher accuracy than the positive reviews. One possible explanation for this outcome may be because a greater variety of ways exist for expressing satisfaction than dissatisfaction. Dissatisfaction usually have similar contexts, and the difference in genre adds to the complexity of showing satisfaction.

Discussion and Learnings

From this analysis, we can observe that the neural network is sufficiently capable of categorizing book reviews to be negative or positive. A success rate of 85% achieves the initial goal of reaching at least ~80% success. Furthermore, the models were similarly capable in both validation and testing sets, indicating there is likely no overfitting. Given that it has been found that human sentiment analysis is ~80% [3], the neural network can be said to perform no worse than a human. Assuming that sentiment analysis is a subjective exercise, then it should be noted that it is quite likely that an accuracy rate significantly higher than human ability would be somewhat difficult. This would also be exacerbated by other pitfalls of automated sentiment analysis such as double negatives, proper nouns/brand names, sarcasm, etc. Therefore 85% is a very good rate of prediction for the scope of this project.

The original goal of the project was to create a rating from 1-5. However, due to the characteristics of the dataset, it was found to be more suitable to apply binary classification. This is because of the relative lack of reviews below 3-stars. The dataset was balanced initially to compensate, but given a shuffled set of testing data, the network greatly struggled to determine a rating from 1-5 accurately. Thus, the scope was eventually shifted from 1-5 to binary classification to remedy this. Though this is different than what was originally intended, the network is still capable of providing essentially the same information in a different representation. This is because it is still a numerical representation of a review's sentiment towards a book. It should be noted however, that it is still somewhat infeasible to create a direct translation of this binary classification to 1-5 (i.e. multiplying a prediction of 0.6 by 5 to generate a rating of 3). The binary representation still addresses the scope of the project, as the suggested applications are still fully possible.

Surprisingly, the results of using genre one-hot encoding were found to be significantly worse. The intention for the genres was to act as a classifier which indicated which features should have been weighted differently. This is because it is likely that features which are desired in a Fiction novel would be distinct from those which would be desired in a Nonfiction text. This may be due to the natural imbalance of certain categories, as well as the over-emphasis of genre as an indicator of rating. The dataset was originally balanced to maintain an equivalent number of negative and positive reviews, and was not balanced for genre, as this would not only have been infeasible (as there are a significant number of distinct genres), but also inaccurate due to the fact that there may be significant bias between genre and score. In other words, we believe that genre plays an over-emphasized role in correlation to rating score. For example, it may be common for certain genres to receive favourable reviews (i.e. Mystery). As such, the neural network inaccurately favours books of the Mystery genre, the effect of which would detract from the original goal of a rating based on the review text.

Some things that may be done differently would be to have a more diverse dataset, attempt new types of models, and investigate influences outside of just the network. Furthermore, expanding the scope of the project to include previous user input to create recommendations would be an interesting avenue to explore.

Ethical Framework

Non-Maleficence: Since the network represents the values of the population, the system rates books that most of the population will enjoy. This could potentially reduce exposure of books of controversial topics, though they may be of political or cultural significance. For example, a book may be heavily criticized not for its quality but because it discusses a nation's historical misdeeds and would thus be rated poorly by the system. Should the network be involved in the recommendation process, it would reduce exposure of the topic to the public. This reduced exposure would further feedback as only similar perspectives are echoed and unchallenged. This violates non-maleficence as this would be a negative impact on the progression of society and culture.

Autonomy: A network capable of predicting the population's reception to a book would be incredibly significant to the financial well-being of an author. Authors which take advantage of the network may choose to produce books which capture most of the market. With an oversaturation of books that cater to the majority, authors may eventually feel compelled to write books that generate profit, as books of their own choosing may not sell as well. Therefore, this system coerces a specific standard of writing in an author's writing process in pursuit of financial gain, and thus constrains the options available.

Beneficence: In terms of beneficence, the neural network would be a valuable tool in delivering favourable reading options to consumers which they will enjoy reading and purchase. This is especially true if the network incorporated past reviews and purchases of the consumers. Predicated upon this is an increase of sales, and thus a healthier literature industry, which in turn promotes more authors to publish books. Furthermore, certain books which are less popular may be able to be catered towards the niche readers that do enjoy them. As such, a functioning recommendation system brings benefits to both the authors/publishers, and readers.

Justice: In the current incarnation of the network, it incorporates only the favourability of the book. Thus, it is not influenced by certain factors such as sales or political/cultural agendas. In this sense, the network is a representation of what the public thinks makes a good book, rather than something that indicates how well a book will sell, though the two are closely linked. Since the collection of the data was impartial and does not contain other influencers, the network does not impose an unfair bias in any particular manner.

[1] Srigiriraju, Bharadwaj. “Amazon Reviews: Kindle Store Category.” Kaggle, 22 May 2018, https://www.kaggle.com/bharadwaj6/kindle-reviews.

[2] McAuley, Julian. “Amazon Product Data.” Amazon Review Data, jmcauley.ucsd.edu/data/amazon/.

[3] https://mashable.com/2010/04/19/sentiment-analysis/