Issues in Information Systems
Volume 22, Issue 4, pp. 33-48, 2021

DOI: https://doi.org/10.48009/4_iis_2021_34-50

Reliability of NLP Models to Predict Human Sentiments

Amrut Khatri, *University of St. Thomas, amrut.khatri@stthomas.edu*
Dennis Labajo, *University of St. Thomas, laba3383@stthomas.edu*
Manjeet Rege, *University of St. Thomas, rege@stthomas.edu*

Abstract

Advancements in Artificial Intelligence have led to widespread adoption across various industries (McKinsey Survey 2019). It has attained a level of inferential performance that would have been otherwise challenging to achieve a few decades ago (Anyoha, Rockwell 2017). Models that perform time-series prediction, regression, facial recognition, among others have become widely accepted in various industry applications given the effectiveness and reliability of the technology. As AI continues to evolve, companies are beginning to look at AI to gain meaningful customer insight by interpreting human thoughts, sentiments, and empathy (Prentice, Catherine, Nguyen, Mai 2020). Traditionally, AI could correlate patterns from a vast collection of words but how will it overcome challenges given the distinctiveness of each person's expressions, emotions, and experiences (Purdy 2019)? A typical sentiment analysis focuses on predicting a positive or negative polarity of a given sentence. This task works in the setting that the given text has only one aspect and polarity. A typical ML model does well in this situation. However, a more general and complicated task is to predict based on different aspects mentioned in a sentence and the sentiments associated with each one of them. Our paper will attempt to show the challenges associated with the issue of multi-polarity and the role it plays in incorrectly predicting a neutral sentiment. As AI and human lives become increasingly intertwined, this paper will attempt to test the reliability of AI to wade through the complex human sentiments from words and sentences within a contextual domain and attempt to uncover the challenges that impedes the reliability of AI to accurately infer human sentiments (Hussein, D.M.E 2018).

Keywords: Sentiment Analysis, Natural Language Processing, Transfer Learning, Classification

Introduction

Businesses rely on customer feedback to understand customer needs and to help improve products and services (Gallagher, Conor, et. al. 2019). Feedback exists in different formats, for example star ratings 1 through 5 and can be combined with textual data to capture deeper customer insight that may serve valuable for the business. An abundant collection of customer feedback is an ideal environment for training an AI model as it contains actual reflections of unique customer experiences. Inclusion of star ratings in conjunction with textual review makes it more ideal as it provides important labels to correlate the type of sentiment expressed. To that end, this research acquired a rich dataset of customer reviews from Yelp, an online platform for customers to share their experiences on purchased products and services. The dataset contains the ingredients for building our AI model notably, the 5-star rating where 3 can effectively be designated as a neutral sentiment and test AI's reliability of correctly inferring multi-classification of sentiments. The goal for our AI mode is to identify correlations between input and output data. That is, given a set of words, can the model be trained to predict customer sentiments not only into positive or negative but neutral as well. In AI parlance, the general practice is commonly known as Sentiment Analysis and is a Natural Language Processing (NLP) technique where a trained neural network learns the various correlations from sequences of data to make a sentiment prediction (Fang, Xing, Zhan, Justin 2015).

Problem Statement

Given the unique nature of human expressions represented by series of words, can a model be trained to effectively recognize various degrees of contextual word patterns and correctly classify human sentiments into multi-classes: positive, negative, and neutral sentiments? Are human expressions, thoughts, and emotions too complex for an AI to model to decipher? This study will analyze AI's effectiveness by comparing its predictive performance on binary and multi-classification.

Technical Process/Tools and Methodology

This study follows a logical workflow outlined below and highlights various tools used in the execution of each step. The tools are all Python-based, a popular programming language used in the field of Data Science. JupyterLab was the primary integrated development environment (IDE) used for steps 1, 2, and 4 while PyCharm IDE was utilized in the development of custom scripts in step 3.

1. Data Preparation – This is a two-part step: first, the dataset was curated to focus only on customer reviews of the food and beverage business and second, perform a series of data cleaning processes to address several data issues that include null values, duplicates, non-English characters, mis-spelled words, and invalid reviews. To manipulate and transform the data, a software library called Pandas is used which works well on slicing and dicing large datasets. Custom Python scripts were then created to address each of the data issues mentioned. A spell-checking software library was adopted to check the spelling of words in each of the reviews and help determine whether the entire text is still review-worthy for our AI model to train from. Data preparation took approximately 60% of the entire project time and was worth the investment to ensure clean and tidy data (Wickham 2014).

2. Exploratory Data Analysis – To develop an intuition of the customer reviews as well as discover hidden insights, software libraries such as Seaborn and Matplotlib were used to visualize statistics of the dataset. Seaborn provides an easy and convenient out-of-the-box visualization features with less code while Matplotlib provides flexible capability to further customize your plots. This step was crucial in discovering and understanding how sentiments seem to be influenced by the length of customer reviews. This step also addresses the data imbalance across the different classes (positive, neutral, and negative) by utilizing another software library, Imblearn, that under-samples the data to the class with the lowest number of samples.

3. Building and Training Deep Learning Models – Custom Python scripts were built to automate the repeatable process of training the various AI models with increasing degrees in complexity. A total of four models were considered and evaluated. Figure 1 below illustrates a high-level workflow of this process. The main software library used in the building and training the models is Tensorflow, an extensive ecosystem of tools and libraries initially developed by Google Brain team and later open-sourced to the public under the Apache License 2.0 in 2015.

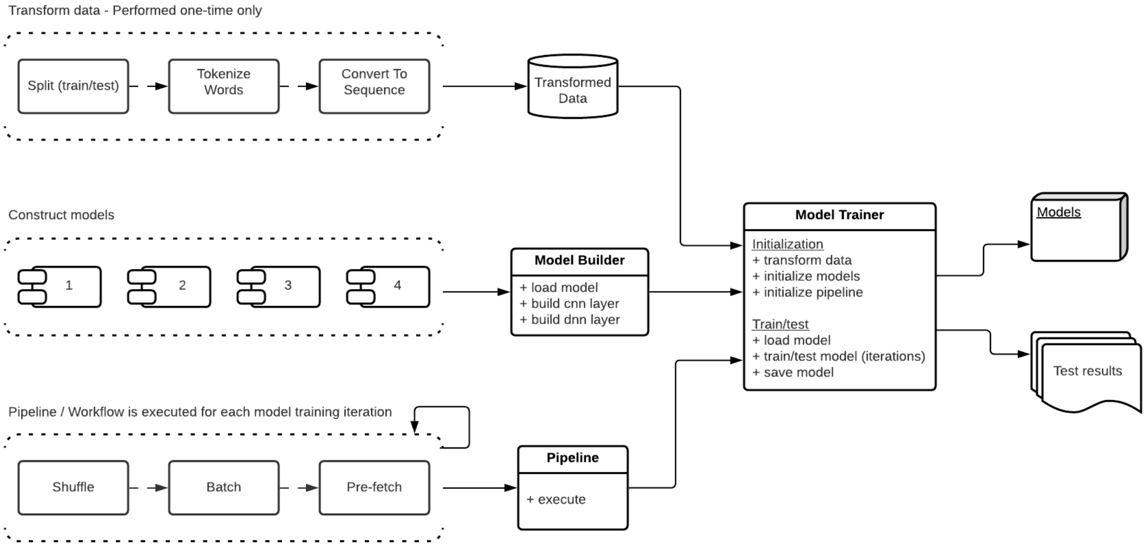


Figure 1: Model Training Workflow

A pre-configured containerized application (Merkel 2014) from Nvidia GPU Cloud (NGC) provides an ideal environment bundled with Deep Learning frameworks, libraries (such as TensorFlow and Scikit-Learn), and runtime toolkit and was used as the primary environment in the AI model training. Using a containerized environment that has been pre-configured and already installed with pertinent software libraries cuts down the time for setting up an environment from ground up and promotes re-use if the experiment were to be recreated. A dedicated GPU hardware, Nvidia Jetson Xavier, provided a local platform and runtime for the docker container with full access to the GPU resource enabling the training process to execute faster as compared to a regular CPU. The figure below illustrates the logical architecture of the runtime environment and the software stack used.

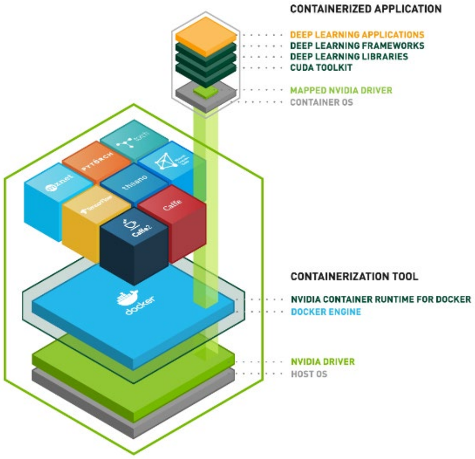


Figure 2: Container Runtime Logical Architecture
Image Credit: https://developer.nvidia.com/ai-hpc-containers

4. Model Assessment and Interpretation – Test results generated from the entire model training were extracted from the docker container and then plotted in JupyterLab for analysis and comparison using the Matplotlib visualization software library.

The sections that follow describes in more detail each of the methods that were summarized above.

Data Preparation

This section describes the steps how a dataset was curated and cleaned for the AI model to train from to effectively infer human sentiment into 3 main classes. To achieve this goal, the main data ingredients are essential: large enough dataset, labelled or that can be derived, and clean. Yelp's dataset provides the ideal size and includes star rating that can be converted to sentiment classes. This section describes the steps that were done to transform/prepare the data and ensure cleanliness that may otherwise adversely affect model's inferential performance.

Background

The customer reviews data was obtained with permission from Yelp, Inc and consists of the following datasets in JSON format:

1. Reviews – Actual customer reviews that also include star rating, useful/funny/cool votes, etc.
2. Business – Information about the business establishment e.g., id, name, address, is_open, etc.

There were other sub-datasets included but were not used in this study: Photos, Tip, Check-in, and User. A total of 8.2M reviews were collected from 160,585 businesses located in Ontario, Arizona, Nevada, Ohio, Quebec, North Carolina, Pennsylvania, and Alabama between the years 2005 – 2019. For the purposes of this study, only the Reviews dataset was needed with the Business dataset serving as a reference to filter reviews that focuses only on food and beverage type of business.

Curation

Yelp's dataset reviews cover a wide range of business types. To keep the complexity level down (model learning from different domains), the focus of the study is concentrated around restaurants. Hence a filter is created to select customer reviews from businesses whose categories fall into the following: "Food, Restaurants, Bars, Pub". Applying the filter resulted in the reduction of the reviews dataset from 8.2M to 5.6M.

The customer reviews in the dataset include associated star rating that ranges from 1- 5 with 5 being the highest. While the goal of the study is to determine model's ability to infer sentiment on multi-classes which would suffice with 5 classes, only 3 were used in this research. There are two approaches considered to limit the reviews to only 3 main classes:

1. Combine star ratings 1 and 2 into the "Negative" class, designate 3-star rating as "Neutral", combine star ratings 4 and 5 into the "Positive" class.

2. Drop reviews with 2 and 4 ratings while retaining reviews with 1, 3, and 5 to convey "Negative", "Neutral", and "Positive".

Since there were no guidelines that define what a truly negative nor positive reviews are, the second option was the most logical approach. The exclusion of 2 classes resulted in the additional reduction of data down to 3.7M reviews.

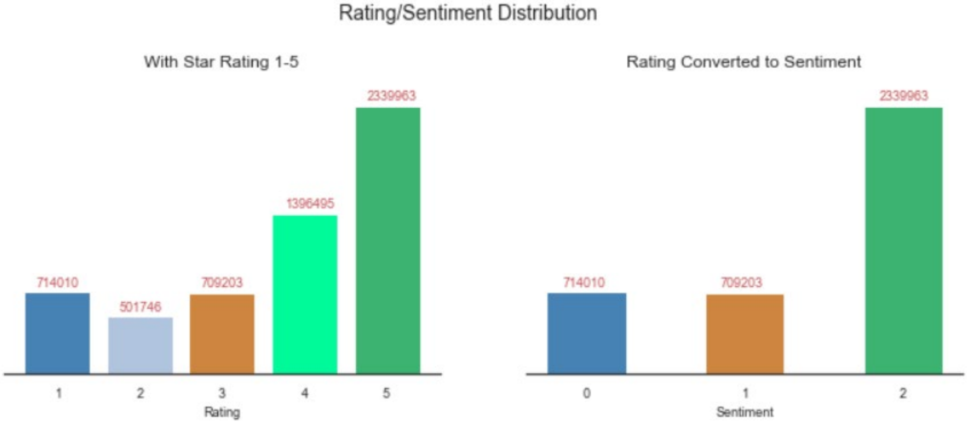


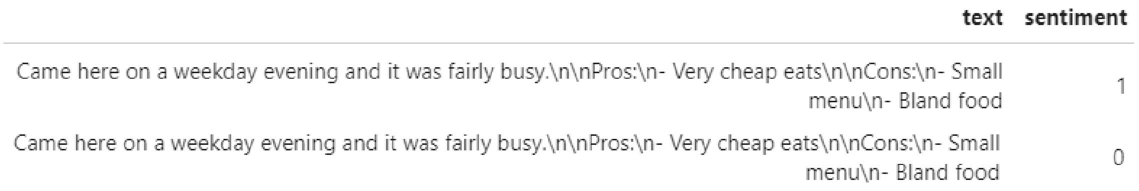
Figure 3: Reducing the Classes

Cleaning

After a careful review of the data, there were several issues discovered:

1. Null sentiment values – Empty sentiments provide no value to the model and hence removed from the dataset.

2. Duplicate reviews – The duplicate reviews came in two forms:
    a. Duplicate reviews with exact review text content and sentiment – For this scenario, the duplicate review was simply removed.
    b. Duplicate reviews with exact review text content but different sentiment – this anomaly posed a dilemma because deciding which review is legitimate is extremely subjective if not tedious. As a result, all instances of the duplicates were dropped from the dataset. An example of this anomaly is shown in Figure 4 below.



text	sentiment
Came here on a weekday evening and it was fairly busy.\n\nPros:\n- Very cheap eats\n\nCons:\n- Small menu\n- Bland food	1
Came here on a weekday evening and it was fairly busy.\n\nPros:\n- Very cheap eats\n\nCons:\n- Small menu\n- Bland food	0

Figure 4: Neutral or Negative?

3. Reviews with non-English characters – These reviews in Japanese (Kanji, Hiragana, Katakana), Chinese (Hanzi), Korean (Hangul), etc. Since this study did not have the resource to incorporate/translate these types of reviews to the English language, the reviews were instead excluded.

4. Remove reviews with large number of misspelled words – Each review in the dataset was processed through a spell-checking library where each word is assessed. If a review exceeds a total error of 1% of mis-spelled words, the review is excluded. The rationale for the extremely less forgiving

threshold is based on the garbage-in, garbage-out principle where processing bad data through the model will most likely produce unreliable result.

5. Invalid reviews – These reviews include limited text, in some cases contain only hyper-links (spam-like content) or unintelligible/meaningless series of characters that do not convey a legitimate review. These occurrences were removed from the dataset.

There was a total of 1.63M reviews removed from the dataset resulting in 2.12M remaining reviews.

Exploratory Data Analysis

The objective of this section is to discover insights and build a level of intuition about the dataset. This is also to understand how the model may be affected in case there are biases embedded in the data if not carefully addressed.

Feature Engineering

Customer reviews come in various lengths and the study was interested in learning the bulk of the distribution for review lengths and the occurrences of outliers. As a first step, a new feature called "length" is feature-engineered from the count of words for each review. Using a histogram, the visualization reveals a right-skewed distribution where the bulk (50%) of reviews falls between 33 and 119 words. The statistically accepted boundary for outliers is 248 words and was based on the IQR rule:

$$\text{IQR} = \text{Q3} - \text{Q1}$$
$$\text{Lower bound} = \text{Q1} - (1.5 * \text{IQR})$$
$$\text{Upper bound} = \text{Q3} + (1.5 * \text{IQR})$$

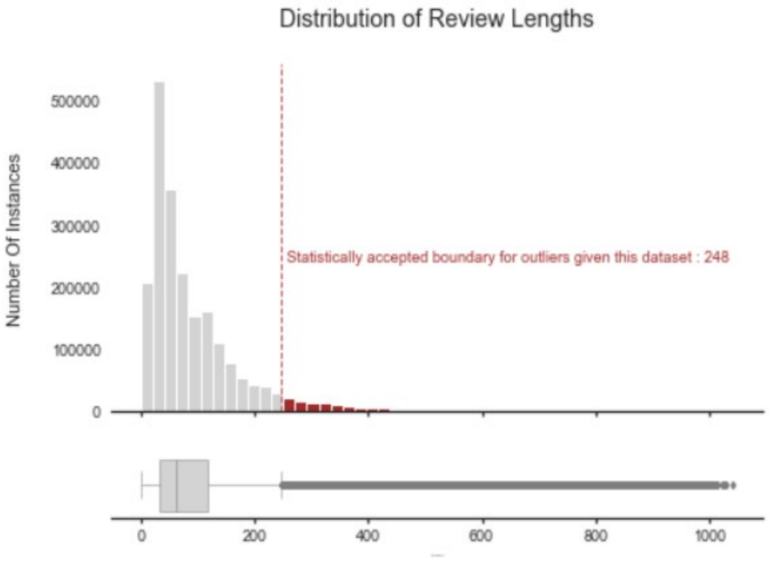


Figure 5: Distribution of Review Lengths

The next step is to understand the relationship between the length of a review and the resulting sentiment. Using a scatter plot to visualize the relationship reveals that shorter reviews are more likely to be positive than neutral or negative. Most interestingly is that customers who takes time to write longer reviews seemed more passionate about their experience but unfortunately is most likely to be an unpleasant review.

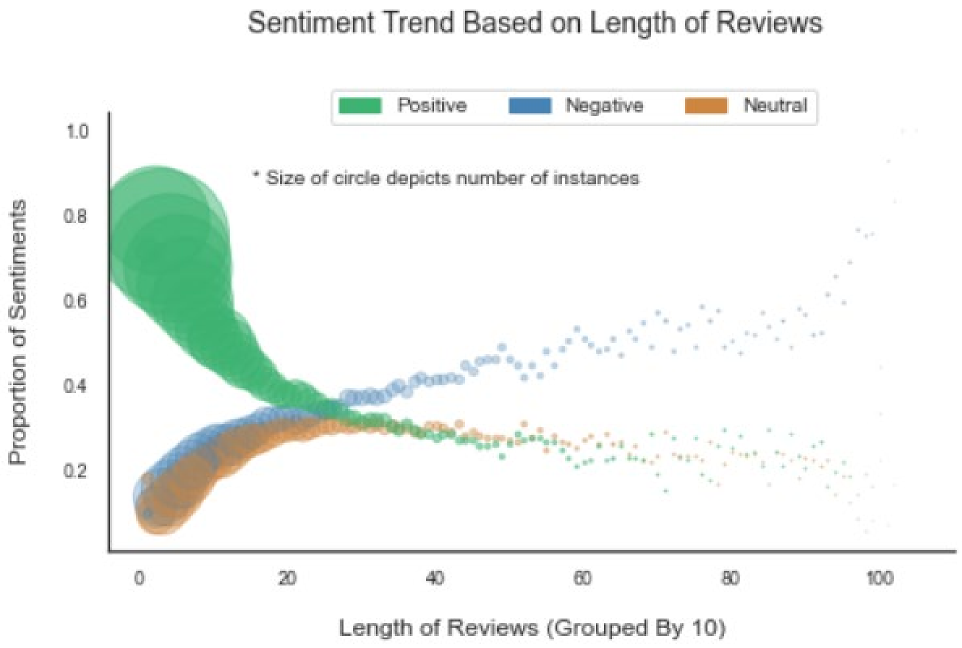


Figure 6: Relationship Between Review Lengths and Sentiment

Typically, outliers are removed to improve the accuracy of estimators (Acuna, Edgar and Rodriguez, Caroline 2004). However, there is a dilemma of removing the outliers in this case as outliers seem to exist and convey a special meaning and not generated from abnormal behavior. If the outliers were to be removed the explanation of that meaning may be lost and possibly resulting in a higher mis-classification rates. On the other hand, if the outliers were allowed to exist, this will make the model training way longer than necessary as each review most especially the shorter ones (1.9M total), will be required to be padded with tokens to equal the length of the longest review. Padding is essential in model training/prediction because machine learning models cannot process data with varying input lengths.

The option to only include reviews within the statistically accepted boundary was decided but contingent upon the resulting misclassification rate on both multi-classification and binary tests. This means that the change/removal of outliers will be reverted if there is a high rate of misclassification and evidence that the removal of outliers contributed to this factor.

Visualizing the proportions of sentiments by review lengths reveal an almost equal proportions between the three sentiment classes at the statistically accepted boundary for outliers. The removal of the outliers resulted in the further reduction of the dataset to 128K reviews.

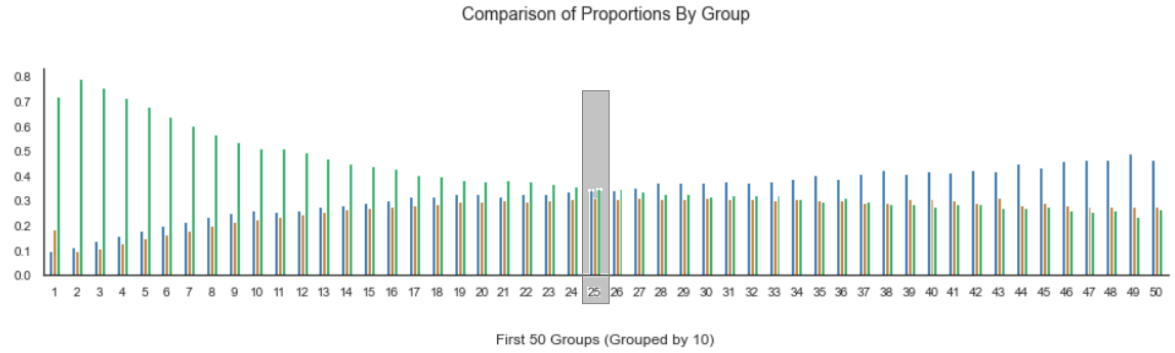


Figure 7: Proportions of Sentiments by Review Lengths

Imbalanced Classes

In the "Data Preparation" section, an imbalanced class was observed where "Positive" reviews are the dominant class and "Neutral" the minority. The distribution after a series of data transformation, cleaning, and removal of outliers, indicate the gap has narrowed a bit between the dominant and minority classes.

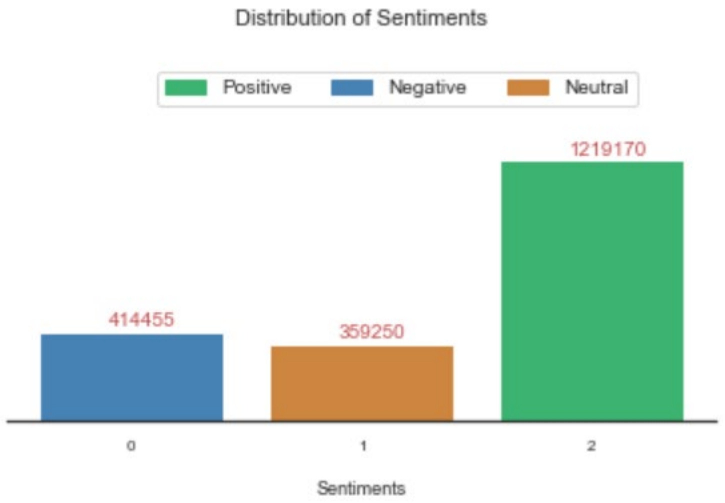


Figure 8: Imbalanced Classes

An imbalanced data creates bias towards dominant classes, so it is critically important to address this issue (Provost, Foster 2000). There are 2 ways to approach this:

1. Under-sample (reduce) the dataset so that the negative, neutral, and positive sentiments have an equal number of instances. The least number of sentiments is neutral which means reducing the dataset down to the size equivalent to neutral's number of sentiments.

2. Keep the imbalance but assign appropriate class weights on the call to `model.fit` so that the negative and neutral sentiments will be treated equally with the positive sentiments.

With this approach, no data is discarded however the disadvantage of doing this is that a much substantial amount of data will still be used for training slowing down training time.

Option 1 was selected as the solution to address the imbalance primarily due to help with the limited compute resources available for this study. A python library "imblearn" (Lemaitre, Guillaume, Victor, Dayvid, and Aridas, Christos 2017) was used to under-sample which resulted in the reduction of the dataset to 1.07M with the balanced classes.

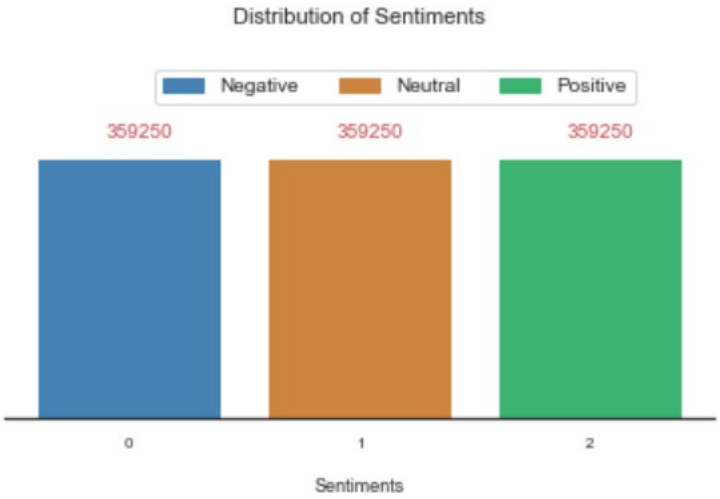


Figure 9: Resulting Balanced Classes

Model Training and Evaluation

This section describes the various models tested and then a discussion on the performance of the model. Model training was executed on Nvidia's Jetson Xavier dedicated GPU. Setting up the environment involves simply downloading a Docker image from Nvidia's NGC website that provides pre-packaged containers with Python runtime, TensorFlow, Scikit-Learn, and other pertinent Deep Learning libraries.

Pre-processing

The previous sections made several data transformations to prepare the data for the model to consume during training and testing. However, before data is fed into the model, the dataset must be in a form/shape the model expects. The high-level steps below describe the pre-processing required by the model:

1. Represent review text as numbers – Each word is tokenized with a corresponding integer value.
2. Padding - Reviews that are shorter than the longest review needs to be padded with a placeholder value to equal the length of the longest review.

Network Architecture

Four network architectures were selected for the models with varying degrees of complexity: Simple Bi-Directional, Deep Convolutional 1D, and 2 Pre-Trained Models 50 and 300 dimensional from Glove (Jeffrey, Socher, et. al 2014).

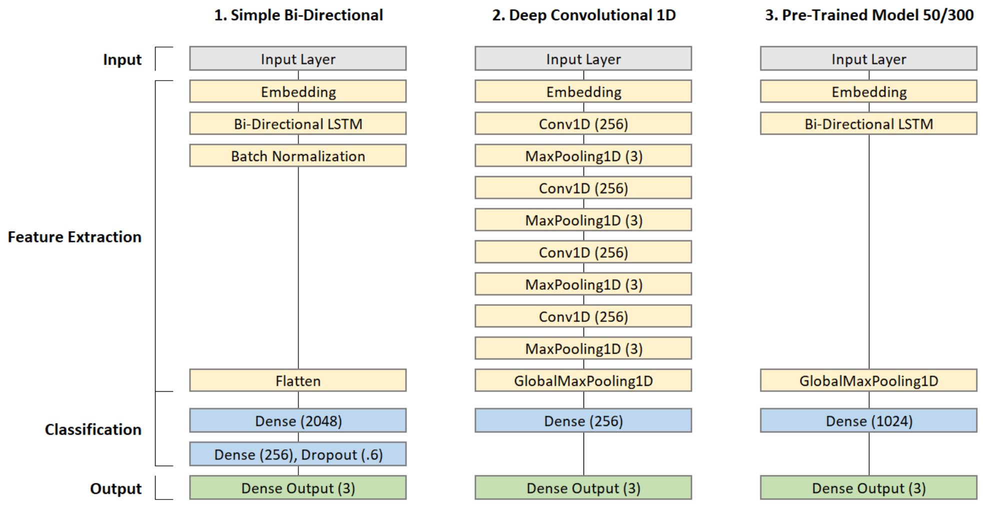


Figure 10: Network Architectures

The main hyper-parameters used in testing each of the models are outlined below:

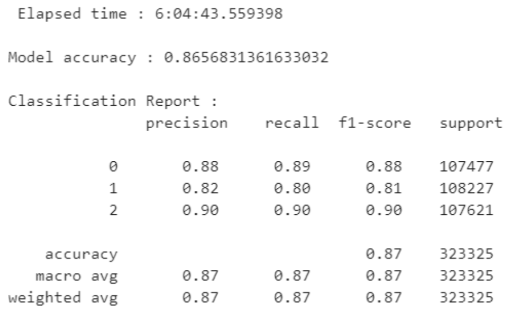
1. Optimizers: Adam, RMSProp, and Stochastic Gradient Descent (SGD) – All models were evaluated for each of the optimizers.

2. Activation: Exponential Linear Unit (ELU)

3. Early Stopping for terminating training epochs once validation loss degrades.

Multi-Classification

After running a total of 12 multi-classification training/validation tests (3 per model for each of the optimizer), the best performing was model 2, which is Deep Convolutional 1D using the SGD optimizer resulting in an accuracy of 86%. Training took a little over 6 hours for just this model alone. There is hardly any gap in the accuracy per epoch between training and validation which indicates the model is not over-fitting on training data.



```
 Elapsed time : 6:04:43.559398

Model accuracy : 0.8656831361633032

Classification Report :
              precision    recall  f1-score   support

           0       0.88      0.89      0.88    107477
           1       0.82      0.80      0.81    108227
           2       0.90      0.90      0.90    107621

    accuracy                           0.87    323325
   macro avg       0.87      0.87      0.87    323325
weighted avg       0.87      0.87      0.87    323325
```

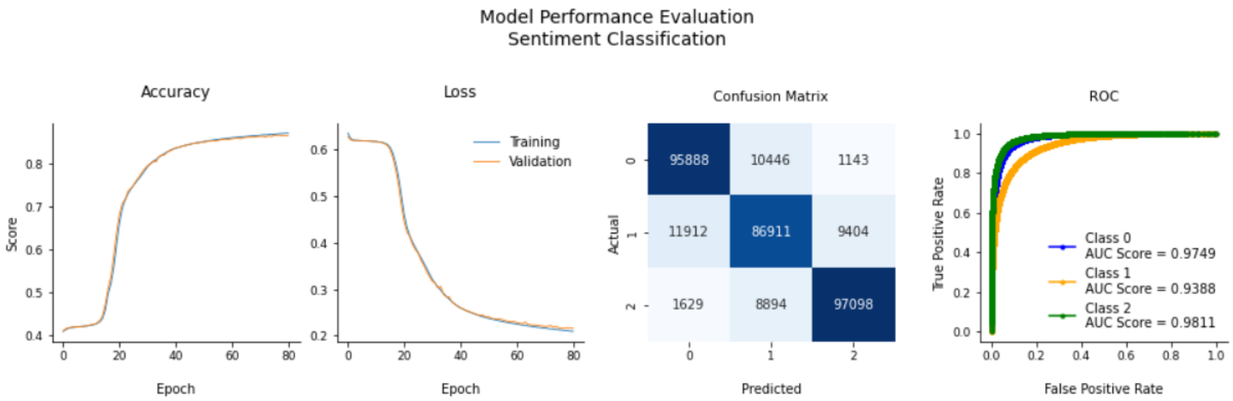


Figure 11: Multi-Classification Performance

The notable metric is the F1 score for the neutral class ("1") which is significantly lower compared to the other two classes. This is also observed in the ROC plot above where there is a noticeable smaller area under the curve (orange line color) for neutral compared to the other two classes. The confusion matrix indicates the model struggling to make sense of a sentiment that involves "neutral". For example, the misclassified counts where neutral is involved:

1. Predicted "negative" (0) but is actually "**neutral**" (1) – 11.9k
2. Predicted "**neutral**" (1) but is actually "positive" (2) – 8.9K
3. Predicted "'positive" (2) but is actually "**neutral**" (1) – 9.4K
4. Predicted "**neutral**" (1) but is actually "negative" (0) – 10.4K

Recall that in previous Exploratory Data Analysis section, some reviews were removed whose total count of words exceeds the statistically accepted boundary for outliers. Did the removal have any adverse effects on the model's performance particularly for neutral sentiments? To find out, the misclassified neutral sentiments review lengths are plotted and determine the length of reviews where there is high rate of misclassifications.

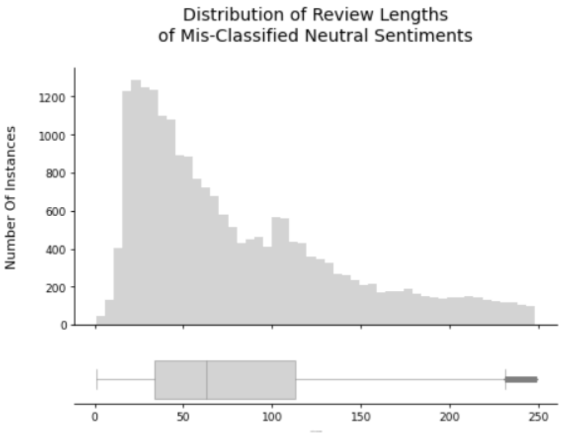
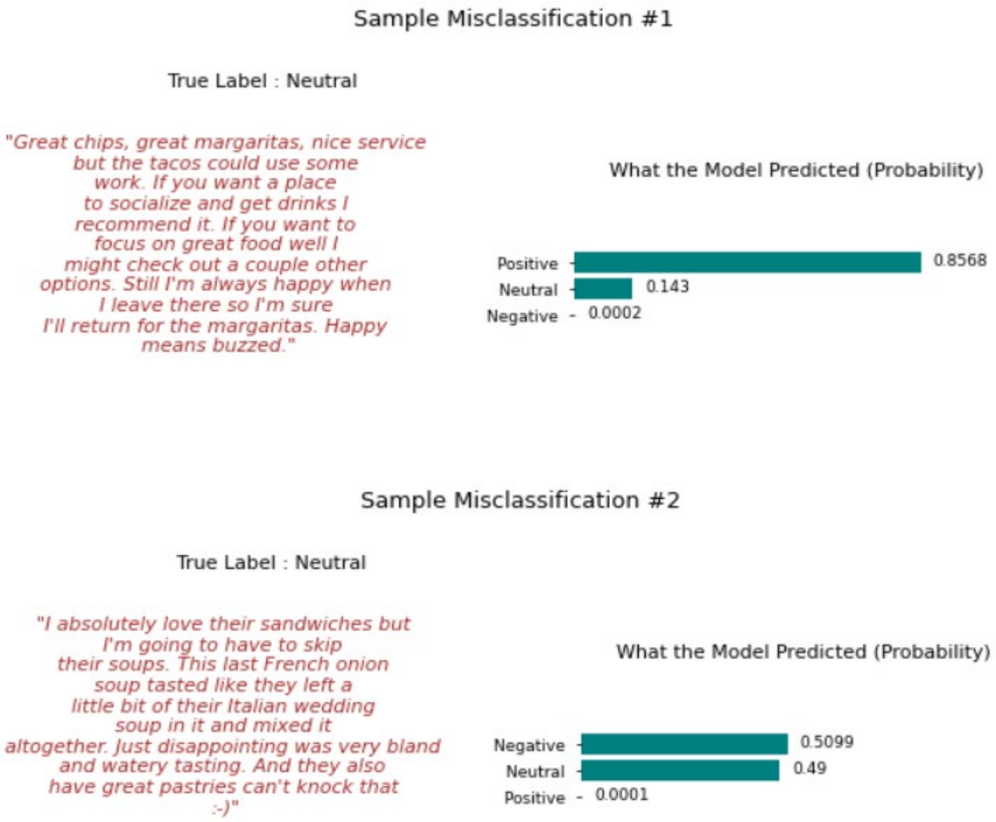


Figure 12: Review Lengths for Mis-classified Neutral Sentiments

The concentration of misclassification of neutral sentiments are from reviews that contain 30 – 50 words and the rate decrease as the length of reviews increases. This proves that removal of outliers identified in the Exploratory Data Analysis do not contribute to the poor performance of the model on neutral classes.

To investigate further the nature of the high rate of misclassification, samples of neutral sentiments are reviewed together with the corresponding probabilities that the model predicted.



"Great chips, great margaritas, nice service but the tacos could use some work. If you want a place to socialize and get drinks I recommend it. If you want to focus on great food well I might check out a couple other options. Still I'm always happy when I leave there so I'm sure I'll return for the margaritas. Happy means buzzed."

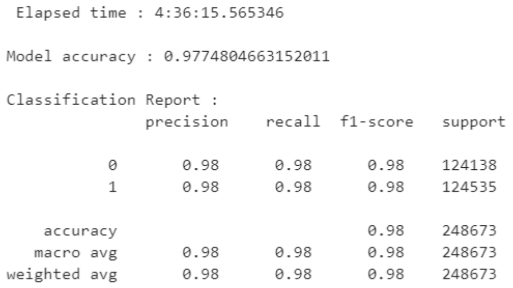
"I absolutely love their sandwiches but I'm going to have to skip their soups. This last French onion soup tasted like they left a little bit of their Italian wedding soup in it and mixed it altogether. Just disappointing was very bland and watery tasting. And they also have great pastries can't knock that :-)"

Figure 13: Sample Misclassified Neutral Sentiments

In the first example, the customer review provided a less-stellar rating while expressing an overwhelmingly number of positives compared to neutral or negative words and yet, it seems for the customer that everything would have to be a totally perfect experience for a rating to be truly positive. The probabilities determined by the model is in line with the occurrences of positive words in the review however the model is unable to gauge the true sentiment of the customer. This is where the challenge lies with sentiment analysis involving multiple classes. The separation lines between classes become blurry and increasingly subjective. Another challenge is that common words co-exist in other classes. Case in point example classification #2, where a neutral review contains positive "love, great" contrasted with negative "disappointing, bland, watery". The existence of both positive and negative is already a clue for a neutral sentiment, but the probabilities gravitated more towards the negative (most likely due to more occurrences of negative words) misclassifying the sentiment by a small margin of just 1%.

Binary-Classification

To prove the problematic subjectivity characteristic of the neutral sentiment, this class was removed from the dataset, and a new model is trained/validated on 12 binary-classification models (3 per model for each of the optimizers and hyper-parameters.).



```
 Elapsed time : 4:36:15.565346

Model accuracy : 0.9774804663152011

Classification Report :
               precision    recall  f1-score   support

           0       0.98      0.98      0.98    124138
           1       0.98      0.98      0.98    124535

    accuracy                           0.98    248673
   macro avg       0.98      0.98      0.98    248673
weighted avg       0.98      0.98      0.98    248673
```

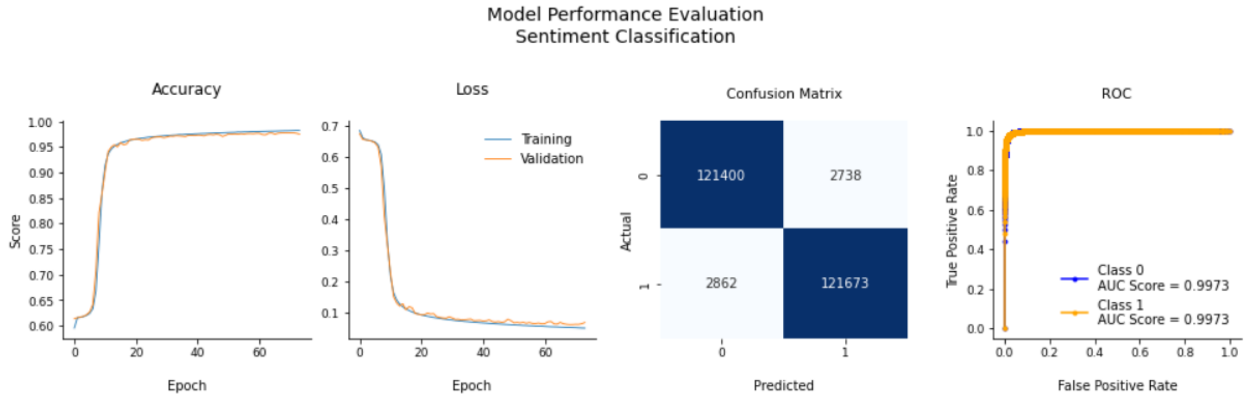


Figure 14: Binary-Classification Performance

The best performing was model 2, Deep Convolutional 1D using the SGD optimizer with a significant increase in accuracy of 97%. Training took a little over 4.5 hours to reach approximately 78 epochs and converge to the lowest error. The result confirms that with just two classes, the separation line is more defined and the sentiment less subjective.

Conclusion

A lot of effort were done to curate a dataset, ensure its cleanliness and bias-free. Different architectural models were trained with varying degrees of complexity as well as inclusion of Transfer Learning from two pre-trained models developed at Stanford that were trained on a large word corpus. However, test result from the best performing model on multi-class can only achieve an overall accuracy of 87%. Result also shows a high misclassification rate on the neutral sentiment which garnered a low F1 score of 81% compared to 88% and 90% for the negative and positive classes, respectively. This clearly shows that the trained model struggles to make sense what defines a neutral sentiment. Reviewing one example of a sentiment from figure 13 that was incorrectly classified as positive shows the patron had a lot of good comments for the food, service, and margaritas except unfortunately for the tacos. The model interpreted the review as positive with a high probability of 85%. The patron however had a different view of the overall experience and prefers neutral rating regardless of how great the margaritas were which, according to the review, the patron will most likely return for. Is it possible other patrons who share the same sentiments around food, service, and margaritas except for the tacos be more forgiving and provide a more favorable rating? It is more likely than not and that is based simply on the fact that no two people are truly alike. One person's neutral sentiment may be somebody else's slightly negative or slightly positive for another person. Some patrons would probably argue that a positive rating is reserved for a truly exceptional night out experience, but other patrons might also argue along the same lines that an overwhelmingly number of positives on other factors far outweighs the bad tacos. The second misclassification example also from figure 13, the patron ranted at length about the soup but had a praise or two for the sandwich and pastry. The model predicted negative with a probability of 51% focusing most likely on negative words around the soup, but the patron was more forgiving with a neutral rating instead. The high rate of misclassification on the neutral sentiment stems from the mix use of both positive and negative words within a neutral class which confuses the trained model. Removing the neutral and re-training the model only on two classes proves this to be the case as results from binary classification test shows a high F1 score of 98% both on positive and negative class.

An important question that needs to be answered when it comes to multi-classification problem on human sentiments is how can an AI model go beyond from simply learning the correlation of words or review lengths, to determining one person’s overall experience? Human thoughts are complex and to decipher the real intent of a sentiment lies in the ability to determine overall contextual background and yet the real answer may still be subjective. NLP models struggle in this regard as the separation lines become blurry as proven on three simple class of sentiments.

Suggestions for Future Studies

In the curation section of this study, the 2 and 4 rating were discarded in favor of 1, 3, 5 to represent the three main classes of negative, neutral, and positive. The rationale is the lack of guidelines on the definition of what the 2 and 4 ratings truly represent. The decision may have had an adverse effect on the neutral sentiment which was the minority class among the two other classes before the dataset was balanced from down sampling. A suggestion for future study is to increase the training size for the neutral class by collapsing the 2 and 4 ratings into neutral to augment the 3 rating. This will enable the model to learn more from the slightly less positive and slightly less negative as well as widening the margin of error for the neutral sentiment. If test result yields a lower misclassification rate for the neutral sentiment, then this just may be a simple case of training size which NLP models may be sensitive to. In addition, the cleaning phase of this study removed a substantial size of reviews due to high occurrence of misspelled words in a review. An alternative approach is to isolate and remove the misspelled words from the review in lieu of discarding the review entirely.

The dataset was filtered from the beginning to mainly focus on the food and beverage related reviews. Another suggestion for increasing the training size is to cautiously explore the possibility of considering reviews from other type of businesses and determine if it improves model performance.

Another suggestion is to explore a more complex model and powerful enough to find the correlations that generalize well on multiclassification sentiments. A grid search framework from the Scikit-learn library provides a facility to discover a wide array of models and hyper-parameters to find the best combinations that yields the lowest error. Grid search has proven to work well on machine learning models; however, this study was not able to explore its use on Deep Learning models.

Combining all the suggestions above will result in substantial increase in data as well as increased compute resource requirement from using grid search, so the final suggestion for future study is to utilize a more powerful GPU for training.

References

Acuna, Edgar and Rodriguez, Caroline (2004). ”An empirical study of the effect of outliers on the misclassification error rate”, University of Puerto Rico at Mayaguez

Anyoha, Rockwell (2017). "The History of Artificial Intelligence", Harvard University

Fang, Xing, Zhan, Justin (2015). "Sentiment analysis using product review data", Journal of Big Data, Article 5

Gallagher, Conor, et. al (2019). "The Application of Sentiment Analysis and Text Analytics to Customer Experience Reviews to Understand What Customers Are Really Saying", IGI Global

Hussein, D.M.E (2018). "A survey on sentiment analysis challenges", Journal of King Saud University, Vol. 30, Issue 4, Pages 330-338

Lemaitre, Guillaume, Victor, Dayvid, and Aridas, Christos (2017). "Imbalanced-learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning", Journal of Machine Learning Research, Vol. 18 Nos. 17, 1-5

Merkel, D. (2014). Docker: lightweight linux containers for consistent development and deployment. Linux Journal, 2014(239), Article 2

Pennington, Jeffrey, Socher, Richard, and Manning, Christopher D. (2014). "GloVe: Global Vectors for Word Representation", Computer Science Department, Stanford University

Prentice, Catherine, Nguyen, Mai (2020). "Engaging and retaining customers with AI and employee service", Vol. 56.

Provost, Foster (2000). “Machine Learning from Imbalanced Data Sets 101”, New York University

Purdy, Mark et. al (2019). “The Risks of Using AI to Interpret Human Emotions”, Harvard Business Review.

Survey (2019). "Global AI Survey: AI proves its worth, but few scale impact", McKinsey and Co.

Wickham, Hadley (2014). “Tidy Data”, The Journal of Statistical Software, Vol. 59.