# Machine Learning Engineer Nanodegree

## Capstone Project

Lu Sun
January 22, 2019

## I. Definition

### Project Overview

Reviews written by customers are unstructured data and sentimental analysis is to understand the categories of the reviews, such as positive or negative. Natural language processing is used to program computers to process and analyze large amounts of natural language data, including reviews. Sentimental analysis is one of the applications of text classification, which is an important and typical task in supervised machine learning, assigning categories (positive, negative, etc.) to text.

The dataset used for this project is sourced from Kaggle. It is a Women's Clothing E-Commerce dataset revolving around the reviews written by customers. Its nine supportive features offer a great environment to parse out the text through its multiple dimensions. This dataset includes 23486 rows and 10 feature variables.

### Problem Statement

The goal of this project is to develop a classification model which can assign sentimental categories, i.e. positive or negative, to customer reviews written in text.

### Metrics

As the dataset is potentially skewed, Accuracy Rate may be biased if the majority of reviews are positive or negative. Precision and Recall for each classification need to be reviewed to ensure the model works well for skewed dataset. Therefore, F1 score, which combines Precision and Recall, is a suitable metric for this project.

F1 score is defined as below:

$$F1 = 2 * (precision * recall) / (precision + recall)$$

# II. Analysis

## Data Exploration

This dataset includes 23486 rows and 10 feature variables. Each row corresponds to a customer review, and includes the variables:

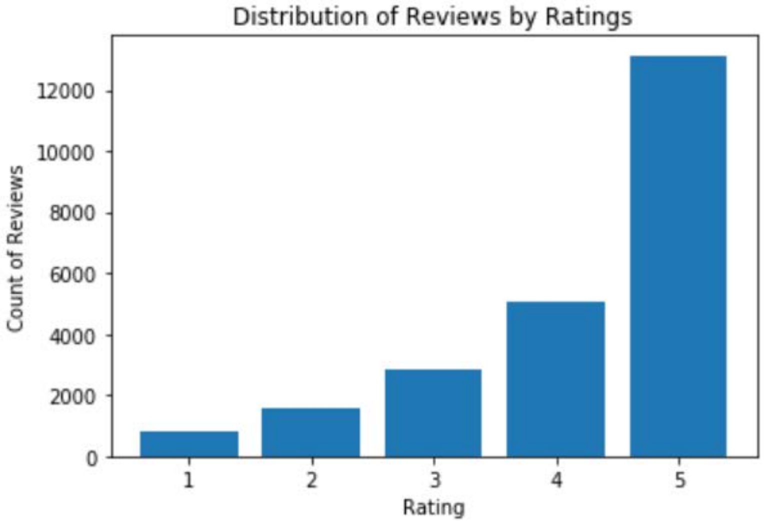
- Clothing ID: Integer Categorical variable that refers to the specific piece being reviewed.
- Age: Positive Integer variable of the reviewers age.
- Title: String variable for the title of the review.
- Review Text: String variable for the review body.
- Rating: Positive Ordinal Integer variable for the product score granted by the customer from 1 Worst, to 5 Best. Ratings 1 and 2 are considered as negative, 3, as neutral, and 4 and 5 are positive.
- Recommended IND: Binary variable stating where the customer recommends the product where 1 is recommended, 0 is not recommended.
- Positive Feedback Count: Positive Integer documenting the number of other customers who found this review positive.
- Division Name: Categorical name of the product high level division.
- Department Name: Categorical name of the product department name.
- Class Name: Categorical name of the product class name.

The percentages of missing data for each feature are shown below. *Title* misses 16% of data, which is considerably high and should be used for modeling. Other features don’t have significant missing data.

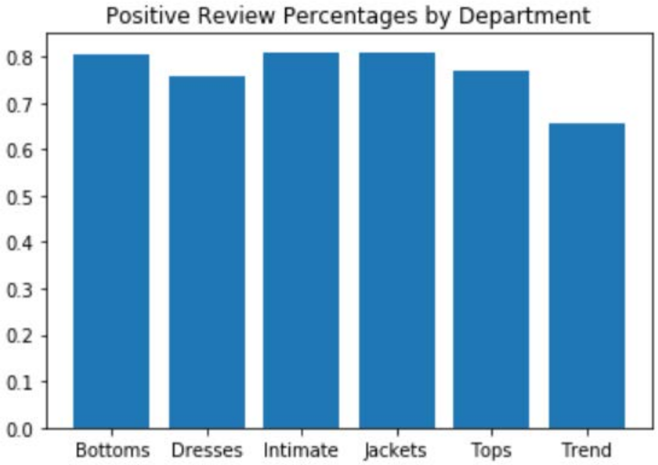
```
Clothing ID                0.000000
Age                        0.000000
Title                      0.162224
Review Text                0.035979
Rating                     0.000000
Recommended IND            0.000000
Positive Feedback Count    0.000000
Division Name              0.000596
Department Name            0.000596
Class Name                 0.000596
```

## Exploratory Visualization

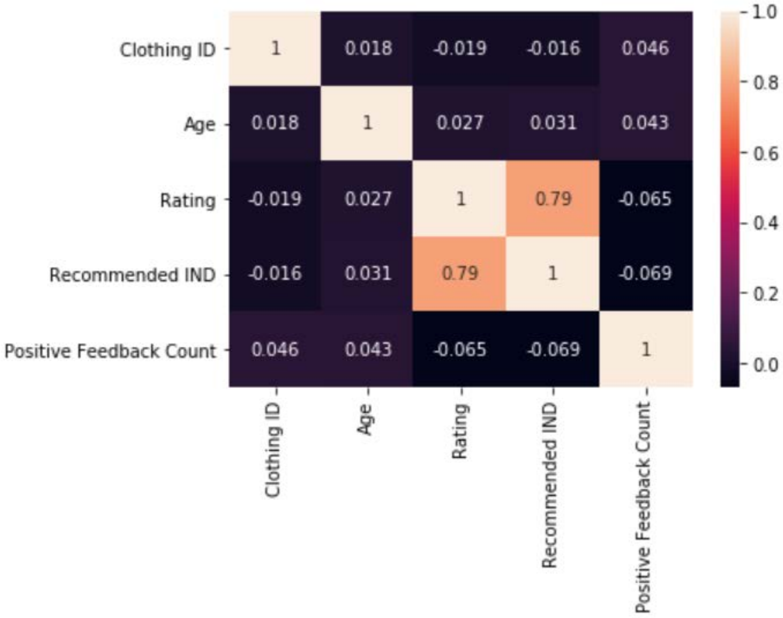
The count of reviews by ratings is displayed below. Per the chart, the dataset is highly skewed towards positive reviews, and the higher the ratings the more reviews. When choosing model training, validation and testing dataset, as well as selecting model features, techniques are required to solve the imbalanced data issue so as to avoid biased prediction toward positive reviews. The details will be explained in Methodology section.



In order to check if the data skewness appears across the categorical variable - department, the distribution of positive reviews (ratings in 4 or 5) by department was further examined. Per chart below, it is consistent across department that the majority of reviews are positive. Department doesn’t provide much information in inferring customers’ sentiment.



Next, the correlations between numeric features and Rating were checked. Most features, as shown in black cells, don't have strong correlations with Rating, and therefore are not suitable for predicting sentiments. *Recommended IND* has strong positive correlation with *Rating*, which makes sense as customers hold positive views would be more willing to recommend the products. However, since the goal of the project is to infer customers' sentiment based on their reviews, and the *Recommended IND* already indicates the sentiment, this feature should not be used. In summary, *Review Text* will be the only feature used to predicting sentiments.



## Algorithms and Techniques

The natural language processing tasks including the following steps:

1. Converting text to vectors that the machine learning algorithms can understand. Bag of Words technique will be used for this conversion.

2. The features of the classification model are selected based on the frequency of words appeared in negative reviews. Note that stop words will be removed when evaluating the frequency of each word.

3. Develop a supervised machine learning model, i.e. Support Vector Machine (SVM), to process the data and solve the classification problem. SVM is suitable and fast to train classification models with large amount of features. Various kernel functions, including linear and RBF will be trained and the one with the best performance will be selected.

## Benchmark

Since the dataset is highly skewed, with more than 78% of reviews are positive, use a simple model which predicts all reviews positive as a benchmark model. The accuracy of the model will be pretty good around 0.7. The F1 score of the positive category will be 0.88, having Precision being 0.78 and Recall being 1, while the F1 score of negative category will be 0, having infinity Precision and 0 Recall.

# III. Methodology

## Data Preprocessing

- Data records with any missing features are removed from the model dataset.
- Ratings of 1 and 2 are grouped as ‘Negative’. Rating of 3 is ‘Neutral’. Rating 4 and 5 are grouped as ‘Positive’.
- Texts were split into words. Punctuation, numbers, single letters, and stop words were removed from the list of words in each review.

## Implementation

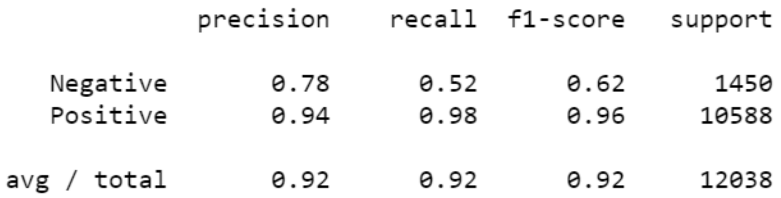
The model implementation involves the following steps:

1. Feature selection: The features of the classification model are selected based on the frequency of words appeared in negative reviews. Note that stop words were removed when evaluating the frequency of each word. FreqDist() function in NLTK library was used to find the most common words in dataset.
2. Tokenization: Convert strings to vectors based on the dictionary. The elements in a vector represents the count of the words in dictionary appeared in one review. Bag of Words technique was used for this conversion.
3. Model data selection: The model dataset is split into training, validation and testing dataset at 60:20:20 ratios. All data from Negative class are selected, and the same number of positive reviews are select so as to balance the dataset.
4. Develop a Support Vector Machine (SVM) to process the data and solve the classification problem. Two kernel functions, including linear and RBF were trained.
5. Model training: The model candidates were trained on training set.
6. Model selection: The candidate models were evaluated on validation set. The one with better performance in validation set was selected as the final model.

7. Model evaluation: The final model was evaluated against testing set to ensure that the model can be generalized.

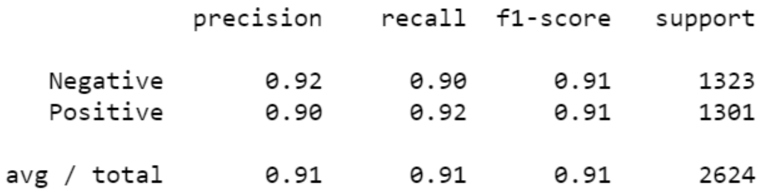
### Refinement

The model was first developed using all data available after data processing step. However, as the dataset was highly skewed toward positive reviews, the model performance in training dataset was not consistent between Positive class and Negative class. Both the Precision and Recall of negative reviews were not satisfied, leading to a low 0.62 F1 Score.



```
             precision    recall  f1-score   support

   Negative       0.78      0.52      0.62      1450
   Positive       0.94      0.98      0.96     10588

avg / total       0.92      0.92      0.92     12038
```

To enhance the model performance, model data selection was adjusted to select all negative reviews, and the same number of positive reviews from the original dataset. This leads to a balanced dataset for two classes. As shown in the table below, the metrics for Negative class improved a lot to over 0.90, which indicates good performance.



```
             precision    recall  f1-score   support

   Negative       0.92      0.90      0.91      1323
   Positive       0.90      0.92      0.91      1301

avg / total       0.91      0.91      0.91      2624
```

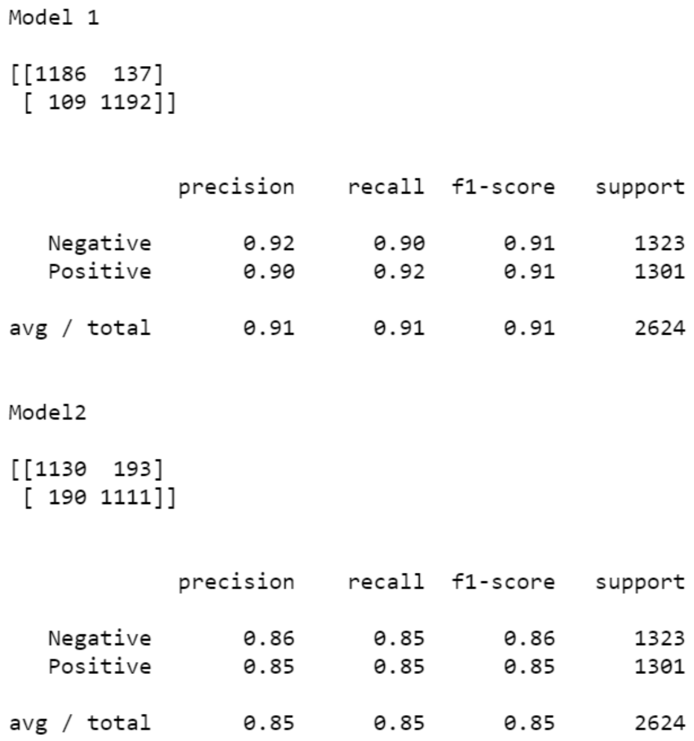
# IV. Results

### Model Evaluation and Validation

The model performance was evaluated using all training, validation, and testing sets.

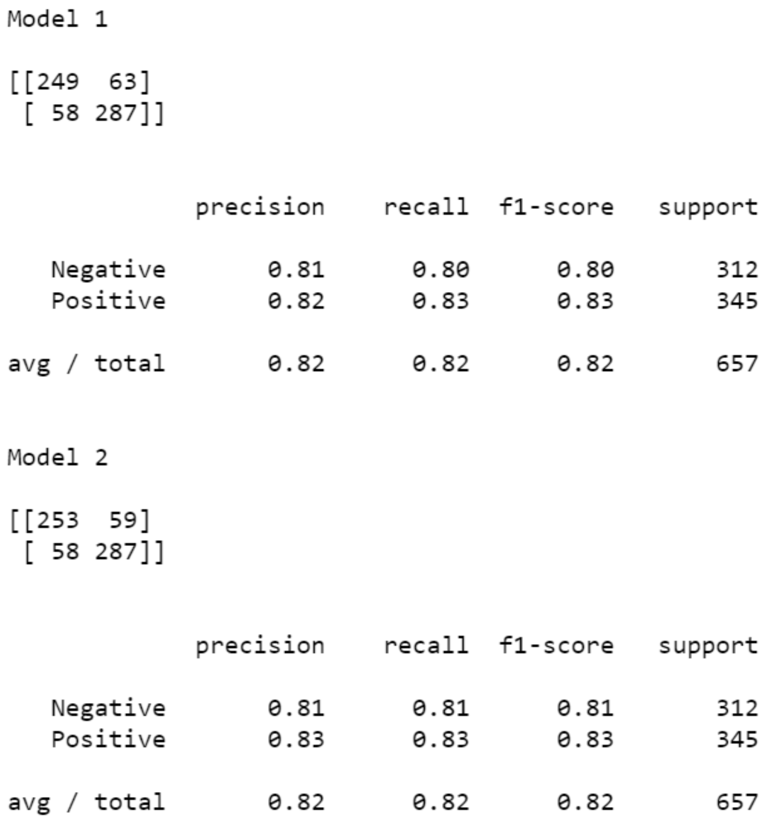
Specifically, two models were evaluated against training set, so as to confirm that both of them are qualified as model candidates. As shown below, the F1 Scores were over

0.85 which indicating good performance. Among them, Model 1 has better performance.



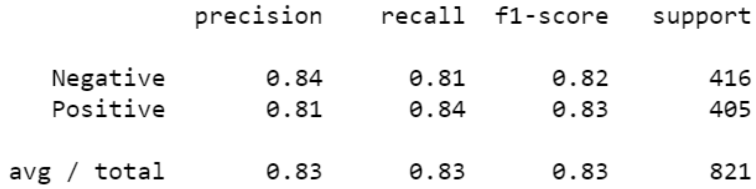
```
Model 1

[[1186  137]
 [ 109 1192]]


             precision    recall  f1-score   support

   Negative       0.92      0.90      0.91      1323
   Positive       0.90      0.92      0.91      1301

avg / total       0.91      0.91      0.91      2624


Model2

[[1130  193]
 [ 190 1111]]


             precision    recall  f1-score   support

   Negative       0.86      0.85      0.86      1323
   Positive       0.85      0.85      0.85      1301

avg / total       0.85      0.85      0.85      2624
```

The two models were then evaluated against validation set, and the one with better performance was selected as the final model. As shown below, both models have very similar performance. However, since predicting the minor class, negative reviews, correctly has more business impact, Model 1 which has slightly higher F1 Score was selected as the final model.



```
Model 1

[[249  63]
 [ 58 287]]


             precision    recall  f1-score   support

   Negative       0.81      0.80      0.80       312
   Positive       0.82      0.83      0.83       345

avg / total       0.82      0.82      0.82       657

Model 2

[[253  59]
 [ 58 287]]


             precision    recall  f1-score   support

   Negative       0.81      0.81      0.81       312
   Positive       0.83      0.83      0.83       345

avg / total       0.82      0.82      0.82       657
```

Finally, the model selected was evaluated against testing set, so as to confirm that the model performance can be generalized. As shown below, the F1 Scores for both classes were acceptable.



```
             precision    recall  f1-score   support

   Negative       0.84      0.81      0.82       416
   Positive       0.81      0.84      0.83       405

avg / total       0.83      0.83      0.83       821
```

## Justification

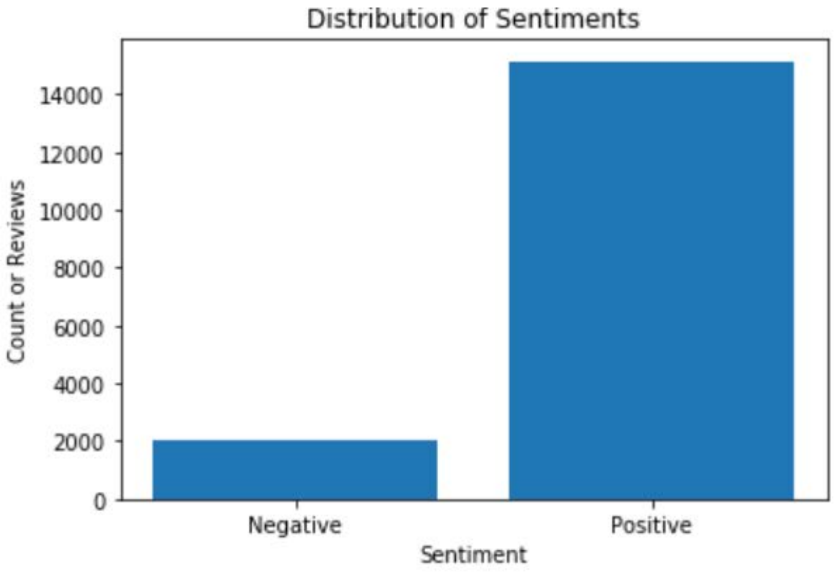
As discussed in the Benchmark section, a simple model to label all reviews as positive could achieve a good accuracy rate of 78%, a high 0.88 F1 score for the positive category and a zero F1 score for negative category.

The final model in this project reaches to an accuracy rate of 88%, and high F1 scores of 0.83 and 0.82 for positive and negative classes respectively. Overall, this model has much performance than the Benchmark model.

# V. Conclusion

## Free-Form Visualization

The below is to demonstrate how the dataset is unbalanced. This led to the discussions around feature selection and model data selection in earlier sections in order to enhance the model performance for predicting rare class.



## Reflection

The following steps were used in this project.

8. Converting text to strings. Remove strings which are not relevant to model development, such as stop words, single letters, punctuations, etc., from the list of strings.
9. Convert strings to vectors that the machine learning algorithms can understand. Bag of Words technique will be used for this conversion.
10. The features of the classification model are selected based on the frequency of words appeared in negative reviews. Note that stop words will be removed when evaluating the frequency of each word.
11. Develop a supervised machine learning model, i.e. Support Vector Machine (SVM), to process the data and solve the classification problem. SVM is suitable and fast to train classification models with large amount of features. Various kernel functions, including linear and RBF are trained.

12. The model dataset is split into training, validation and testing dataset at 60:20:20 ratios. The model candidates are trained on training set, and evaluated on validation set. The one with better performance in validation set is selected as the final model. Then the final model is evaluated against testing set to ensure that the model can be generalized.

As the model dataset is highly skewed, emphasizing the importance of the data and features from Negative class in step 3 and 5 are crucial. In step 3, only features from Negative class are selection. In step 5, all data from Negative class are selected, and the same number of positive reviews are select so as to balance the dataset.

## Improvement

To enhance the model performance, the following things could be done:

- The model is designed by using words as features only. Phrases could be added to features to make the features more comprehensive.
- Further evaluate the features and remove some noun features, such as dress, clothes, and pants, from the dictionary.