Capstone Project

Group -5

Debaarati Mitra
Hemachandar Nagarajan
Michael Sadi
Pavan Reddy Mamidi
Pawan Punera

Introduction

Listerine is an American brand of antiseptic mouthwash products. It is promoted with the slogan "Kills germs that cause bad breath". Named after Joseph Lister, who pioneered antiseptic surgery at the Glasgow Royal Infirmary in Scotland, Listerine was developed in 1879 by Joseph Lawrence, a chemist in St. Louis, Missouri. Originally marketed by the Lambert Pharmacal Company (which later became Warner-Lambert), Listerine has been manufactured and distributed by Johnson & Johnson since that company's acquisition of Pfizer's consumer healthcare division on December 20, 2006. The Listerine brand name is also used in toothpaste, chewable tablets, and self-dissolving teeth-whitening strips.

According to our client, they want to build a data mining platform in order to analyze the incoming data i.e. reviews through various channels of customer services such as calls, texts, reviews, and emails. The data(reviews) are predominantly derived from three major platforms namely listerine.com, social media platforms, and Google consumer care centers. In terms of scope, the data is from the US for the year 2018. Our end goal is to analyze the customer perspective on Listerine and categorize them into three subject levels.

Data Exploration

Data Overview:

This dataset is obtained from J&J for its Listerine product. It consists of 11215 observations and 69 parameters. Out of those 69 parameters, more than 50% of the data is missing. The Listerine dataset has a combination of numeric, categorical and character data. Subject level 1,2&3 are the target variables. We need to create separate supervised models for each target variable. We need to add or create additional variables from the reviews column that can have a high predictive power of the subject classes.

Preliminary Data Dictionary:

Variable Name	Data Type	Description	Example
subject_level_1_desc	Factor	Target Variable 1 – Categories under which the complaints fall under	Inquiry
subject_level_2_desc	Factor	Target Variable 2 – Categories under which the complaints fall under	Direct Sale
subject_level_3_desc	Factor	Target Variable 3 – Categories under which the complaints fall under	Product Stability
sentiment_desc	Factor	The sentiment of the verbatim assigned by the agent	Negative
comm_type	Factor	Type of communication	Facebook
verbatim_english	String	The review /complaint given by the consumer which is directly typed by the agent	What is this Iâ??m seeing in my bottle of Listerine?? It looks like cobweb or mold mixed with mucus. (SCN)
fiscal_period	Date	A fiscal year is a one-year period that companies and governments use for financial reporting and budgeting	7/4/2019

Table 1

Descriptive Statistics:

Univariate Analysis: Subject levels are our target variables, so our descriptive analysis revolves around them along with some other important variables like communication type and review column. There are very few missing values in subject- level 1 compared to other subject levels. From the figure 1 below we can observe that Feedback communication dominates subject-level 1 by having more than 80% of the column numbers. Inquiries are the next highest in subject-level 1. Subdivision of subject-level 1 is subject-level 2 where marketing services and product-based cases dominate subject-level 2. They both combine 85% of entries in subject-level 2. Subdivision of subject-level 2 is subject-level 3. Advertising, Product attribute, Availability, and Promotions are the values having more than 200 rows in subject-level 3. Advertising is the highest and Literature is the lowest in count in subject-level 3. Communication type is an independent variable and Facebook is the most used communication method in this column.

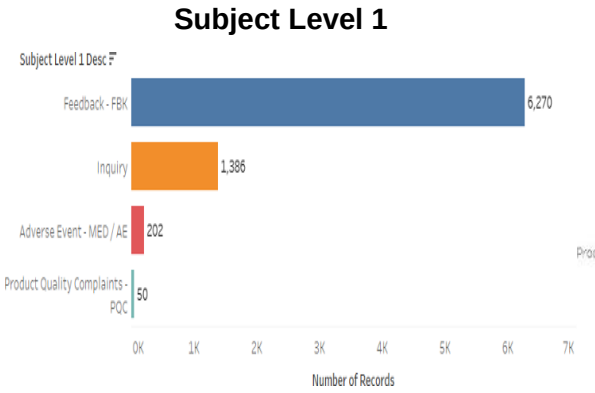

Figure 1

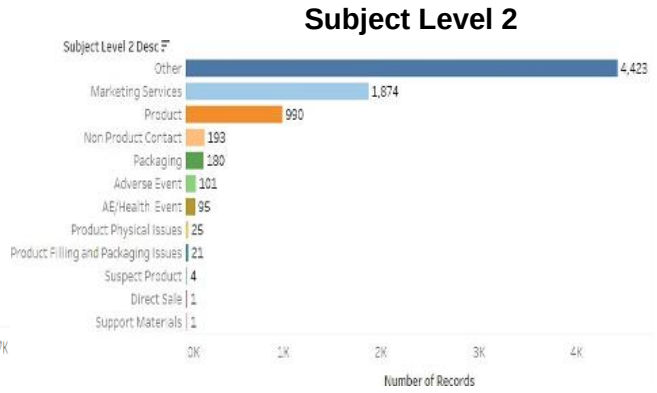

Figure 2

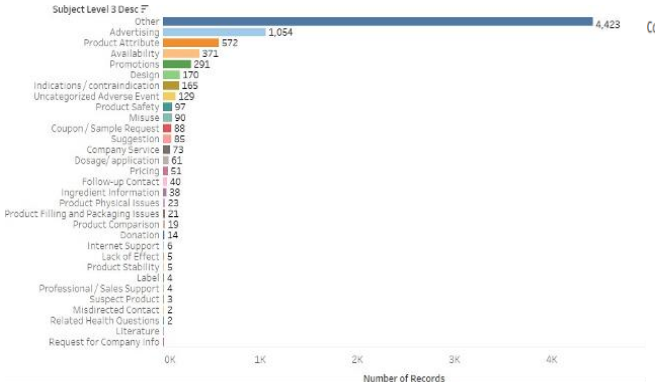

Figure 3

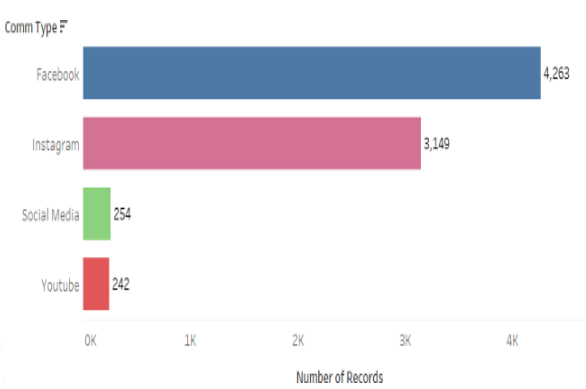

Figure 4

Bivariate Analysis: Most of the responses from the subject- level 1 are Feedback segments and it has high positive sentiment followed up by neutral sentiment. We can see the same in the figure 5 below. Consumers are predominantly using Facebook and Instagram for responses; Facebook has high neutral sentiment and almost equal positive and negative sentiment. Responses from Instagram give a lot of positive sentiment and very less negative sentiment. From

sentiment across subject-level 1 plot we can infer that most of the responses are feedback where most of the sentiments are positive and negative.

Sentiment Across Subject-Level 1

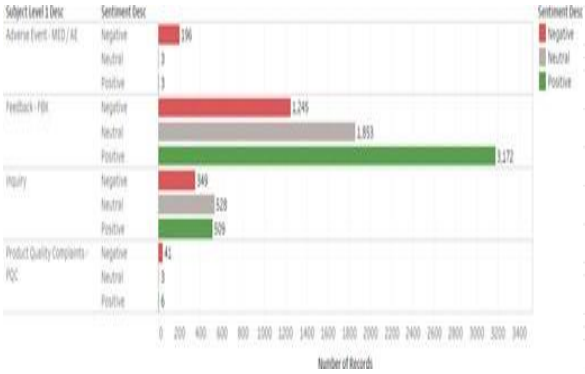

Figure 5

Sentiment Across Subject-Level 2

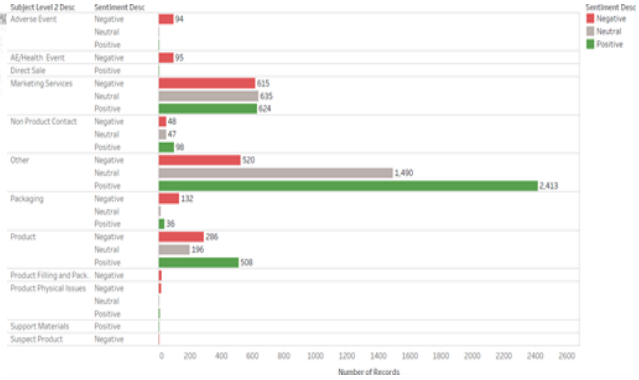

Figure 6

Sentiment Across Subject-Level 3

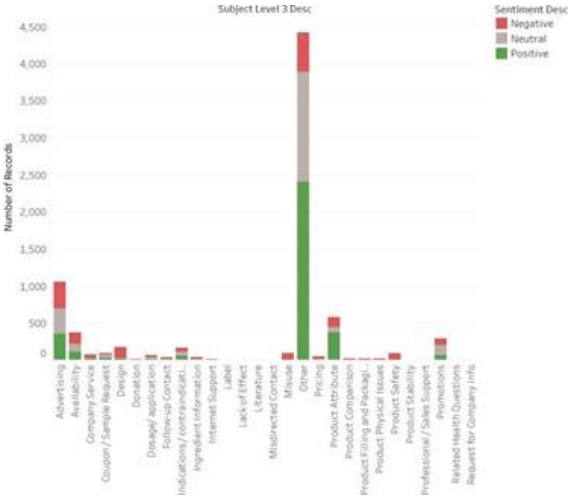

Figure 7

Sentiment Across Different Platform

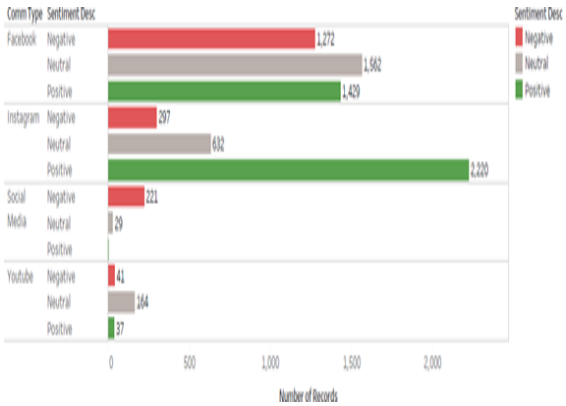

Figure 8

Data Transformation:

Our data has 11215 rows and 69 columns. Out of the 69 columns, in almost 38 columns, 70% of data is missing.

1. **Features selected:** Based on the initial exploratory analysis we selected 7 variables as mentioned above in the data dictionary.
2. **Dropping duplicates:** The were duplicates of the verbatim column. Meaning the same communications were repeated. We removed these and the number of rows came down to 7910.
3. **Handling missing values:** After the preprocessing steps, there was one missing value in Subject level 1 and Verbatim and 4425 each on subject level 2 and subject level 3. We removed the missing ones directly from subject level 1 and verbatim. But removing the ones in the other two doesn't make sense as those communications fall under some

subject level 1. It is better to assume them as **Other category**. We replace those missing values as **Other**.

Number	Coloumn Name	Missing Values
1	subject_level_1_desc	1
2	subject_level_2_desc	4425
3	subject_level_3_desc	4425
4	sentiment_desc	0
5	comm_type	0
6	verbatim_cdownload_content_without_symbol	1
7	fiscal_period	0
8	dtype: int64	

Table 2

Text Preprocessing: We performed following steps:

1. **Tokenization:** Split the text into sentences and sentences into words then lowercase the words and remove punctuation. Tokenization helps us to break the raw text into words, sentences which are called tokens. These tokens help in understanding the context or developing the model for the NLP. Generally, tokenization is helpful in interpreting the meaning of the text by analyzing the sequence of the words. For instance, for those rows with some sentences tokenization breaks then down to some separate chunks and using word tokenization breaks down all the chunks and sentences to the separated words.
2. Words that have fewer than 3 characters are removed.
3. All **stop-words** are removed which are noise in the text. For instance, text may contain stop words such as is, am, are, this, an, a, the, etc. by filtering a list of tokens from these words we removed the stop-words.
4. Words are **stemmed** (*words are reduced to their root form*) which is a process of linguistic normalization, which reduces words to their root word or chops off the derivational affixes. For example, connection, connected, connecting word reduce to a common word “connect”.
5. Words are **lemmatized** (*words in the third person are changed to first person and verbs in past and future tenses are changed into the present tense*) which reduces word to their base word, which is linguistically correct lemmas. It is more complicated than stemming. Stemmer works in an individual word without knowledge of the context. For example, the word “worse” has “bad” as its lemma. This will be missed by stemming because it requires a dictionary look-up.

Sentiment analysis: Sentiment Analysis or opinion mining is the contextual mining of the text. It helps businesses understand the sentiment of the customers associated with their brand and their products. Sentiment analysis models can focus on the polarity (positive, negative neutral), feelings (happy, angry, sad, etc.), and intentions (interested, not interested) of the customers. The library we have used for sentiment analysis is textblob. Textblob is used to process textual data. It performs the normal natural language processing tasks like part-of- speech tagging, noun phrase extraction, sentiment analysis, classification, and translation. The sentiment property returns a “namedtuple” of the form Sentiment (polarity, subjectivity). The polarity score is a float

within the range [-1.0, 1.0]. The subjectivity is a float within the range [0.0, 1.0] where 0.0 is very objective and 1.0 is very subjective.
The segmentation of the reviews done on the basis of the sentiment by the agent at the data centre is depicted in the figure 9 below:

Sentiment Recorded by the Agent and Vader

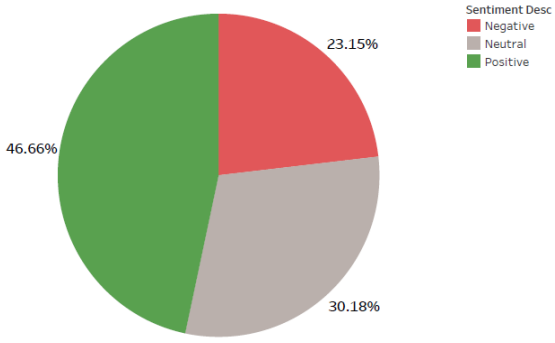

Figure 9

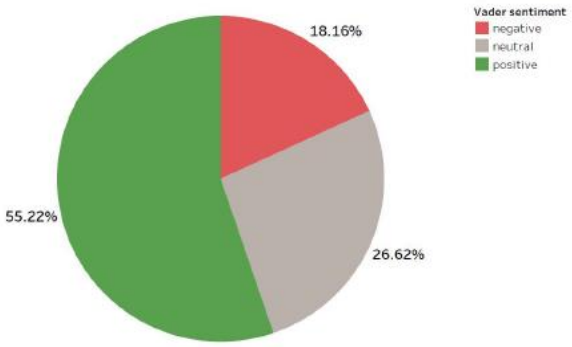

Figure 10

The sentiment from the VADER library is shown in figure 10. Here more communications have been classified as positive.

Polarity Scores using TextBlob package

When we perform sentiment analysis on our data using the textblob package, we get the output seen in Figure 11 and Figure 12. We can see that for most of the values "*Polarity Scores fall in the range 0.0 to 0.20*" which means that most of the reviews in our dataset are neutral in nature, followed by positive and then negative sentiments of the reviews. Hence it is similar to the categorization done by the agent at the data center.

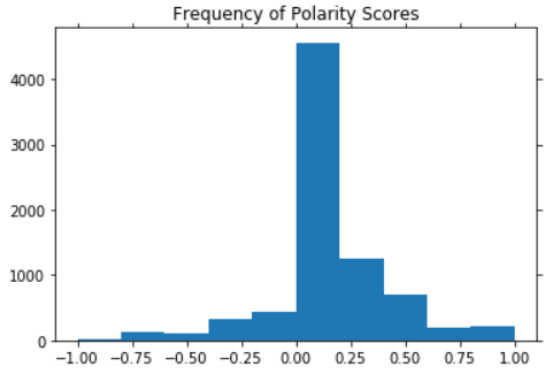

Figure 11

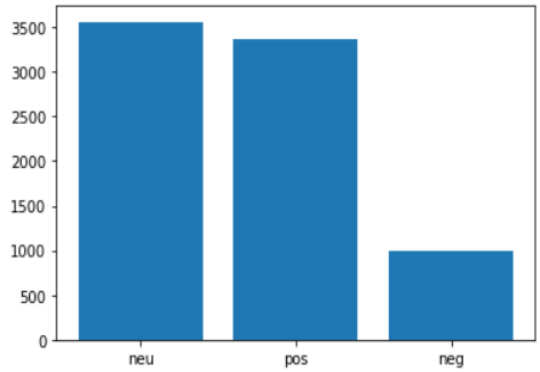

Figure 12

Text Vectorization is the process of converting text into a numerical representation. It is a fundamental step in the process of machine learning for analyzing data. A word vector represents a document as a list of numbers, with one for each possible word of the corpus. Vectorizing a document is taking the text and creating one of these vectors, and the numbers of the vectors somehow represent the content of the text. TF-IDF enables us to give us a way to associate each word in a document with a number that represents how relevant each word is in that document. Then, documents with similar, relevant words will have similar vectors, which is what we are looking for in a machine learning algorithm.

Classification Modeling

Our data setting is a multi-label classification where we want to predict more than one label. Using communication data, we can train three separate classifiers each predicting one subject level. There are 58 different combinations between the subject levels in the given dataset. These three independent classifiers might output a combination which might not be there from the above 58 combinations. We do not know whether the new set of combinations given by classifiers is possible or not.

So, for this dataset we are going to treat the combinations of class as class of their own and do a multiclass-single label classification. Out of those 58 different combinations some of the combinations have very few instances so we have selected an instance threshold of 20 due to which total combination came down to 23. So, we have splitted the data in the ratio of 70:30, 70% train and 30% test.

Multilayer Perceptron classifier:

	Precision	Recall	F1	Accuracy
Train	0.88	0.81	0.83	0.81
Test	0.84	0.78	0.8	0.78

Table 3

	Precision	Recall	F1	Accuracy
Train	0.82	0.81	0.81	0.79
Test	0.81	0.8	0.81	0.77

Table 4

From table 3 we can say that multilayer perceptron classifiers give train accuracy of 0.81 and test accuracy of 0.78. In order to get better results, we tuned our model and cross validated it so the matrix for the new model is shown in table 4. As we can see there is not much of an improvement in precision and accuracy.

Decision Tree:

	Precision	Recall	F1	Accuracy
Train	0.93	0.87	0.89	0.86
Test	0.82	0.76	0.78	0.77

Table 5

	Precision	Recall	F1	Accuracy
Train	0.93	0.87	0.89	0.87
Test	0.81	0.76	0.78	0.77

Table 6

After running the Decision tree, we obtained the train accuracy of 0.79 and test accuracy of 0.77 with the precision of 0.93 and 0.82 for train and test model respectively which is shown in table 5. We again tuned and cross-validated our model in order to get better results. Table 6 represents the matrix for this new model.

Naive Bayes:

	Precision	Recall	F1	Accuracy
Train	0.48	0.43	0.42	0.44
Test	0.47	0.43	0.42	0.43

Table 7

	Precision	Recall	F1	Accuracy
Train	0.48	0.43	0.43	0.44
Test	0.47	0.43	0.42	0.43

Table 8

From table 7 we can say that Naive Bayes give train accuracy of 0.44 and test accuracy of 0.43 with the precision of 0.48 for train and 0.47 for test model respectively. In order to get better results we tuned our model and cross validated it so the matrix for the new model is shown in table 8. So even after performing cross validation the results are not improving.

Recurrent Neural Network:

	Precision	Recall	F1	Accuracy
Train	0.63	0.59	0.6	0.88
Test	0.09	0.08	0.08	0.45

The above table shows the matrix for RNN. As we can see the model is overfitting so we have to overcome the overfitting of the model and fine tune the parameters to get better results which will be done in the final report.

What are going to do next?

- To get better matrix from Recurrent neural network
- Create classification pipeline to automate the subject level classification

References

1. En.wikipedia.org. 2020. Listerine. [online] Available at: <https://en.wikipedia.org/wiki/Listerine> [Accessed 29 October 2020].
2. Bird, S., Klein, E. and Loper, E., 2009. Natural Language Processing With Python. Sebastopol: O'Reilly Media, Inc.
3. Sciencedirect.com. 2020. Sentiment Analysis - An Overview | Sciencedirect Topics. [online] Available at: <https://www.sciencedirect.com/topics/computer-science/sentiment-analysis> [Accessed 29 October 2020].
4. Medium. 2020. Natural Language Processing: Text Data Vectorization. [online] Available at:
    a. <https://medium.com/@paritosh_30025/natural-language-processing-text-data-vectorization-af2520529cf7> [Accessed 29 October 2020].
5. ML+. 2020. Machine Learning Plus - Simplified Tutorials In R & Python. [online] Available at:
    a. <https://www.machinelearningplus.com/> [Accessed 29 October 2020].