Steps taken (code)

Libraries imported:

import numpy as np

import pandas as pd

import seaborn as sns

import matplotlib.pyplot as plt

%matplotlib inline

import warnings

warnings.filterwarnings('ignore')

from nltk.corpus import stopwords

from sklearn.feature\_extraction.text import TfidfTransformer, CountVectorizer

#from sklearn.metrics import classification\_repo

#rt, confusion\_matrix

from sklearn.model\_selection import train\_test\_split

import string, nltk

from nltk import word\_tokenize

from nltk.stem import PorterStemmer

from nltk.stem import WordNetLemmatizer

https://regenerativetoday.com/exploratory-data-analysis-of-text-data-including-visualization-and-sentiment-analysis/

**Exploratory data analysis**

The chosen dataset was sourced form outscraper.com . It consists of x data columns and x data rows or google maps reviews using the queries with ‘Dublin’ as a location and ‘restaurant’ as a filter. The csv data was read into a panda’s dataframe. The dataset contain’s 8 variables with quantitative data such as:

* ‘reviews’,
* ‘rating’
* ‘author id’
* ‘owner\_answer\_timestamp’
* ‘review\_rating’
* ‘review\_timestamp’
* ’ review\_likes’
* ‘review\_id’,

and 17 variables with qualitative information such as:

‘query’

‘names’

‘google\_id’

* ‘place\_id’
* ‘location link’
* ‘review\_per\_score’,
* ‘review\_id’
* ‘author\_link’
* ‘author\_title’
* ‘author\_image’
* ‘review\_text’
* ‘review\_img\_url’
* ‘review\_img\_urls’
* ‘owner\_answer’
* ’ owner\_answer\_timestamp\_datetime\_utc’
* ‘review\_link ‘
* ’ review\_datetime\_utc’

The following 7 variables: ‘name’ ( business name), ‘place\_id’, ‘rating’, ‘owner\_answer’, ‘owner\_answer\_timestamp’, ‘review\_text’ and ‘review\_text\_timestamp’, will be the focus for this report as part of the NLP processing.

Exploratory data analysis showed a range of review ratings from x to x providing a good spread of positive and negative reviews. Rows with missing data in review\_text and review\_img\_url column were deleted as these are not useful to the analysis. Other columns with missing such as and owner\_answer, owner\_answer\_timestamp, owner\_answer\_timestamp\_datetime\_utc were left as is as they will not be considered in the dataset analysis. The dataset consists of quantitative numerical data such as review rating, review id and number of review\_likes and descriptive quantitive data in the review\_text column.

Wordclouds provided visualisations of the frequency of certain word occurances in the dataset.

Visualizations of review length, the distribution of the word count, and the sentiment polarity, stop word distribution, and character count distribution were generated to judge if the dataset is skewed in any way.

Exploratory data analysis consisted analysing the columns :

-looking at the number of rows and columns using ‘.shape’

-checking for missing data

Summary statistics of the dataset were obtained using the python function ‘.describe’ such as count, mean , standard deviation, minimum and maximum values and the quantiles. ‘Review rating‘ showed a range of between 1 and 5 stars, with a mean value of 4.3. The range of ‘review likes’ spread from 1.5 to 17 like showing a level of engagement among reviewers.

**Data cleaning consisted of:**

The text cleaning was completed as part of the preparation for the application of NLP and classification. The number of stop words, the frequency and type of occurring punctuation symbols as well as the syntactic and lexical category quantities in each review adds valuable information and contributes to its correct classification.

Other pre-processing was necessary to remove certain characters such as hashtags, emoticons, non-word characters to ensure that the analysis could be performed effectively and automatically.

The counter class imported from ‘collections’ module was used to process and store punction types and counts for analyzing frequency and generate visualizations of punctuation types. The punctuation marks of each line were accumulated the in a dictionary and each element was transpose into data frame columns for a greater overview. Inauthentic reviews typically have typos, either an excessive amount or complete lack of punctuation in relation to their word count and poor grammar.

Positive speech tagging (POS) for nouns, adjectives and verbs were completed with Averaged Perceptron Tagger as part of the Textblob package. N-grams were used to extract certain features from the dataset such as the frequency of certain words for positive and negative data.

Sentiment Analysis was completed with Vadar, (Valence Aware Dictionary and sentiment reasoner) and Sentiment Intensity Analyzer from the nltk package. Since this tool is a lexicon and operates on rule based sentiments, it is particularly suited to social media language, which is appropriate for this dataset. Applying the vadar compound Thesholds were set to <+0.5

**Data processing consisted of:**

-word\_tokenize from nltk.tokenize was used for word tokenization, which splits text into individual words

-FreqDist from nltk.probability was used to calculate the frequency distribution of words, identify common rare words

- pos\_tag function to perform part-of-speech tagging on a list containing only the last token. The result is a list with a single tuple containing the last token and its part-of-speech tag, the pos\_tags list generated, which contains tuples of words and their corresponding part-of-speech tags

-chunking and vader sentiment to add a vader compound and sentiment (positive/negative/neutral) to the dataset as columns