CS410 Fall 2023 - Project Progress Report
Project: Yelp Customer Review Sentiment Analysis
Team Name: CaffeineCrew

Team members:

- Goutam Debnath (Captain) - goutamd2@illinois.edu
- Naveen Baskaran - nc42@illinois.edu
- Rohit Narula - rnarula2@illinois.edu
- Umesh Kumar - umesh2@illinois.edu

1) Which tasks have been completed?

❖ Data Collection & format Conversion

- ➢ The first task to build the tool was to collect the data and review its format. Yelp provides access to a collection of datasets through their Yelp Open Dataset program. We downloaded the JSON format of the data from the site.
- ➢ Upon inspection of the data downloaded, we noticed there were several files focusing on different areas, the only file of interest for the project was the yelp_academic_dataset_review.json file.
- ➢ The JSON file was too large for a text editor. We used the Python Pandas library to create a CSV file out of the JSON file.

```
import pandas as pd
yelp_json_path = 'data/yelp_academic_dataset_review.json'
df_b = pd.read_json(yelp_json_path, lines=True)
df_b.head()
csv_name = "data/yelp_dataset_review.csv"
df_b.to_csv(csv_name, index=False)csv_name = "data/yelp_dataset_review.csv"
df_b.to_csv(csv_name, index=False)
```

❖ Raw Data Analysis

- ➢ We decided to use OpenRefine for initial analysis because it is capable of loading large amounts of data / large files.
- ➢ We were able to load the data into OpenRefine for analysis. OpenRefine was used to do initial review and view the columns of interest most importantly the review format.
- ➢ We decided to focus on 20000 rows as a sample for our initial analysis as the model training will take a long time with the downloaded review data which is huge (~6 GB in size).

❖ Data Exploration:

Yelp dataset contains 20,000 reviews with the following attributes:

1. business_id (Unique ID of the business being reviewed)
2. date (Date the review was posted)
3. review_id (Unique ID for the posted review)
4. stars (1–5 rating for the business)
5. text (Review text)
6. type (Type of text)
7. user_id (User's id)
8. {cool / useful / funny} (Comments on the review, given by other users)
9. Topic: type of product being reviewed, i.e goods, services

```
In [4]: yelp.head()
```

Out[4]:

	business_id	date	review_id	stars	text	type	user_id	cool	useful	funny	topic
0	9yKzy9PApeiPPOUJEtnvkg	1/26/11	fWKvX83p0-ka4JS3dc6E5A	5	My wife took me here on my birthday for breakf...	review	rLtl8ZkDX5vH5nAx9C3q5Q	2	5	0	goods
1	ZRJwVLyzEJq1VAihDhYiow	7/27/11	IjZ33sJrzXqU-0X6U8NwyA	5	I have no idea why some people give bad review...	review	0a2KyEL0d3Yb1V6aivbluQ	0	0	0	service
2	6oRAC4uyJCsJl1X0WZpVSA	6/14/12	IESLBzqUCLdSzSqm0eCSxQ	4	love the gyro plate. Rice is so good and I als...	review	0hT2KtfLiobPvh6cDC8JQg	0	1	0	goods
3	_1QQZuf4zZOyFCvXc0o6Vg	5/27/10	G-WvGaISbqqaMHINnByodA	5	Rosie, Dakota, and I LOVE Chaparral Dog Park!!...	review	uZetl9T0NcROGOyFfughhg	1	2	0	service
4	6ozycU1RpktNG2-1BroVtw	1/5/12	1uJFq2r5QfJG_6ExMRCaGw	5	General Manager Scott Petello is a good egg!!!...	review	vYmM4KTsC8ZfQBg-j5MWkw	0	0	0	service

❖ Data Plotting & Understand relationships among the features:

To figure out the relationship between the length of the text review and stars received, we performed the following steps:

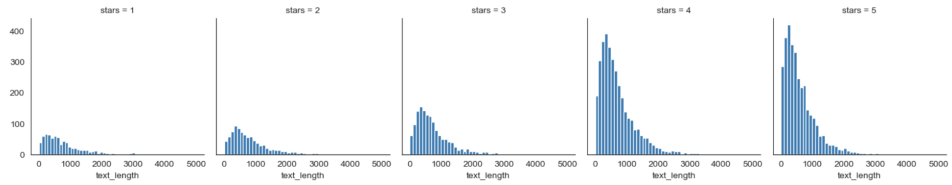
	business_id	date	review_id	stars	text	type	user_id	cool	useful	funny	topic	text_length
0	9yKzy9PApeiPPOUJEtnvkg	1/26/11	fWKvX83p0-ka4JS3dc6E5A	5	My wife took me here on my birthday for breakf...	review	rLtl8ZkDX5vH5nAx9C3q5Q	2	5	0	goods	889
1	ZRJwVLyzEJq1VAihDhYiow	7/27/11	IjZ33sJrzXqU-0X6U8NwyA	5	I have no idea why some people give bad review...	review	0a2KyEL0d3Yb1V6aivbluQ	0	0	0	service	1345
2	6oRAC4uyJCsJl1X0WZpVSA	6/14/12	IESLBzqUCLdSzSqm0eCSxQ	4	love the gyro plate. Rice is so good and I als...	review	0hT2KtfLiobPvh6cDC8JQg	0	1	0	goods	76

[1]

Sample Yelp Review Data Set

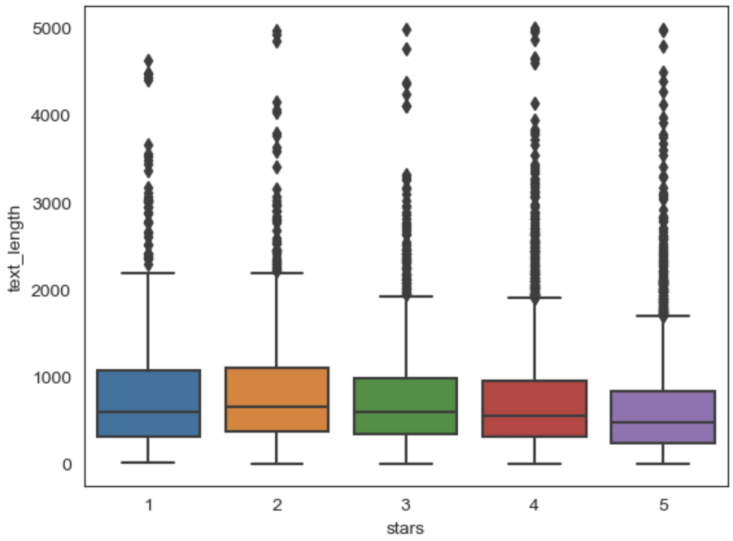
[1]

```
<seaborn.axisgrid.FacetGrid at 0x1448ee4d0>
```



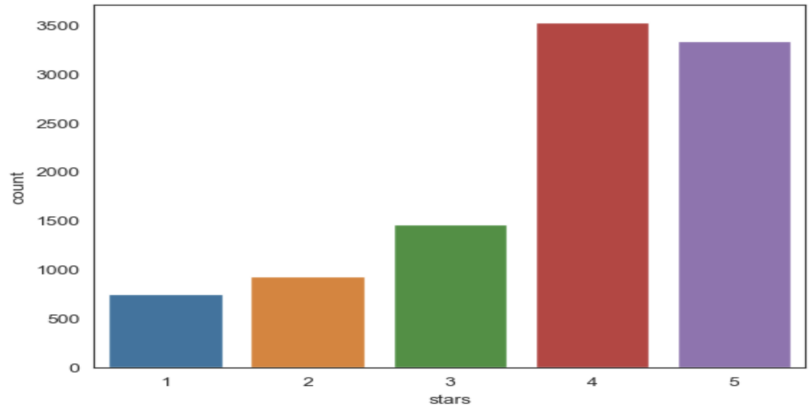
Created a plot between stars text length of reviews

```
Out[7]: <Axes: xlabel='stars', ylabel='text_length'>
```



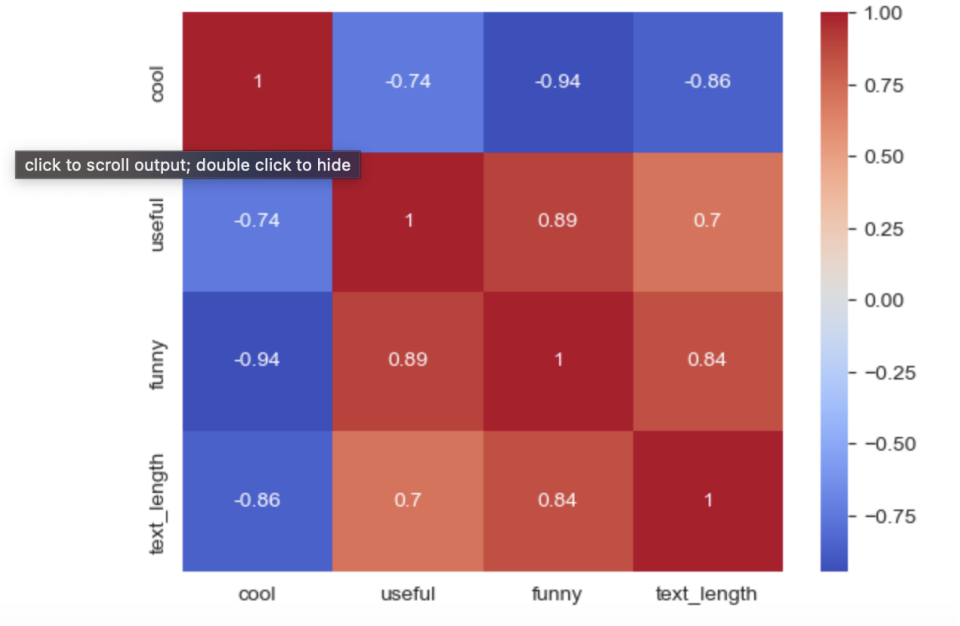
Box plot to correlate stars to text length of reviews

```
Out[8]: <Axes: xlabel='stars', ylabel='count'>
```



Created a Bar chart to count occurrences of different review categories (1 – 5)

```
In [11]: sns.heatmap(stars.corr(),cmap='coolwarm', annot=True)
Out[11]: <Axes: >
```



HeatMap to correlate review with category (cool, useful or funny)

❖ Data Preprocessing:

1. Clean the data by removing irrelevant information, such as special characters, and punctuation.
2. Convert text to lowercase to ensure consistency.
3. Tokenize the text into words or phrases using NLTK library
4. Remove stop words (common words like "and," "the," "is") that don't contribute much to sentiment.
5. Used Porter Stemmer from nltk for suffix stripping

```
def preprocess_text(text):
    text = re.sub(r'[^a-zA-Z\s]', '', text)
    text = text.lower()
    tokens = nltk.word_tokenize(text)
    tokens = [word for word in tokens if word not in set(stopwords.words('english'))]

    # Apply Porter Stemmer
    stemmer = PorterStemmer()
    stemmed_tokens = [stemmer.stem(word) for word in tokens]

    preprocessed_text = ' '.join(stemmed_tokens)
    return preprocessed_text
```

❖ Data Splitting:

Split the dataset into training and testing sets. We used 80% for training and 20% for testing.

- **Feature Extraction:**
  Converted the text data into numerical format. We utilized TF-IDF (Term Frequency-Inverse Document Frequency) by importing TfidfVectorizer from scikit-learn to vectorize review text

2) Which tasks are pending?

- Classification and Evaluation
  - **Model Selection and Training:**
    We are currently working on training the model using below four machine learning methods for Classification
    - Support Vector Classifier
    - Multinomial Naïve Bayes
    - Logistic Regression
    - Random Forest
  - **Model Evaluation**:
    Evaluate the model's performance on the testing dataset using metrics such as accuracy, precision, recall, and F1 score
  - **Sentiment analysis** of reviews based on the trained model to help up label each review with the reviewers opinions of the business. It would be useful for the business to understand their current performance in terms of customer opinion with relation to the categories from the classification section. There are many APIs available to perform sentiment analysis calculations. We plan to use the NLTK and the scikit-learn libraries.

- **Data Visualization**
  - Finally, having classified each comment as either goods or service we will try to represent this data in an useful graph format so that it helps with easy decision making for the business. We plan to use the popular graphing python library matplotlib for this purpose.

- **Final Project Presentation**

3) Are you facing any challenges? None at the moment