Yelp Review Usefulness Analysis

Alip Yalikun
Department of Computer Science
Alexandria, VA, United States
ayalikun@vt.edu

Figure 1: Yelp Review with 5-star Rating

ABSTRACT

In this project, I explored what makes a Yelp review "useful" by analyzing 60,000 reviews from the Yelp open dataset. I tried to answer three different questions; what kind of writing makes a review useful? Does length of the review change the quality of the review itself? Does social network influence matters when it comes to the usefulness of the review? To answer those questions, I combined sentimental analysis, data exploration, social network metrics, and classifiers that predict if the review will receive at least one useful vote. I extracted the sentiment using VADER analyzer, constructed a social network using the user data provided from the dataset to find degree and centrality metrics. I then compared models with different classifiers including, Random Forest, Logistic regression, and LinerSVC. The finding shows that both sentiment and social network of users contribute to having a useful review.

1 Introduction

Reviews are everywhere, we make a lot of decisions based on the review of a place we want to visit, whether it is a restaurant, an escape room, or even a barbershop, we look at their review to see if we wanted to make the choice to go there. Platforms like Yelp provide detailed descriptions for each business, and most importantly, people leave reviews with their honest opinions on the business listing page. However, not all reviews are equally helpful, this project investigates what makes a Yelp review "useful" and if we can predict usefulness based on review's content, the reviewer's friend circles on the platform. Identifying helpful reviews could enhance the user experience and platform credibility. This study's approach integrates sentimental analysis, network analysis, and developing a predictive model to analyze trends using various classification techniques.

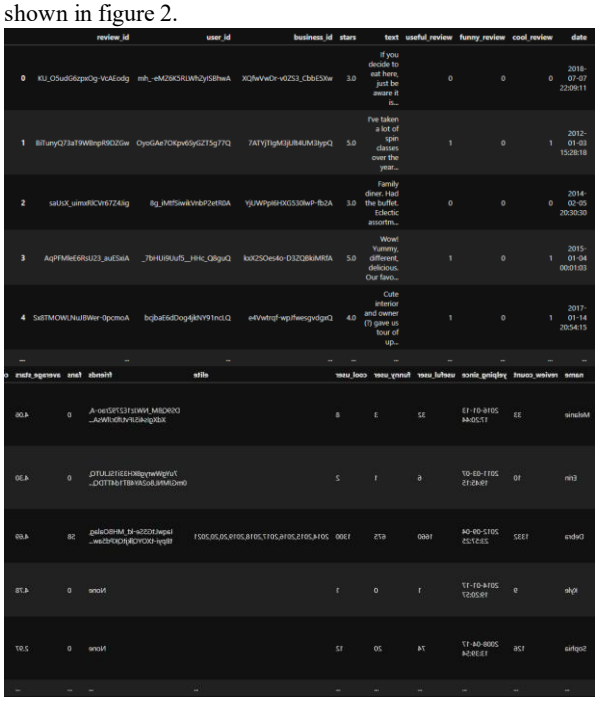
2 Background

Previous studies have shown that how people leave their review impacts the quality or usefulness of the review itself.

There are several factors like, lengths of the review, sentiment of the text, and the number of people they have influenced plays a key part in leaving useful reviews. McAuley and Leskovec (2013) did a study on product recommendations, where they discussed about the importance of review, they have found that more "topics" a user reviewed are considered as having a higher authorities or being more infomercial when recommending a product, in this case, the length of review can be determined as the number of topics a user reviewed, the higher the length of review, the better the other user would understand and find something that they would think is more helpful. My study is based on this foundation by combining text base sentimental analysis, user data, and network analysis within predictive models.

3 Approach

I used the Yelp open dataset available for academic research and other individual projects. This dataset contains about 8 gigabytes of information, so I extracted a sample of 60,000 reviews, user info data and merged them based on the user ID, shown in figure 2.



	review_id	user_id	business_id	stars	text	useful_review	funny_review	cool_review	date
0	KU_O5udG6zpxOg-VcAEodg	mh_-eMZ6K5RLWhZyISBhwA	XQfwVwDr-v0ZS3_CbbE5Xw	3.0	If you decide to eat here, just be aware it is...	0	0	0	2018-07-07 22:09:11
1	BiTunyQ73aT9WBnpR9DZGw	OyoGAe7OKpv6lyGZT5g77Q	7ATYjTIgM3jUBt4UM3IypQ	5.0	I've taken a lot of spin classes over the year...	1	0	1	2012-01-03 15:28:18
2	saUsX_uimxRlCVr67Z4Jig	8g_iMtfSiwikVnbP2etR0A	YjUWPpI6HXG530lwP-fb2A	3.0	Family diner. Had the buffet. Eclectic assortm...	0	0	0	2014-02-05 20:30:30
3	AqPFMeE6RsU23_auESxiA	_7bHUi9Uuf5__HHc_Q8guQ	kxX2SOes4o-D3ZQBkiMRfA	5.0	Wow! Yummy, different, delicious. Our favo...	1	0	1	2015-01-04 00:01:03
4	SdTMOWLNuJBWer-0pcmoA	bcjbaE6dDog4jkNY91ncLQ	e4Vwtrqf-wpJfwesgvdgxQ	4.0	Cute interior and owner (?) gave us tour of up...	1	0	1	2017-01-14 20:54:15

Figure 2: Data frame of initially selected dataset

Those initial data were then cleaned using Python Re library. Afterwards, I used VADER from NLTK python library. Following that, sentiment analysis was then ran on the cleaned text of reviews, after the model stopped running, I got the "neu", "pos", "neg", and "compound" scores. Each represents a sentiment as follow, neutral, positive, and negative, compound score is the overall calculation on the score of each sentiment. Anything ranging from 0 to 1 is considered positive sentiment, and -1 to 0 is considered negative sentiment, anything that is close to 0 is considered having a neutral sentiment. Figure 3 showcase the newly added features on the table.

clean_text	neg	neu	pos	compound
ting we usually opt for another diner or restaurant on the weekends in order to be done quicker	0.0	0.89	0.11	0.8491
from anyone russell always wears a smile on his face even when hes kicking your butt in class	0.05	0.707	0.243	0.9855
ttentive staff good place for a casual relaxed meal with no expectations next to the clarion hotel	0.035	0.709	0.257	0.9201
r you because we almost changed our mindsgo in and try something new youll be glad you did	0.0	0.674	0.326	0.9509
eeses lots of beer and wine as well as limited cocktails next time i will try one of the draft wines	0.017	0.71	0.272	0.9798
a frequent customer and great tipper glad that kanella just opened never going back to dmitris	0.133	0.672	0.195	0.5859
through the tour for a bathroom break if needed this tour was one of my favorite parts of my trip	0.031	0.814	0.155	0.9597
delicious nice selection of craft beers would definitely recommend checking out this hidden gem	0.0	0.477	0.523	0.9678
st looking pretty and not like a dumpster even cachuma lake looks like they put a bit more effort	0.2	0.677	0.123	-0.8995
was pretty good and im hoping it was just an off night here but for the money i wouldnt go back	0.016	0.743	0.242	0.9782
earned my satisfaction and loyalty as a customer very grateful for such a wonderful experience	0.1	0.754	0.146	0.9678
milltooth and its an amazing jewel of indianapolis im glade i had the chance to experience this	0.0	0.581	0.419	0.8442
e veal milanese but with mixed greens instead of pasta theyll modify the menu to suit your taste	0.0	0.8	0.2	0.8047
as very good too the service was very slow but despite this id go back the food is just that good	0.0	0.783	0.217	0.775
elo excellent tortilla salty melty cheese and great carne super cheap and you can drive through	0.045	0.725	0.23	0.9768
d home fries which were nice and smashed and crunchy friendly waitstaff will definitely be back	0.0	0.582	0.418	0.9477

Figure 3: Data frame where sentiment analysis was performed and labeled

After fining the sentiment analysis result, I then went ahead and created a social network using the existing data, where there is "friends" feature, I used it to create a new feature called "friends list" where it contain all the friends of that one user in a single list, then I mapped the number of the list to another field called "network degree" where it is total number of friendship that specific user has. In addition to that, I have also added degree centrality and degree closeness to the data frame so it will be easier to see which user is more central and has more influence than other users. In figure 4 the newly added fields are shown.

friends_list	network_degree	centrality_degree	centrality_closeness
2MXveozLalDl5oqt6_EQ']	1	1.978669938067631e-05	0.0821243972641308
70ZCK6Usom2eKo9kaQ']	2	3.957339876135262e-05	0.07375087902091755
_djPL_dHoq4TXm3hP6g']	4	7.914679752270524e-05	0.10317538915269826
['None']	0	0.0	0.0
['None']	0	0.0	0.0
0c33LlCxUt0Ju2lUDMw']	0	0.0	0.0
AM6o_-u1zKRtrqPewzA']	0	0.0	0.0
yiQvUu5mDqbTL4CNxA']	0	0.0	0.0
['None']	0	0.0	0.0
pbFoYEyYskuvM78Z3Q']	1	1.978669938067631e-05	0.06306192869028676
3BbKslozfH50rxmnMew']	1	1.978669938067631e-05	0.08501712636581375
['None']	0	0.0	0.0
_DggEFCbOfn1491Uijw']	4	7.914679752270524e-05	0.08523427171910788
5xR2VzZbxGBMpq_VRA']	9	.0001780802944268679	0.09417891552876671

Figure 4: Data frame where network degree and degree centrality add

In figure 5, I plotted a sub-graph to see what kind of network is going on, in terms of realistic design, I find it very accurate. It shows that only small amounts of users are connected to each other whereas other users have no friends or no friendship with any other users. This may be the reason for the irrelevancy of having a friend on a review-based mobile application.

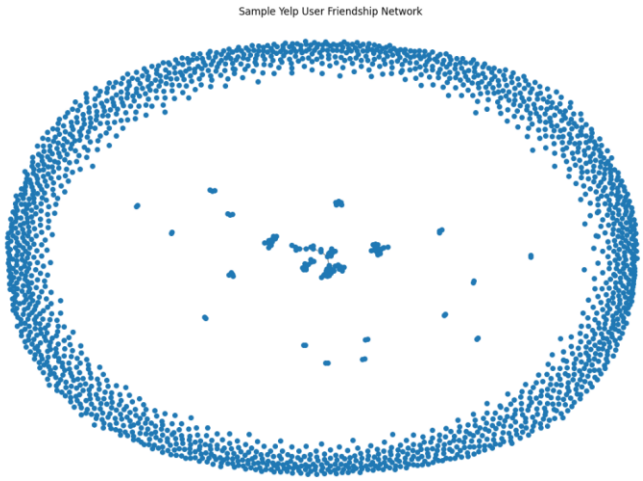


Figure 5: Sub-graph of 2000 nodes (users) and their social circle

Finally, the approach of using this dataset is concluded by adding a "review length" field in the data frame, those would later help me to get an answer to one of the questions I was trying to find out the answers for. From figure 6, I used the snippet code that would extract the text and find out about the length of the text with a lambda function.

```
merged_df = merged_df.copy()
merged_df['review_length'] = merged_df['clean_text'].apply(lambda x: len(x.split()))
```

pos	compound	friends_list	network_degree	centrality_degree	centrality_closeness	is_useful	review_length
0.11	0.8491	2MXveozLaDI5oqtS_EQ']	1	1.978669938067631e-05	0.08212439726411308	0	10
0.243	0.9855	78ZCK6Usxm2eKo9kaQ']	2	3.957339876135262e-05	0.07375087902091755	1	15
0.257	0.9201	djPL_dHoq4TXm3hP6g']	4	7.914679752270524e-05	0.10317538915269826	0	5
0.326	0.9509	['None']	0	0.0	0.0	1	4
0.272	0.9798	['None']	0	0.0	0.0	1	9
0.195	0.5859	i0c33LlCxUl8Ju2tUDMw']	0	0.0	0.0	1	8
0.155	0.9507	AMl6o_-u1cKRlrqPewzA']	0	0.0	0.0	0	19
0.523	0.9678	yiOvUu5mDqbTL4CNxA']	0	0.0	0.0	1	7
0.123	-0.8995	['None']	0	0.0	0.0	1	19
0.242	0.9782	qbFoYEyYsluvM78Z3Q']	1	1.978669938067631e-05	0.06306192869028676	0	9
0.146	0.9078	i3BbKsIozfH50nxmnMew']	1	1.978669938067631e-05	0.08501712636581375	1	17

Figure 6: Python code snippet and last piece too add to the dataset before further processing.

4 Experiment

4.1 Dataset

As I mentioned before, the dataset was selected from the Yelp official website where they give out their dataset for students or researchers to do experiments on it.

1. This dataset contains two major JSON files which contain users and reviews data.
    a. User.json:
        i. user_id: Unique identifier for the user.
        ii. name: Username.
        iii. review_count: Number of reviews the user has written.
        iv. yelping_since: Date the user joined Yelp.
        v. useful, funny, cool: Total votes received on the user's reviews.
        vi. elite: Years the user was considered an elite Yelper
        vii. friends: List of user_ids this user is friends with.
        viii. average_stars: Average star rating given by the user.
    b. Review.json:
        i. user_id: Links to user.json.
        ii. business_id: Links to business.json.
        iii. stars: Star rating.
        iv. text: The review text.
        v. useful, funny, cool: Votes on the review.
        vi. date: Date of the review.
2. This dataset contains about 8.6 million reviews, and more than 1.9 million users, so to use this dataset fully it is impossible to run on my local machine, hence we cut down the data to 60000 entries.
3. The last ingredient needed for this study to work is to have a target variable, where I decided to create a field called "is useful" to figure out if the reviews have gotten any "useful" vote, where if there is at least one vote, "is useful" field would be 1 otherwise 0. The below figure snippet of code shows how it was done.

```
# target feature is_usful, based on if the userful review feature has 1 or more count
merged_df['is_useful'] = merged_df['useful_review'].apply(lambda x: 1 if x >= 1 else 0)
```

Figure 7: Coding snippet of having the target variable for this study.

4.2 What kind of writing makes a review useful?

This study contains multiple questions to get an answer from, one of which is what kind of writing would make the review more useful, I have some initial idea where higher the compound score, meaning the more positive the sentiment is the more useful the reviews would be.

1. I started with the finalized data frame where it contains every text that has been cleaned using Python Re library.
2. After the sentiment analysis, there are very mixture of reviews, which are ideal for learning, and it is good to have a more balanced dataset.
3. I plotted the fields I needed to find out my answer using histogram with KDE enabled using the Seaborn library.
4. Figure 7 shows the finalized plot, the histogram shows the distribution of compound sentiment scores for reviews, separated by if they were marked as "useful" where yes = 1, or not, where no = 0.
5. Some discoveries have been made, reviews that are marked as useful tend to have slightly lower sentiment scores than non-useful ones, meaning they are less overly positive and possibly more balanced or detailed.
6. Additionally, the non-useful reviews peaked very high at the sentiment score of 1.0, indicating that overly positive or

having extensive emotion may not always be considered to be helpful.

7. This does not support my initial idea or if the review has more positive sentiment, then it may be more useful. Instead, neutral or balanced sentiment is more likely to be voted useful, since those are the more detailed ones where people will read it and gain significantly more information than overly positive reviews.

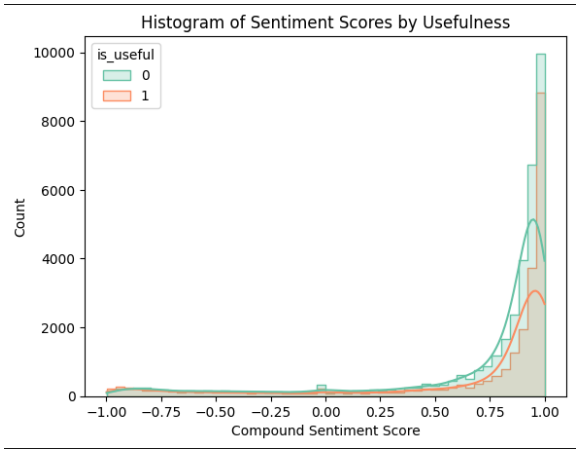


Figure 8: Histogram with KDE enabled plot using seaborn libraries on Python.

4.3 Does the length of the review make any difference?

Another question to be answered in this study is about the length of the review, and if the longer reviews can have more chance to be useful than the shorter reviews.

1. I started with the finalized data frame where it contains the length of the text.
2. I used a bar chart using the Python Seaborn library, where I plotted out the amount of usefulness per text length.
3. From the figure 9, the chart shows the average number of words in reviews that were marked as not useful, where “is useful” = 0, and useful, where “is useful” = 1.
4. Useful reviews in the orange bar have a significantly higher average word count compared to non-useful reviews.
5. This suggests that user values reviews that provide more context, elaboration, or any specific feedback. Whereas the shorter reviews were not as appreciated.

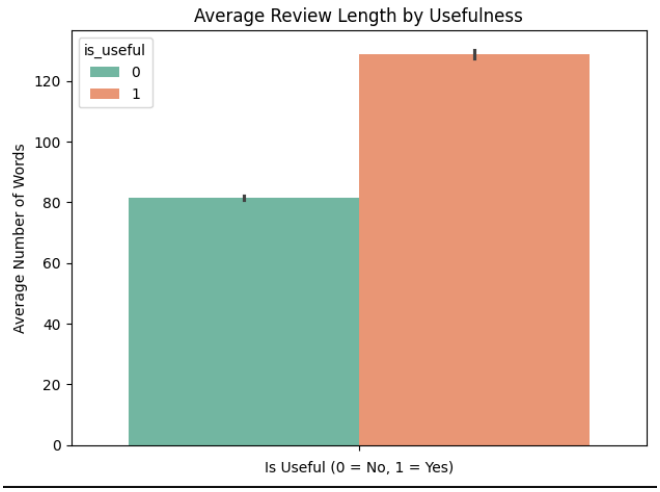


Figure 8: Bar chart plot using seaborn libraries on Python.

4.4 Does social network influence matters?

The last question tackles the network analysis aspect, to see if the number of friends you have indicates the usefulness of one’s review.

1. This question requires me to have a working social network, fortunately, the dataset contains user data with their friends, and I have made the network in previous sections.
2. I used a box plot from Python Seaborn library again to figure out the relationship between number of friends and usefulness of one’s review.
3. Figure 9 shows the distribution of the number of friends or network degree among users who wrote reviews that were either, marked useful where “is useful” = 1, and not useful where “is useful” = 0.
4. I decided to limit the network to 100 because it will give me better visualization.
5. Users with reviews marked as useful tend to have a higher median number of friends than those where their reviews were not useful.
6. The distribution shows a clear shift where the box for useful is wider and higher, validating more social network presence.
7. There are many outliers in both groups, bhut the spread is much larger among users with useful reviews, this suggest that more socially connected users who have more friends are more likely to w4rite reviews that get marked as useful.
8. This finding supports my initial idea where more friends equate to having a more useful review.

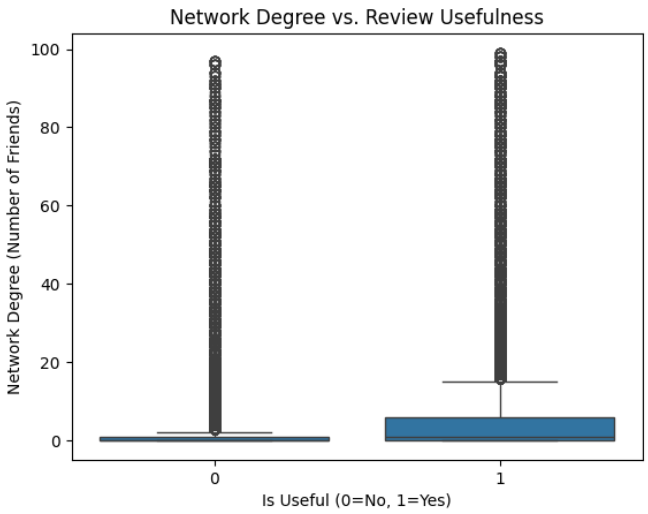


Figure 9: Box plot using seaborn libraries on Python.

4.5 Can we predict if a review is useful?

An additional question, I was thinking about whether we can use all those features, feed them into a classification model, to see if that feature is enough to make valid predictions on reviews being useful or not.

1. To do this task, I had to have a target variable, in this case I have decided to use "is useful" as the target variable as I discussed in earlier section.
2. Other features like 'compound', 'review_count', 'average_stars', 'network_degree', 'centrality_degree', 'centrality_closeness' are used to train the model.
3. I split the model with 80:20, where 80 percents are the training set and 20 percents are the test set.
4. I started with a random forest classifier, then followed by logistic regression classifier and finally a Liner SVC model.
5. Those models were then trained to get their metrics using a python function in figure 10 snippet code.

```
def get_metrics(model_name, y_true, y_pred):
    return {
        'Model': model_name,
        'Accuracy': accuracy_score(y_true, y_pred),
        'Precision': precision_score(y_true, y_pred),
        'Recall': recall_score(y_true, y_pred),
        'F1 Score': f1_score(y_true, y_pred)
    }

metrics = [
    get_metrics('Random Forest', y_test, y_pred_rf),
    get_metrics('Logistic Regression', y_test, y_pred_log),
    get_metrics('SVM (Linear)', y_test, y_pred_svc)
]
```

Figure 10: Code snippet of getting metrics function for each model.

6. Those metrics were then printed as a data frame (figure 11) and a heatmap (figure 12) to give a better insight.

	Model	Accuracy	Precision	Recall	F1 Score
0	Random Forest	0.626583	0.542897	0.406974	0.465211
1	Logistic Regression	0.646250	0.633333	0.269785	0.378386
2	SVM (Linear)	0.646500	0.614483	0.306536	0.409028

Figure 11: Metrics table of each model

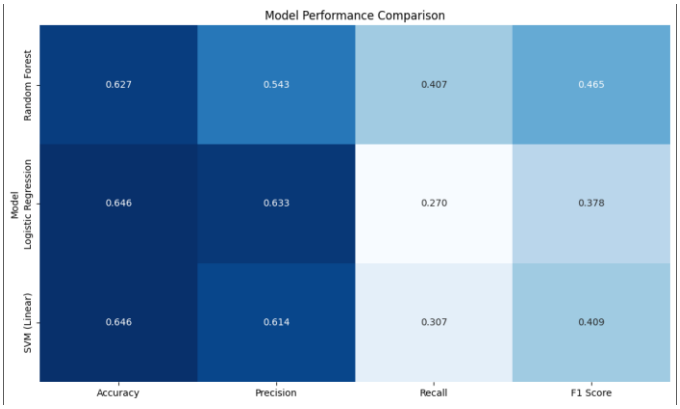


Figure 12: Metrics heatmap of each model

7. Accuracy is similar across each model, about 64%, suggesting consistent overall classification performance.
8. Precision is highest for Logistic regression with 0.633. It correctly identifies useful reviews more often, making it a better choice when false positives are there.
9. Recall is highest for Random Forest with 0.407. It catches more actual useful reviews, which is beneficial if I want to provide as many helpful reviews as possible.
10. F1 score is highest for random forests as well with 0.465, making it the best-balanced model among the three I used here.
11. Those results are not great; it suggests that there is room for improvement. It also means the predictive nature of those models is not good, and the features we used to decide usefulness of reviews are not correctly picked as well.
12. For future development, this study can be done by leveraging pre-train LLMs, using those that may provide better performance around all the key metrics.

5 Conclusion

In this study, I investigated what makes a Yelp review useful by running sentiment analysis, data exploration, and social network analysis, those came from analyzing 60000 reviews and their associated user profiles and friendship. I then extracted meaningful information like sentiment compound score of each review text, creating fields about length of the reviews, and formulating a social network with the existing data that contains the friend field, I then got the degree centrality of the network.

After the successful data extractions and data exploration. I started to answer the questions I have come up with before doing this research. The first one is, what kind of writing makes a review useful, from this question I found out that not all sentiment with high compound score, meaning having high positivity indicates overall usefulness of the reviews, instead, those review with balanced sentiment and neutral sentiment are the ones with higher usefulness, those are the ones that are mostly users commenting about facts with clear tone. The second question is, does the length of the review matter? To answer that, I plotted out a bar chart comparing the average length of reviews, and the result was as I expected, the reviews with higher word counts tend to have a more useful nature. The last question I was trying to answer is, does the social network influence matters? This question is answered by using a box plot, validating my initial thoughts where the higher number of friends tend to have a better time in getting useful votes on their reviews.

Finally, I tried to see if it is possible to predict if the one review can be useful. To answer that, I ran three different classifiers, random forest, logistic regression, and Lenear SVC. The results of those models were not optimal, but it was not terrible, with accuracy among all models of 64%, it definitely needs more work into it, perhaps by using LLMs to significantly boost performance metrics.

REFERENCES

[1] J. McAuley and J. Leskovec, “Hidden factors and hidden topics: Understanding rating dimensions with review text,” RecSys, 2013.

[2] Scikit-learn developers. RandomForestClassifier. https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html. Accessed May 10, 2025.

[3] Scikit-learn developers. LogisticRegression. https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html. Accessed May 10, 2025.

[4] Scikit-learn developers. LinearSVC. https://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVC.html. Accessed May 10, 2025.

[5] Yelp. Yelp Open Dataset. https://www.yelp.com/dataset. Accessed May 10, 2025.

[4] C.J. Hutto. VADER Sentiment Analysis. https://github.com/cjhutto/vaderSentiment. Accessed May 10, 2025.

[8] NetworkX developers. NetworkX: Network Analysis in Python. https://networkx.org/documentation/stable/. Accessed May 10, 2025.

[10] Michael Waskom and the Seaborn Development Team. seaborn: statistical data visualization. https://seaborn.pydata.org/. Accessed May 10, 2025.

[11] Matplotlib Development Team. Matplotlib: Visualization with Python. https://matplotlib.org/stable/. Accessed May 10, 2025.