

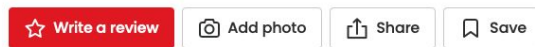
Decoding Customer Feelings With Machine Learning

CSCA 5622 Final

01

Project Intro

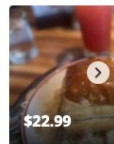
Yelp Reviews



Menu

Popular dishes

View full menu >



Order Food

Takeout Delivery

Start Order

wearefoundingfarmers.com

(202) 822-8783

Project Goal

Binary Classification of Yelp Reviews Using
Logistic Regression

02

Obtaining Data

Scraping Data

[illegible]

Cleaned Raw data

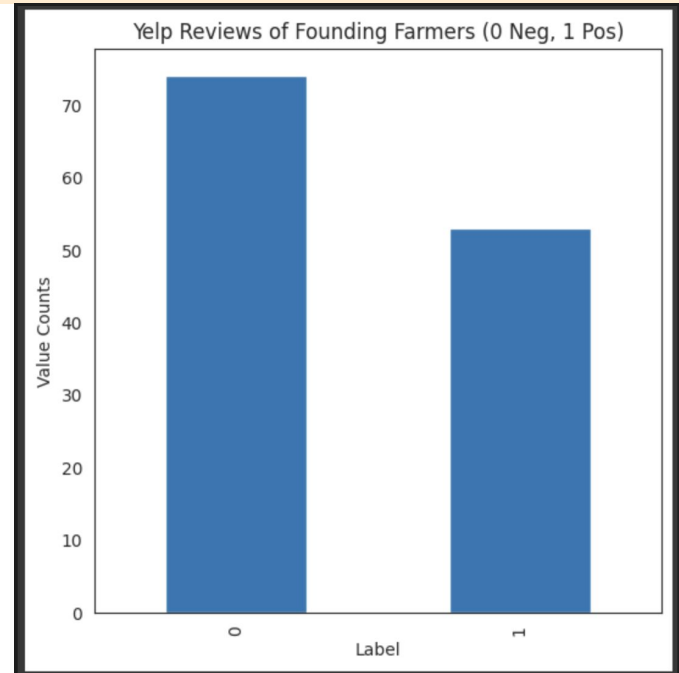
First few reviews:

	review_text	score
0	I am so shocked that I have to write this, but...	1
1	DO NOT BE MISLED BY THEIR REVIEW RATING--CONSI...	1
2	Abusive pricing system, overworked and not eno...	1
3	One of the most overrated (if not the most ove...	1
4	This review is for food only as I ordered take...	1

Review count by score:

score	count
5	28
2	26
4	26
1	25
3	25

Name: count, dtype: int64



03

Tokenize Data

Tokenize

```
1 df['Text_Cleaned'] = list(map(clean_text, df.review_text))
```

```
1 df.sample(n=5)
```

	review_text	score	Label	Text_Cleaned
66	This place has really gone downhill. I always ...	3	0	[place, really, gone, downhill, always, loved, ...
106	One of the best burgers I have ever had.Foundi...	5	1	[one, best, burgers, ever, founding, farmers, ...
86	Founding Farmers: Where the Fried Chicken Work...	4	1	[founding, farmers, fried, chicken, works, mag...
21	Very disappointed in my experience at this loc...	1	0	[disappointed, experience, location, tried, lo...
69	SO packed with people - over an hour wait for ...	3	0	[packed, people, hour, wait, table, opted, sit...

Lemmatize

```
4  
5 lemmatized_words(df.Text_Cleaned)
```

```
1 df.sample(n=5)
```

	review_text	score	Label	Text_Cleaned	lemmatized_text
79	Stopped by with some friends for their brunch ...	4	1	[stopped, friends, brunch, buffet, offered, su...	[stopped, friend, brunch, buffet, offered, sun...
58	I wanted to love this restaurant, but it fell ...	3	0	[wanted, love, restaurant, fell, short, really...	[wanted, love, restaurant, fell, short, really...
83	Great service and delicious food. I sat at the...	4	1	[great, service, delicious, food, sat, bar, gr...	[great, service, delicious, food, sat, bar, gr...
90	I decided to have dinner with the family at Fo...	4	1	[decided, dinner, family, founding, farmers, r...	[decided, dinner, family, founding, farmer, re...
116	Such a yummy experience! This was my first tim...	5	1	[yummy, experience, first, time, dining, disap...	[yummy, experience, first, time, dining, disap...

04

Word Cloud

Word Cloud

Word Cloud for Label 0



Word Cloud for Label 1



05

Model

Test Train Split + Pipeline

```
# Split data into train/test sets
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=5
)

# Define pipeline with vectorizer and classifier
pipeline = Pipeline([
    ('vectorizer', CountVectorizer()),
    ('classifier', LogisticRegression(max_iter=1000, solver='liblinear'))
])
```

GridSearchCV

```
# Hyperparameter grid
param_grid = {
    'vectorizer__ngram_range': [(1, 1), (1, 2)], # Test unigrams and bigrams
    'vectorizer__max_features': [1000, 2000], # Limit vocabulary size
    'classifier__C': [0.1, 1, 10], # Regularization strength
    'classifier__penalty': ['l1', 'l2'] # Regularization type
}

# Perform grid search with cross-validation
grid_search = GridSearchCV(
    pipeline,
    param_grid,
    cv=5,
    scoring='accuracy',
    n_jobs=-1
)
grid_search.fit(X_train, y_train)
```

Initial Output. Yikes!

```
Best Parameters: {'classifier__C': 0.1, 'classifier__penalty': 'l2', 'vectorizer__max_features': 1000, 'vectorizer__ngram_range': (1, 2)}  
Best Cross-Validation Accuracy: 0.78
```

```
Test Accuracy: 0.54
```

```
Classification Report:
```

	precision	recall	f1-score	support
0	0.60	0.60	0.60	15
1	0.45	0.45	0.45	11
accuracy			0.54	26
macro avg	0.53	0.53	0.53	26
weighted avg	0.54	0.54	0.54	26

```
Confusion Matrix:
```

```
[[9 6]  
 [6 5]]
```


Updated Grid

```
# Hyperparameter grid
param_grid = {
    'vectorizer__ngram_range': [(1, 1), (1, 2)], # Test unigrams and bigrams
    'vectorizer__max_features': [500, 1000, 1500, 2000], # Limit vocabulary size
    'vectorizer__max_df': [0.9, 1], # Ignore overly common words
    'classifier__C': [0.001, 0.01, 0.1, 1, 10, 100, 1000], # Regularization strength
    'classifier__penalty': ['l1', 'l2'] # Regularization type
}
```

Much Better

```
Best Parameters: {'classifier__C': 1000, 'classifier__penalty': 'l1', 'vectorizer__max_df': 0.9, 'vectorizer__max_features': 500, 'vectorizer__ngram_range': (1, 1)}
```

```
Best Cross-Validation Accuracy: 0.74
```

```
Test Accuracy: 0.69
```

```
Classification Report:
```

	precision	recall	f1-score	support
0	0.76	0.76	0.76	17
1	0.56	0.56	0.56	9
accuracy			0.69	26
macro avg	0.66	0.66	0.66	26
weighted avg	0.69	0.69	0.69	26

```
Confusion Matrix:
```

```
[[13  4]
 [ 4  5]]
```

06

Conclusion

Extracting the Features

```
# Get the best estimator from GridSearch
best_model = grid_search.best_estimator_

# Extract components
vectorizer = best_model.named_steps['vectorizer']
classifier = best_model.named_steps['classifier']

# Get coefficients (for class 1)
coefs = classifier.coef_[0]

# Get feature names
feature_names = vectorizer.get_feature_names_out()

# Create a DataFrame of features and coefficients
coef_df = pd.DataFrame({
    'feature': feature_names,
    'coef': coefs,
    'abs_coef': np.abs(coefs)
})

# Filter out zero-impact features
sparse_features = coef_df[coef_df['coef'] != 0]

sparse_features_sorted = sparse_features.sort_values('abs_coef', ascending=False)

# Top 10 positive-impact features (predict class 1)
print("Features predicting POSITIVE reviews:")
print(sparse_features_sorted[sparse_features_sorted['coef'] > 0].head(10))

# Top 10 negative-impact features (predict class 0)
print("\nFeatures predicting NEGATIVE reviews:")
print(sparse_features_sorted[sparse_features_sorted['coef'] < 0].head(10))
```

Features predicting POSITIVE reviews:			
	feature	coef	abs_coef
940	victor	5.861484	5.861484
372	group	4.452277	4.452277
420	interesting	4.375921	4.375921
266	enjoyable	3.865939	3.865939
30	also	3.716814	3.716814
473	love	3.584957	3.584957
33	amazing	3.487885	3.487885
740	setting	3.321240	3.321240
1	10	3.318695	3.318695
267	enjoyed	3.245979	3.245979

Features predicting NEGATIVE reviews:			
	feature	coef	abs_coef
408	husband	-7.448656	7.448656
141	check	-5.311083	5.311083
77	behind	-5.273368	5.273368
521	much	-4.527225	4.527225
453	like	-4.121783	4.121783
692	return	-3.598327	3.598327
798	spinach	-2.949052	2.949052
887	tried	-2.827501	2.827501
626	potato	-2.649925	2.649925
328	food	-2.570951	2.570951

Investigating Negative Features



B.W.

Las Vegas, NV

👤 0 🗣️ 9 📅 0

...



Nov 16, 2024

Food and staff were excellent! We were seated quickly on a Saturday night. **Victor** and crew had the table clear of glasses and plates quickly and the food was delicious! Thanks **Victor**!



Helpful 0



Thanks 0



Love this 0



Oh no 0



Kerry S.

Business Customer Service

Nov 18, 2024

Thank you for the wonderful review, B.W! We are glad you had a great experience, and we'll be sure to pass along your thanks to Victor and the team. We look forward to welcoming you back soon! -Kerry

...



Kerie C.

Bakersfield, CA

👤 0 🗣️ 4 📅 0



Nov 23, 2024

Had great service from **Victor** while visiting Founding Farmers. Gave us his time and suggestions on what he recommended as it was our first time coming!! All of our food/cocktails were great and we had the perfect little booth for the 3 of us! We will definitely be back! Thanks **Victor**



Yummy I.

New York, NY

👤 0 🗣️ 21 📅 11



Oct 20, 2024

DO NOT BE MISLED BY THEIR REVIEW RATING--CONSIDER THIS A WARNING.

As someone that's enjoyed the experiences at the Tysons location, the service we experienced today at this DC location was well below acceptable standards. For a restaurant with such strong reviews, the level of service was shockingly poor.

The food was mediocre at its best, delayed significantly. We had to ask for our appetizer, which came without utensils or napkins--another 10 minutes until we had to request them from another staff. A noticeable lack of attentiveness from the server, who didn't **check** on us or refill our water, despite our glasses being empty for some time. No syrup with pancakes until we had to wait for the server to walk by and stop the server to remind him. Two tables behind and front of us that were seated after us were served well before us. The overall experience was far from what I've come to expect. While the place was busy, that doesn't excuse the poor service. The entire experience left such a bad taste, particularly given the circumstances of hosting a farewell. This was a disappointing visit and not reflective of the standard one should expect from a restaurant with such reviews.

Future Work

TLDR: More Data!