Yelp Scraper

Yelp Reviews Extractor

```
csv_files = [
    'JPLicks_AssemblyRow.csv',
    'JPLicks_Cambridge.csv',
    'JPLicks_CharlesSt.csv',
    'JPLicks_OneBrighamCircle.csv'
]
```

```
store_reviews = pd.concat(
    [pd.read_csv(os.path.join(file_path, file)) for file in csv_files],
    keys=[os.path.splitext(file)[0] for file in csv_files],  # Add store name as key
    names=['Store', 'Index']
).reset_index(level=0)
```

```
print(store_reviews)
```

- AssemblyRow
- Cambridge
- Charles Street
- One Brigham Circle

```
filtered_reviews = store_reviews[['Store', 'Comment']]
```

```
print(filtered_reviews)
```

50 Reviews from each store

200 total comments collected,

77% positive

Index	Store	User Id	User Name	User Avatar
0	JPLicks_AssemblyRow	jlwCXjKcX5GwJvxgMCBxsA	Mojo S.	https://s3-media0.fl.yelpcdn.com/photo/J6HstwC...
1	JPLicks_AssemblyRow	mVDO6RJ3GvkUZ3nn2eE41w	Ame G.	https://s3-media0.fl.yelpcdn.com/photo/OAZVzSi...
2	JPLicks_AssemblyRow	NSS2x4EZVBHD4QSR3cDMbA	Jenny T.	
3	JPLicks_AssemblyRow	0mlZTI1wIWOck0jHwkVUGg	Danielle M.	
4	JPLicks_AssemblyRow	pLCnTMUsfgqvSk2Hyp2D1Q	Sophie T.	
...	...	...	...	
45	JPLicks_OneBrighamCircle	gyPkOnj9ETPJlTzjb-pS_w	Nicole Y.	
46	JPLicks_OneBrighamCircle	I4BbW4a19h2IItNOoToDgA	Christine D.	
47	JPLicks_OneBrighamCircle	uOtqnOv6m8JFixWsb_Kf6Q	Cori L.	
48	JPLicks_OneBrighamCircle	HUmClCl1uKP5Ur6X7e306Q	John L.	
49	JPLicks_OneBrighamCircle	uq4R7Uh_v3zGweAt06cqQA	Benjamin J.	

```
48    JPLicks_OneBrighamCircle
49    JPLicks_OneBrighamCircle


                                                Comment
Index
0
They say I'm a J.P. Licks fan but I know I'm not the only one.\n\nIt's been said that there are better ice cream shops out there, especially in the gr
eater Boston area. But honestly I haven't found any. As far as I've experienced, this is the pinnacle of bombass ice cream shops in the country. Every
time I try a new flavor and every time I try a new location, it just works for me.\n\nIt's like a comforting blanket of sweetened cream and sundry mix
-ins and chunky tidbits to lay on. How could ever I want to cheat on that with about ice cream shop? That's sacrilege. More than that, it's disrespect
ful to my first love of ice cream.\n\nI think once you find your ice cream joint, you shouldn't go looking for another. In fact, you should shout abou
t your love to everyone you come across. Tell them how great you have it and invite them to join in the love.\n\nIn that regard I hope someday you joi
n us and the lovers of ice cream will live as one.
1
It was a pleasure experiencing this Boston based ice cream shop. They have a sweet story told in a humorous manner on their website that I enjoyed rea
ding. This lovely ice cream shop had a good layout that made ordering quick and efficient on a hot summer day. The line was almost to the door, but th
e staff seemed to have it down to a science. \n\nFlavor samples were offered, which is always appreciated. This day, we decided upon the four scoop pl
```

Comments

Sentiment Analysis

```
store_reviews['Polarity'] = store_reviews['Comment'].apply(lambda x: TextBlob(x).sentiment.polarity)

store_reviews['Subjectivity'] = store_reviews['Comment'].apply(lambda x: TextBlob(x).sentiment.subjectivity)

store_reviews_summary = store_reviews.groupby('Store').agg({
    'Polarity': ['mean', 'std','min','max','var'],
    'Subjectivity': ['mean', 'std','min','max','var']
}).reset_index()
```

```
store_reviews['Sentiment'] = store_reviews['Polarity'].apply(lambda x: 'Positive' if x > 0.1
                                                              else 'Negative' if x < -0.1
                                                              else 'Neutral')

sentiment_summary = store_reviews.groupby('Store')['Sentiment'].value_counts(normalize=True).unstack()
```

	Store	Polarity				
		mean	std	min	max	var
0	JPLicks_AssemblyRow	0.170955	0.310287	-0.600000	1.000000	0.096278
1	JPLicks_Cambridge	0.297983	0.236140	-0.500000	0.900000	0.055762
2	JPLicks_CharlesSt	0.284757	0.264917	-0.365735	1.000000	0.070181
3	JPLicks_OneBrighamCircle	0.223748	0.232083	-0.412500	0.701786	0.053863

	Subjectivity				
	mean	std	min	max	var
0	0.547728	0.168412	0.223148	1.000000	0.028363
1	0.576353	0.141566	0.050000	0.900000	0.020041
2	0.595197	0.139129	0.337500	1.000000	0.019357
3	0.579382	0.105871	0.260000	0.916667	0.011209

Polarity

- Overall positive sentiment
- Assembly Row has more mixed reviews(high std)
- Cambridge has the most positive average sentiment

Sentiment	Negative	Neutral	Positive
Store			
JPLicks_AssemblyRow	0.14	0.18	0.68
JPLicks_Cambridge	0.04	0.10	0.86
JPLicks_CharlesSt	0.08	0.12	0.80
JPLicks_OneBrighamCircle	0.12	0.14	0.74

Subjectivity

- Mostly opinion-driven (mean closer to 1)
- Sentiment is consistent across all stores
- Charles St's customers share more personal opinions and emotions

Content Analysis

```
new_comments = store_reviews['Comment'].dropna()

all_text = " ".join(new_comments)

cleaned_text = re.sub(r'[^\w\s]', '', all_text.lower())

words = cleaned_text.split()
```

```
word_counts = Counter(words)

most_common_words = word_counts.most_common(50)

print(most_common_words)
```

```
filtered_words = [word for word in words if word not in stop_words]

filtered_word_counts = Counter(filtered_words)

most_common_filtered = filtered_word_counts.most_common(50)

print(most_common_filtered)
```

```
word_freq_df1 = pd.DataFrame(most_common_filtered, columns=['Word', 'Frequency'])
```

```
['they', 'say', 'im', 'a', 'jp', 'licks', 'fan', 'but', 'i', 'know', 'im', 'not', 'the', 'only', 'one', 'its', 'been', 'said', 'that', 'there', 'are',
'better', 'ice', 'cream', 'shops', 'out', 'there', 'especially', 'in', 'the', 'greater', 'boston', 'area', 'but', 'honestly', 'i', 'havent', 'found',
'any', 'as', 'far', 'as', 'ive', 'experienced', 'this', 'is', 'the', 'pinnacle', 'of', 'bombass', 'ice', 'cream', 'shops', 'in', 'the', 'country', 'ev
ery', 'time', 'i', 'try', 'a', 'new', 'flavor', 'and', 'every', 'time', 'i', 'try', 'a', 'new', 'location', 'it', 'just', 'works', 'for', 'me', 'its',
'like', 'a', 'comforting', 'blanket', 'of', 'sweetened', 'cream', 'and', 'sundry', 'mixins', 'and', 'chunky', 'tidbits', 'to', 'lay', 'on', 'how', 'co
uld', 'ever', 'i', 'want', 'to', 'cheat', 'on', 'that', 'with', 'about', 'ice', 'cream', 'shop', 'thats', 'sacrilege', 'more', 'than', 'that', 'its',
'disrespectful', 'to', 'my', 'first', 'love', 'of', 'ice', 'cream', 'i', 'think', 'once', 'you', 'find', 'your', 'ice', 'cream', 'joint', 'you', 'shou
ldnt', 'go', 'looking', 'for', 'another', 'in', 'fact', 'you', 'should', 'shout', 'about', 'your', 'love', 'to', 'everyone', 'you', 'come', 'across',
'tell', 'them', 'how', 'great', 'you', 'have', 'it', 'and', 'invite', 'them', 'to', 'join', 'in', 'the', 'love', 'in', 'that', 'regard', 'i', 'hope',
'someday', 'you', 'join', 'us', 'and', 'the', 'lovers', 'of', 'ice', 'cream', 'will', 'live', 'as', 'one', 'it', 'was', 'a', 'pleasure', 'experiencin
g', 'this', 'boston', 'based', 'ice', 'cream', 'shop', 'they', 'have', 'a', 'sweet', 'story', 'told', 'in', 'a', 'humorous', 'manner', 'on', 'their',
'website', 'that', 'i', 'enjoyed', 'reading', 'this', 'lovely', 'ice', 'cream', 'shop', 'had', 'a', 'good', 'layout', 'that', 'made', 'ordering', 'qui
ck', 'and', 'efficient', 'on', 'a', 'hot', 'summer', 'day', 'the', 'line', 'was', 'almost', 'to', 'the', 'door', 'but', 'the', 'staff', 'seemed', 't
o', 'have', 'it', 'down', 'to', 'a', 'science', 'flavor', 'samples', 'were', 'offered', 'which', 'is', 'always', 'appreciated', 'this', 'day', 'we',
'decided', 'upon', 'the', 'four', 'scoop', 'plate', '9', 'with', 'the', 'mint', 'chip', 'black', 'raspberry', 'cookies', 'n', 'cake', 'batter', 'and',
'the', 'maple', 'butter', 'walnut', 'it', 'was', 'some', 'of', 'the', 'best', 'tasting', 'ice', 'cream', 'weve', 'had', 'the', 'place', 'was', 'clea
n', 'the', 'staff', 'was', 'friendly', 'and', 'the', 'atmosphere', 'was', 'laid', 'back', 'outdoor', 'dog', 'friendly', 'patio', 'seating', 'is', 'ava
ilable', 'which', 'we', 'gladly', 'took', 'advantage', 'of', 'very', 'good', 'visit', 'with', 'very', 'good', 'ice', 'cream', 'thanks', 'jp', 'licks',
```

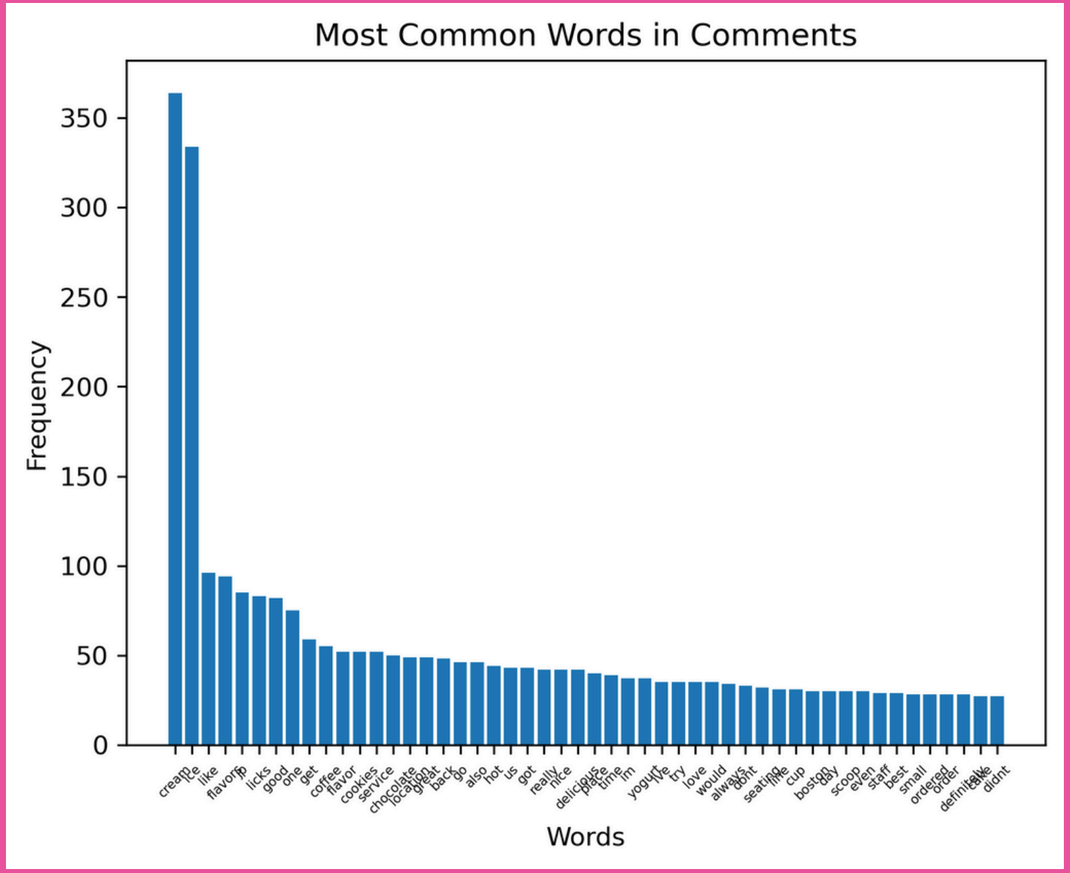
```
[('the', 830), ('and', 646), ('a', 476), ('i', 426), ('to', 407), ('cream', 364), ('ice', 334), ('of', 310), ('was', 309), ('it', 269), ('in', 229), ('fo
r', 195), ('is', 186), ('but', 151), ('they', 144), ('this', 138), ('we', 137), ('that', 134), ('with', 129), ('so', 125), ('on', 120), ('my', 118), ('yo
u', 113), ('not', 107), ('have', 102), ('like', 96), ('flavors', 94), ('had', 87), ('jp', 85), ('me', 85), ('licks', 83), ('good', 82), ('their', 77),
('one', 75), ('were', 75), ('as', 71), ('its', 69), ('just', 69), ('be', 67), ('there', 64), ('out', 62), ('are', 60), ('get', 59), ('if', 59), ('here',
58), ('at', 56), ('coffee', 55), ('flavor', 52), ('cookies', 52), ('some', 52)]
```

```
[('cream', 364), ('ice', 334), ('like', 96), ('flavors', 94), ('jp', 85), ('licks', 83), ('good', 82), ('one', 75), ('get', 59), ('coffee', 55), ('flavo
r', 52), ('cookies', 52), ('service', 52), ('chocolate', 50), ('location', 49), ('great', 49), ('back', 48), ('go', 46), ('also', 46), ('hot', 44), ('u
s', 43), ('got', 43), ('really', 42), ('nice', 42), ('delicious', 42), ('place', 40), ('time', 39), ('im', 37), ('yogurt', 37), ('ive', 35), ('try', 35),
('love', 35), ('would', 35), ('always', 34), ('dont', 33), ('seating', 32), ('line', 31), ('cup', 31), ('boston', 30), ('day', 30), ('scoop', 30), ('eve
n', 30), ('staff', 29), ('best', 29), ('small', 28), ('ordered', 28), ('order', 28), ('definitely', 28), ('cake', 27), ('didnt', 27)]
```

	Word	Frequency
0	cream	364
1	ice	334
2	like	96
3	flavors	94
4	jp	85
5	licks	83
6	good	82
7	one	75
8	get	59
9	coffee	55

- Clean the comments
- Find out the count for each character
- Delete all the “stop words”
- Create a list

Data Visualisation



Customers tend to focus on

- **Service**
- **Product Variety**
- **Objective Experience**
- **Good-Hearted**

```
top_words, top_counts = zip(*most_common_filtered)
```

```
plt.bar(top_words, top_counts)
plt.xlabel('Words')
plt.ylabel('Frequency')
plt.title('Most Common Words in Comments')
plt.xticks(rotation=45, fontsize=5)
plt.savefig('bar_chart.png', dpi=300, bbox_inches='tight')
plt.show()
```

```
wordcloud = WordCloud(width=800, height=400, background_color='white').generate(cleaned_text)
```

```
plt.figure(figsize=(10, 5))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
wordcloud.to_file('wordcloud.png')
plt.show()
```

- Customer Service
- Taste/flavor
- Employee/staff
- manager
- Coffee
- Hot chocolate
- Cookie
- Cake/cream
- Nice
- Great
- Love
- Best

Results & Recommendations

Focus on Consistent Service Quality

Insight: Assembly Row has more mixed reviews with high variability in sentiment.

Recommendation:

- Investigate specific complaints from this location and address them; while studying why other outlets are doing better.
- Consider hosting community-focused events to build stronger customer relationships.

Leverage Customer Sentiment for Marketing

Insight: Positive phrases like “nice,” “great,” “love,” and “best” frequently appear in customer reviews.

Recommendation:

- Highlight these phrases in promotional campaigns and customer testimonials to attract new patrons.
- Share “customer favorite” items (e.g., coffee, hot chocolate, cookies, cakes/cream) on social media.

Results & Recommendations

Invest in Product Feedback

Insight: Customers emphasize taste, flavor, and product variety.

Recommendation:

- Regularly collect feedback on new flavors or products to gauge customer interest and adapt offerings.
- Run limited-time seasonal flavors or customer-choice voting contests to increase engagement.

Improve Subjective Review Sentiment

Insight: Charles Street customers tend to be more emotionally driven in reviews.

Recommendation:

- Encourage customers to share positive stories or experiences through storytelling campaigns.
- Empower staff to create personalized experiences that evoke positive emotions.