Preferences on Asian Cuisines in Cities with most Population in the U.S. based on Twitter Tweets

1. Introduction

Asian cuisine includes several major regional cuisines: Central Asian, East Asian, North Asian, South Asian, Southeast Asian, and West Asian. A cuisine is a characteristic style of cooking practices and traditions, usually associated with a specific culture. Asia, being the largest and most populous continent, is home to many cultures, many of which have their own characteristic cuisine.

In Asia, different regions have a variety of styles of food associated with their unique cooking styles. People from different places also have various preferences in tastes. Asian food is also very popular in the U.S. So we want to know which type of Asian food is most popular in the U.S., especially in the several cities with the most population.

2. Prior Work

2.1. Cities and Types of Food Selected

Based on the list shown on tasteatlas (https://www.tasteatlas.com/100-most-popular-dishes-in-asia), we selected the 10 most common types of food in the U.S. for comparison within the 29 cities in the U.S. with the most population according to (https://worldpopulationreview.com/us-cities) to search for tweets.

Rank	Name	State	2022 Pop. ▼	2010 Census	Change	Density (mi²)	Area (mi²)
1	New York City	New York	8,177,025	8,190,209	-0.16%	27,222	300.38
2	Los Angeles	California	3,985,516	3,795,512	5.01%	8,499	468.96
3	Chicago	Illinois	2,671,635	2,697,477	-0.96%	11,750	227.37
4	Houston	Texas	2,325,353	2,100,280	10.72%	3,632	640.19
5	Phoenix	Arizona	1,759,943	1,449,038	21.46%	3,400	517.67
6	San Antonio	Texas	1,598,964	1,332,299	20.02%	3,296	485.11
7	Philadelphia	Pennsylvania	1,585,480	1,528,283	3.74%	11,807	134.28
8	San Diego	California	1,429,653	1,305,906	9.48%	4,387	325.88
9	Dallas	Texas	1,348,886	1,200,350	12.37%	3,970	339.74
10	Austin	Texas	1,028,225	806,164	27.55%	3,214	319.94
11	San Jose	California	1,003,120	954,940	5.05%	5,642	177.81
12	Fort Worth	Texas	958,692	748,441	28.09%	2,774	345.58
13	Jacksonville	Florida	938,717	823,114	14.04%	1,256	747.47
14	Charlotte	North Carolina	925,290	738,444	25.30%	3,012	307.24
15	Columbus	Ohio	921,605	790,943	16.52%	4,204	219.20
16	Indianapolis	Indiana	892,656	821,579	8.65%	2,469	361.57
17	San Francisco	California	884,108	805,505	9.76%	18,850	46.90
18	Seattle	Washington	787,995	610,630	29.05%	9,396	83.86
19	Denver	Colorado	760,049	603,359	25.97%	4,958	153.29
20	Washington	District of Columbia	718,355	605,226	18.69%	11,750	61.14
21	Boston	Massachusetts	696,959	621,048	12.22%	14,418	48.34
22	El Paso	Texas	687,287	650,671	5.63%	2,670	257.42
23	Nashville	Tennessee	682,262	604,589	12.85%	1,435	475.54
24	Oklahoma City	Oklahoma	676,492	582,516	16.13%	1,116	606.45
25	Las Vegas	Nevada	675,592	584,576	15.57%	4,766	141.77
26	Portland	Oregon	666,453	585,429	13.84%	4,995	133.42
27	Detroit	Michigan	661,193	711,131	-7.02%	4,767	138.72
28	Memphis	Tennessee	650,980	652,326	-0.21%	2,051	317.36
29	Louisville	Kentucky	615,067	596,482	3.12%	2,335	263.43
30	Milwaukee	Wisconsin	586,503	594,865	-1.41%	6,098	96.18

2.2. Python Packages Required

Required Python Packages:

twitter: https://pypi.org/project/twitter/

nltk: https://pypi.org/project/nltk/

textblob: https://pypi.org/project/textblob/

3. Process Flow

1. Find the target cities and target types of food
2. Find useful python package

3. Coding
4. Test output
5. Make data report
6. Write the analysis and report

4. Data

4.1. Calling the API

Twitter API is the official API provided by Twitter that allows developers to send requests to and receive responses. We use this to search for tweets with keywords of the food types we want to analyze within the cities we assigned.

Here are the necessities for requesting data from the Twitter API:

- API Keys from the Twitter Developer Platform
    - This can be applied by creating a Twitter account and creating a new app on the Twitter Developer Portal.
- Python package “twitter”
    - This is an API toolset for Twitter API.
    - This can be installed using pip.
- Chapter 9 - Twitter Cookbook.py (Python code from Chapter 9 of Mining the Social Web, 3rd Ed.)

4.2. API Limitations

Twitter Standard Search API Rate Limits:

- Requests / 15-min window (user auth): 180
- Requests / 15-min window (app auth): 450

We did not purchase any higher levels of access on the Twitter Developer Platform, so the limits for us are the ones shown above.

When the limit is reached, it automatically waits and continues when the 15-min wait time is over.

4.3. API Methods Used

4.3.1. Search

This is the only API method we used in the project.
Standard Search API from Twitter: https://developer.twitter.com/en/docs/twitter-api/v1/tweets/search/api-reference/get-search-tweets
(Used within the 'twitter' package instead of calling the API directly)

We did not call this API directly. Instead, we call it using the "twitter" package:

```
twitter_api.search.tweets
```

We created a dictionary in the code that stores all the cities we want to search for tweets within including their names and their latitudes and longitudes, and a list that stores all types of food we want to search for as keywords.

Parameters used:

- `q` keyword of searching
- `geocode` specifies the searched tweets by users located within a given radius of the given latitude/longitude; format: `latitude,longitude,radius`; we set the latitudes and longitudes for each city in each request; we set the radius to `50mi` for all requests
- `count` number of tweets in each search request; maximum is 100; we set it to `100` for all requests

4.4. Where does the data go?

The data, which are all the tweets we searched using the API, are stored in the variable results_tweets. All these tweets will be analyzed afterwards. We also output them in JSON

format to a file called “all_tweets.txt” in the same directory as the code file. This will be overwritten each time the code runs and finishes. So running the code multiple times will not add newly fetched tweets to the file, nor will they be analyzed together with previously fetched tweets. The fetching and analysis processes are independent for every time of code execution. If the file does not exist when it runs, it will automatically create one.

5. Sentiment Analysis

5.1. What is Sentiment Analysis?

5.1.1. Overview

Sentiment analysis is also known as opinion mining. It is done by using the natural language processing technique to determine the data. It rates the data with a positive/negative/neutral score. It helps visualize people’s opinions, sentiments, evaluations, appraisals, attitudes, and emotions. Sentiment analysis is a collection of a lot of tasks including sentiment analysis, opinion mining, sentiment mining, subjectivity analysis, review mining, emotion analysis, effect analysis, and opinion extraction. Thus, sentiment analysis is a super powerful tool for us to do the evaluation with specific data sets.

5.1.2. History

The first sentiment analysis paper could be considered “Thumbs up? Sentiment Classification using Machine Learning Techniques” by Pang et al. It was published in 2002, and was the first paper to classify movie reviews with positives and negatives. Then sentiment analysis evolute with a more specific score system called polarity. The polarity score gathers the data’s feelings, emotions, urgency, and intentions. The level of positive and negative also become more accurate with time. It can classify as Very positive, Positive, Weakly positive, Neutral, Negative, and Very negative.

Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), Philadelphia, July 2002, pp. 79-86. Association for Computational Linguistics.

Thumbs up? Sentiment Classification using Machine Learning Techniques

Bo Pang and **Lillian Lee**
Department of Computer Science
Cornell University
Ithaca, NY 14853 USA
{pabo,llee}@cs.cornell.edu

Shivakumar Vaithyanathan
IBM Almaden Research Center
650 Harry Rd.
San Jose, CA 95120 USA
shiv@almaden.ibm.com

5.1.3. Importance

Humans have and care about emotions. They also have the need to express their feelings and emotions. Sentiment analysis is a powerful tool to understand the sentiments of humans. Sentiment analysis can process millions of data in a short time. Which is way faster than analysis by humans. The accuracy of programming sentiment analysis might not be as accurate as human control, but it could be improved by the algorithms. The algorithms are also customizable by different needs. People can use sentiment analysis tools to analyze more data than expected. It can analyze how people react to food, brand, movie, and all other things. It is also not limited to the platforms if the platform opens the option to collect data.

Sentiment analysis has three major benefits of using.

- Real-time analysis

  Sentiment analysis can identify emotions on real-time data, such as the rating of presidential candidates. It can help you get the information you want quickly and get the response within seconds.

- Sorting data

  Sentiment analysis helps you to select a set of data automatically. For example, you can load those people who have positive reviews about the product you are selling. Then you can know that these are your potential customers.

- Consistence

Sentiment analysis has a consistent scoring system, and the scores will not be changed by subjective or biased opinions. Thus the consistency and accuracy would be better than analysis done by humans.

5.2. Popular Sentiment Analysis Tools

5.2.1. NLTK

The Natural language Toolkit was created by the University of Pennsylvania in 2001.
“NLTK is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries, and an active discussion forum.”[1]

5.2.2. SpaCy

This is a new generation of Sentiment analysis released in 2015 by Explosion AI.
“Features: Support for 66+ languages, 73 trained pipelines for 22 languages, Multi-task learning with pretrained transformers like BERT, Pretrained word vectors, State-of-the-art speed, Production-ready training system.”[2]

5.2.3. TextBlob

TextBlob was created by Steven Loria in 2013.
“TextBlob is a Python (2 and 3) library for processing textual data. It provides a simple API for diving into common natural language processing (NLP) tasks such as part-of-speech tagging, noun phrase extraction, sentiment analysis, classification, translation, and more.”[3]

5.3. Prerequisites

5.3.1. Tokenization

The tokenization will break a stream of characters into tokens. Some words are delimiters, and some words are not. Some special characters like space, tab, and newline will not count as tokens since they will not help with the sentiment analysis.

5.3.2. Stopwords

In our project, we used “from nltk.corpus import stopwords”. Stopword is a collection of words that have no predictive power for classifying text. For example, “a, the, it, they”. Those words have no sentiments, thus we want to remove the Stopwords from the data so that the analysis will be more accurate.

5.3.3. Lemmatization

After the data is converted into tokens, we will do the lemmatization which will turn tokens into standard form. For example: turn “acts” to “act”, “books” to “book”. Basically, it will strip off the prefix or suffix, so the word will be turned into a known word from the dictionary.

5.4. Tools We Used

5.4.1. TextBlob

TextBlob is the Python sentiment analysis tool we used in our final project.

Features: Noun phrase extraction, part-of-speech tagging, sentiment analysis, classification, tokenization, word and phrase frequencies, parsing, n-grams, word inflection, lemmatization, spelling correction, WordNet integration.

Download TextBlob with the following commands

```
$ pip install -U textblob
$ python -m textblob.download_corpora
```

Or you can use the same method as we used in our project.

```
from textblob import TextBlob
```

We used textblob to help us get a polarity score. Then we weighted the score to our standard.

```
for tweet in tweets:
    count += 1
    lemma_text = lemmatize(tweet)
    analysis = TextBlob(lemma_text)
    total_popularity += analysis.sentiment.polarity

    if analysis.sentiment.polarity == 0:  # adding reaction of how people are reacting to find average later
        neutral_count += 1
    elif 0 < analysis.sentiment.polarity <= 0.4:
        weakly_positive_count += 1
    elif 0.4 < analysis.sentiment.polarity <= 1:
        positive_count += 1
    elif -0.6 < analysis.sentiment.polarity <= 0:
        weakly_negative_count += 1
    elif -1 < analysis.sentiment.polarity <= -0.6:
        negative_count += 1
```

5.4.2. NLTK

In our final project, we used this function to implement the 3 prerequisites of sentiment analysis. The first line will remove those words without predictive power(it, her, they, etc..). It would help to generate more accurate results.

The second line will tokenize the tweet data. Remove the special cases(spcace, tab, newline).

The third line will execute each token, and turn the words into the most basic form.

After those processes. The raw data is ready to be analyzed.

```
def lemmatize(input_text: str) -> str:
    stop = stopwords.words('english')
    tokens = tokenize.word_tokenize(input_text)
    tokens_filtered = [w for w in tokens if w.lower() not in stop and w.lower() not in string.punctuation]
    return ' '.join(tokens_filtered)
```

6. Outputs and Reports

We fetched 9821 tweets in total. With the consideration of deep data mining, we have counted the positive rate, weakly positive rate, neutral rate, weakly negative rate, and negative rate as the analytical aspects. We found that Sushi has the most tweets, which is 2733, as the first place in the whole ranking. Pad Thai only has 248 related tweets. We combined all the data collected as an overview of information.

Table 1

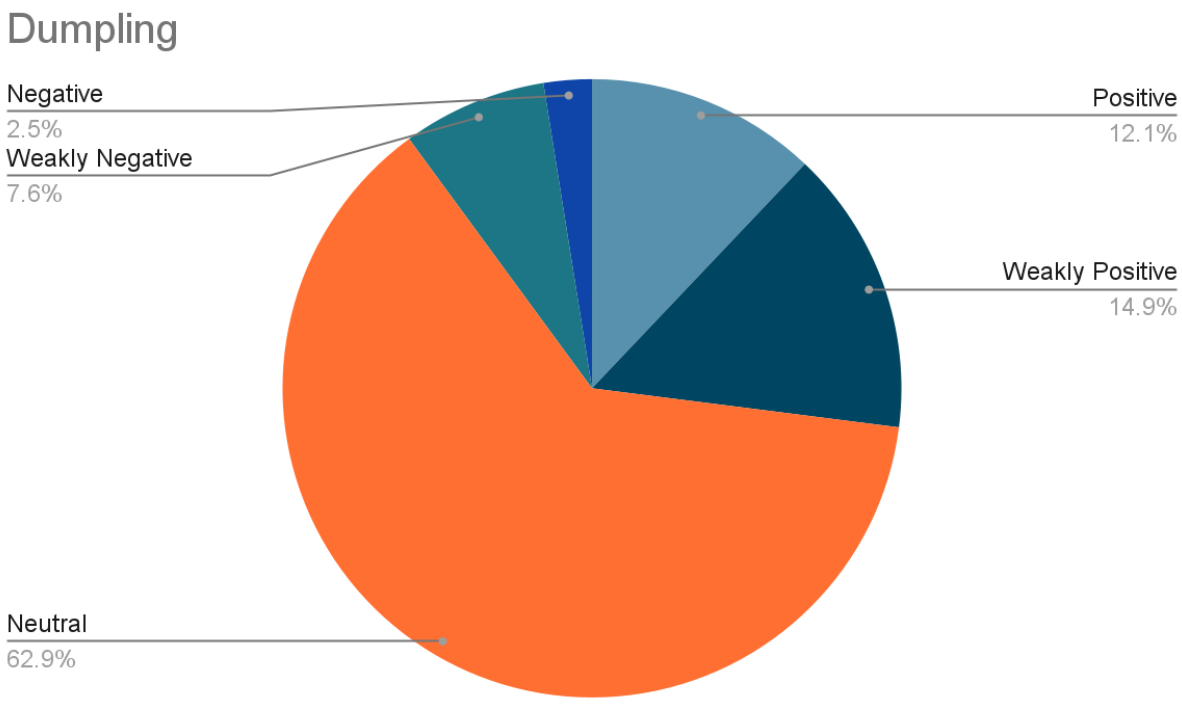
	Dumpling	Hot Pot	Sushi	Sashimi	Ramen

General Result	Weakly Positive	Weakly Positive	Weakly Positive	Weakly Positive	Weakly Positive
Positive Rate(%)	12.06	10.59	13.38	13.28	9.26
Weakly Positive Rate(%)	14.91	68.99	16.91	25.78	12.15
Neutral Rate(%)	62.81	12.14	55.83	48.83	68.11
Weakly Negative Rate(%)	7.54	8.01	10.38	10.55	8.88
Negative Rate(%)	2.51	0.26	3.19	1.56	1.56

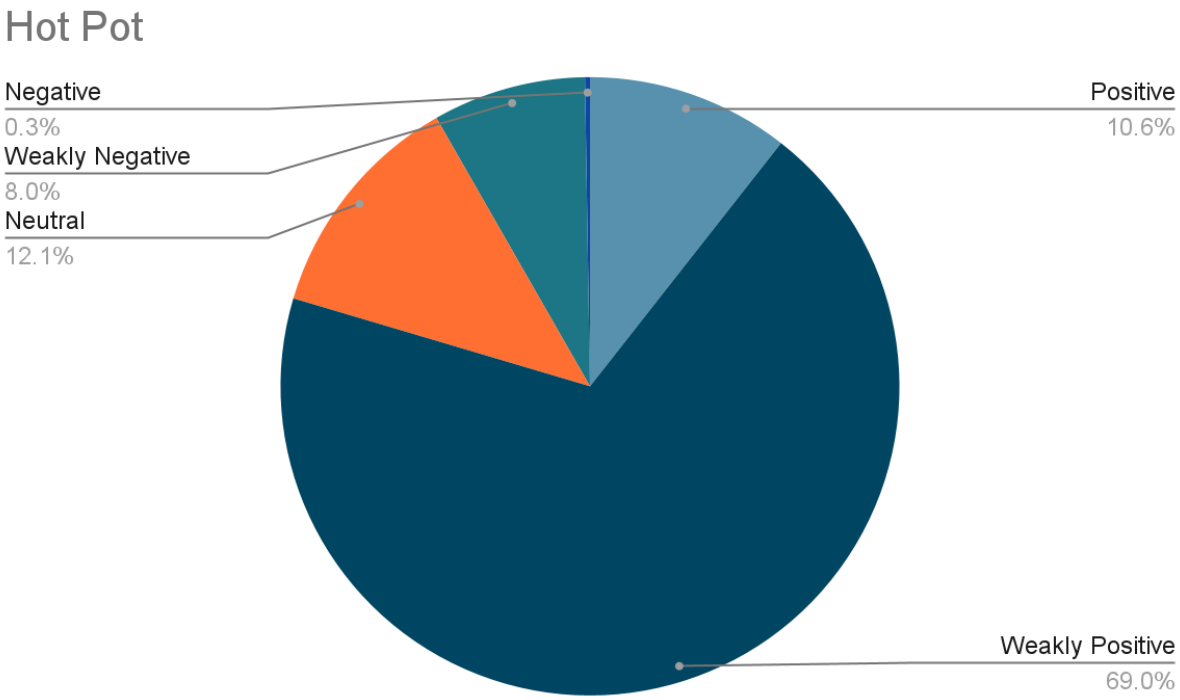
Table 2

	Pho	Kimchi	Green Curry	Pad Thai	Butter chicken
General Result	Weakly Positive	Weakly Positive	Weakly Positive	Weakly Positive	Weakly Negative
Positive Rate(%)	15.58	14.37	8.27	15.00	2.88
Weakly Positive Rate(%)	15.97	19.08	30.08	23.33	14.16

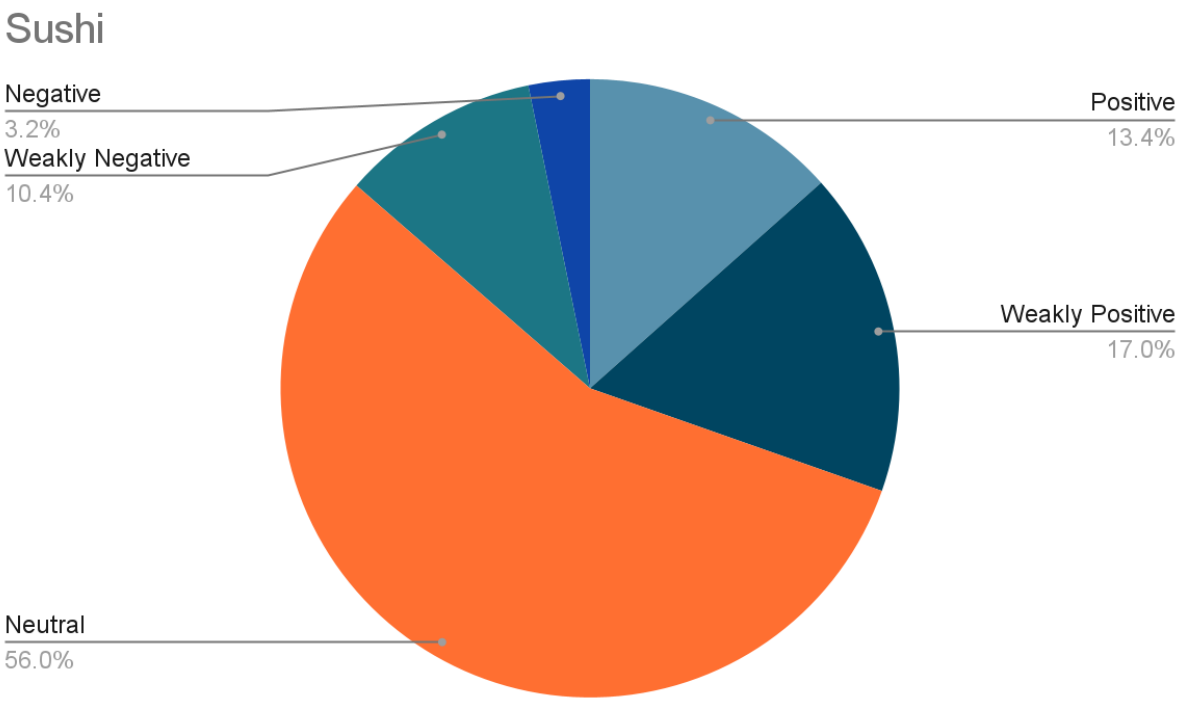
Neutral Rate(%)	57.27	56.77	18.05	49.17	5.53
Weakly Negative Rate(%)	8.47	7.66	42.86	7.92	36.50
Negative Rate(%)	2.72	2.12	0.75	4.58	40.93



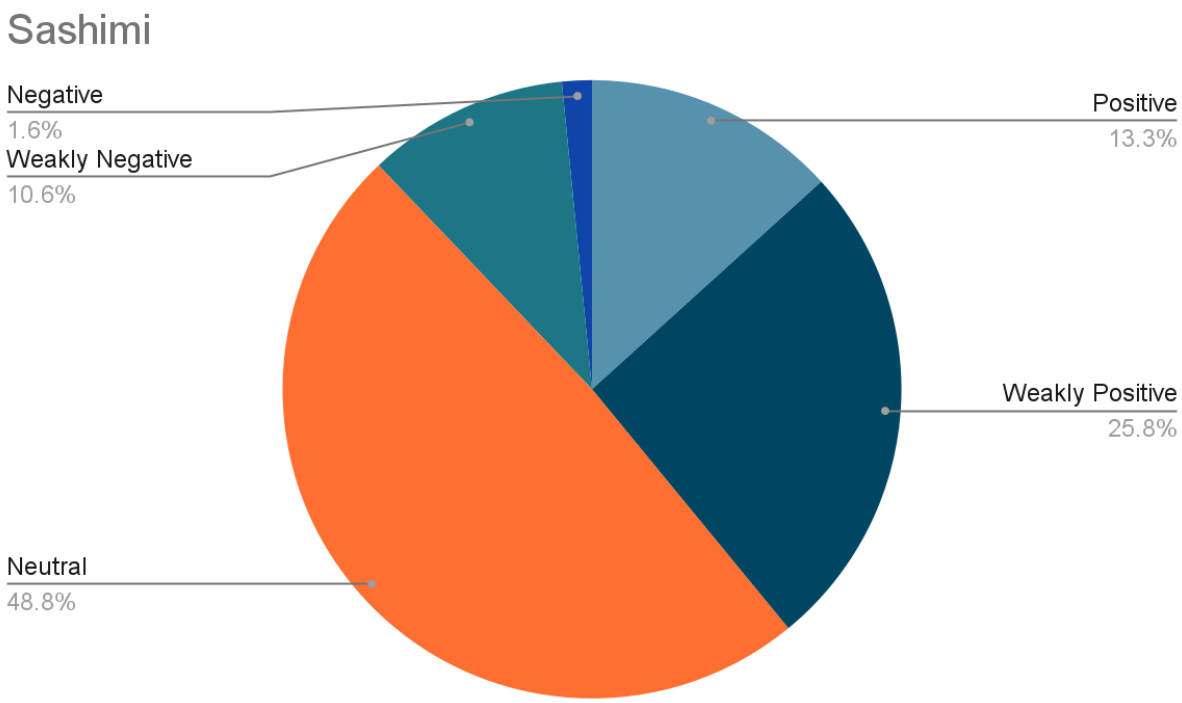
Dumpling: Weakly Positive



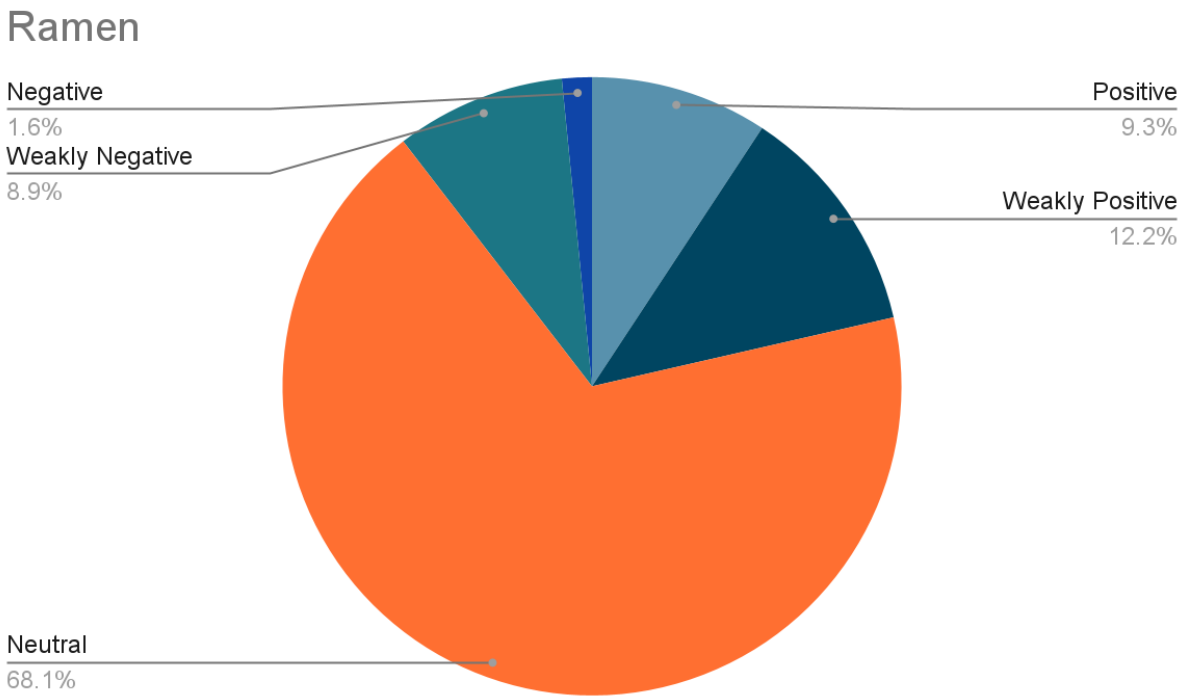
Hot Pot: Weakly Positive



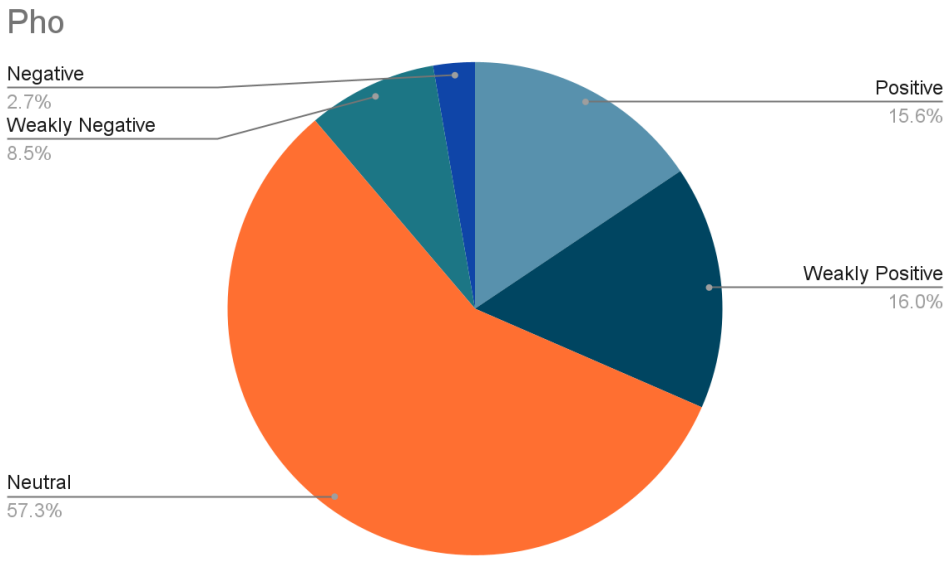
Sushi: Weakly Positive



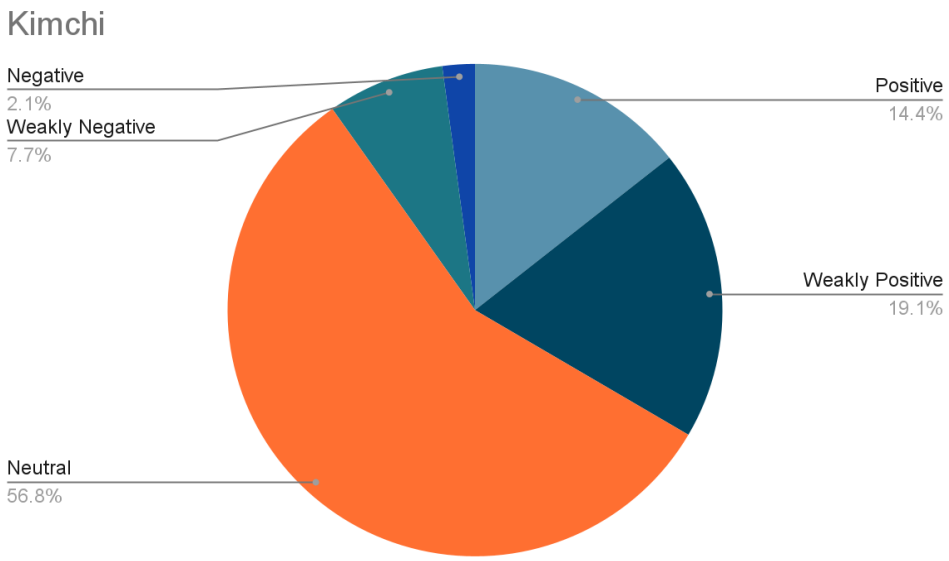
Sashimi: Weakly Positive



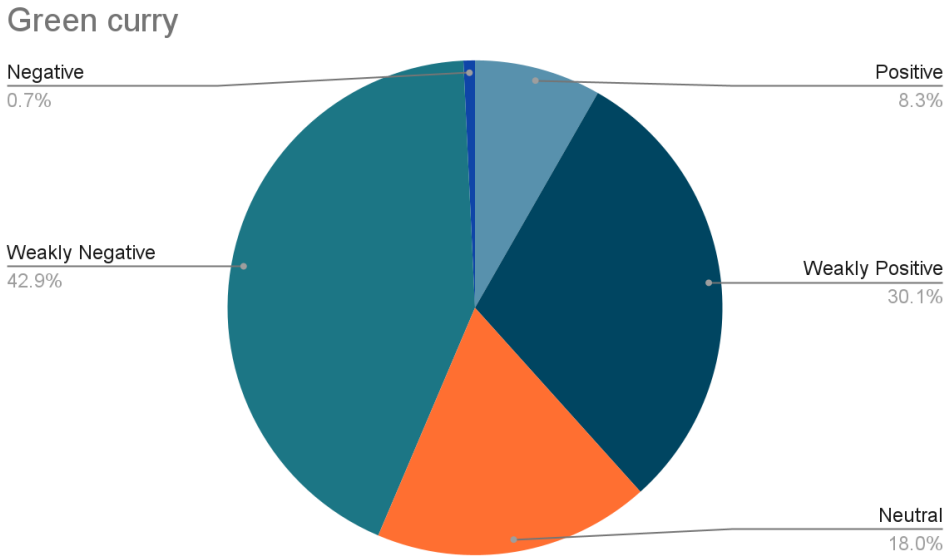
Ramen: Weakly Positive



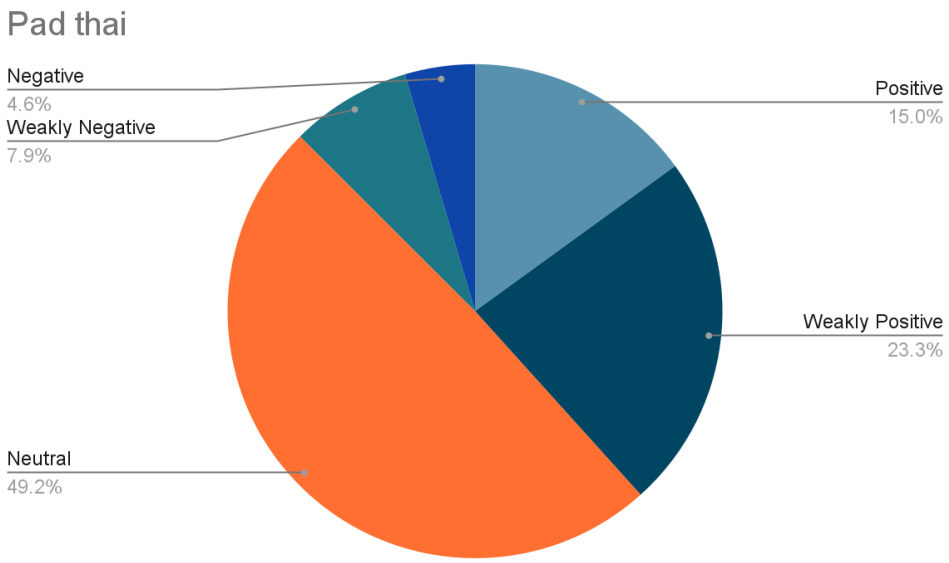
Pho: Weakly Positive



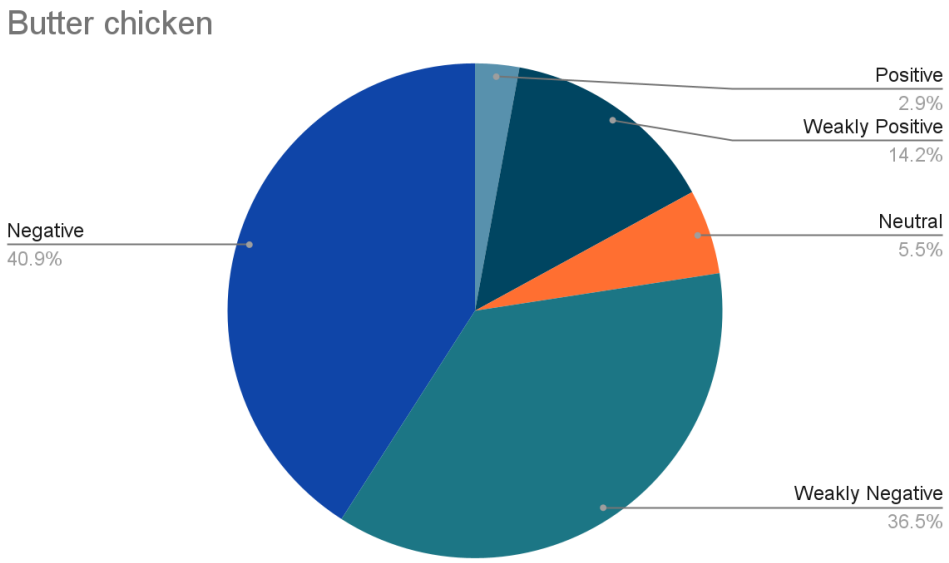
Kimchi: Weakly Positive



Green curry: Weakly Positive



Pad thai: Weakly Positive



Butter chicken: Weakly Negative

Almost all types of food are rated as “Weakly Positive”, except for Butter chicken, which is rated as “Weakly Negative”. Among all the foods, Pho has the most positive rate. For ramen, 68.11% of the tweets reserve neutral reviews. Hot Pot has the least negative rate of 0.26%.

For 7 of the 10 food types, at least around half of the tweets are neutral. And for most food, most tweets other than the neutral ones are either positive or weakly positive. This means that most tweets on Twitter about Asian food are relatively positive.

Here is the overall score of all the foods as a result table. (see table 3)

Table 3

	Score
Dumpling	100.39
Hot Pot	101.51
Sushi	96.69
Sashimi	100.07
Ramen	96.21
Pho	103.69
Kimchi	103.14
Green Curry	84.13
Pad Thai	99.32
Butter Chicken	49.36

As we can see, the overall ranking for the 10 types of food is: Pho > Kimchi > Hot Pot > Dumpling > Sashimi > Pad Thai > Sushi > Ramen > Green Curry > Butter Chicken.

Among the 10 types of food, Hot Pot is surprisingly welcomed by people. While all other types of food have most tweets that are neutral, about 68.99% of the tweets for Hot Pot are Weakly

Positive. And the Negative and Weakly Negative tweets only sum to 8.27%, which is also the lowest among all.

```
------------------------------------
Fetched 387 tweets on Hot pot
General Result: Weakly Positive
Positive Rate: 10.59%
Weakly Positive Rate: 68.99%
Neutral Rate: 12.14%
Weakly Negative Rate: 8.01%
Negative Rate: 0.26%
------------------------------------
```

The positive rate for Butter Chicken is low and the negative rate for it is abnormally high. In general, the negative rate for a kind of food is between 1% and 4%. However, Butter Chicken gets 40.93% on the negative rate. We also did some research on this. Possible reasons could be that Butter Chicken is less popular, or it could be that our sentimental analysis algorithm is defective.

Here are the quartiles and average for the scores:

Lowest	Q1	Median	Q3	Highest	Average
49.36	96.21	99.695	101.51	103.69	93.451

7. Additional Analyses

7.1. Different Results

There will be different results based on the tweets in the recent 7 days. Our ranking results are also subject to subtle changes. By changing the keyword in the project, the analysis of different kinds of food can be fulfilled.

7.2. Amount of Tweets Fetched

In general, The number of tweets that can be crawled by keywords may vary considerably from food to food. For example, there are 2631 tweets on Sushi, while there are only 240 tweets on Pad Thai. Just in terms of the number of crawls, Sushi might be more popular than Pad Thai. The main reason could be that Sushi is more common in life, and more people eat it and post it on social networks.

```
-------------------------------------
Fetched 2631 tweets on Sushi
General Result: Weakly Positive
Positive Rate: 13.38%
Weakly Positive Rate: 16.91%
Neutral Rate: 55.83%
Weakly Negative Rate: 10.38%
Negative Rate: 3.19%
-------------------------------------
```

```
-------------------------------------
Fetched 240 tweets on Pad thai
General Result: Weakly Positive
Positive Rate: 15.00%
Weakly Positive Rate: 23.33%
Neutral Rate: 49.17%
Weakly Negative Rate: 7.92%
Negative Rate: 4.58%
-------------------------------------
```

References

[1] NLTK: https://www.nltk.org/

[2] spaCy: https://spacy.io/

[3] TextBlob: https://textblob.readthedocs.io/en/dev/

[4] https://textblob.readthedocs.io/en/dev/quickstart.html#sentiment-analysis

[5] https://monkeylearn.com/sentiment-analysis/

[6] https://www.iflexion.com/blog/sentiment-analysis-python