Лабораторна робота №6

Аналіз тексту оглядів

In [1]:
```
# import the required libraries here
# two lines of code here:
import pandas as pd
import matplotlib.pyplot as plt
```

In [2]:
```
reviews = pd.read_csv('out/05_lab5_reviews_filtered.csv')
```

Згадаймо, який вигляд має сформована нами структура даних типу DataFrame. Перевіримо, чи читається файл reviews_filtered.csv.

In [3]:
```
reviews.head()
```

Out[3]:

	review_id	user_id	business_id	stars	useful	funny
0	fdiNeiN_hoCxCMy2wTRW9g	w31MKYsNFMrjhWxxAb5wIw	eU_713ec6fTGNO4BegRaww	4	0	0
1	Z7wgXp98wYB57QdRY3HQ3w	GYNnVehQeXjty0xH7-6Fhw	FxLfqxdYPA6Z85PFKaqLrg	4	0	0
2	svK3nBU7Rk8VfGorlrN52A	NJlxGtouq06hhC7sS2ECYw	YvrylyuWgbP90RgMqZQVnQ	5	0	0
3	4bUyL7lzoWzDZaJETAKREg	_N7Ndn29bpll_961oPeEfw	y-Iw6dZflNix4BdwIyTNGA	3	0	0
4	Amo5gZBvCuPc_tZNpHwtsA	DzZ7piLBF-WsJxqosfJgtA	qx6WhZ42eDKmBchZDax4dQ	5	1	0

Перший огляд

In [4]:
```
reviews.loc[0, 'text']
```

Out[4]:
```
'I\'ll be the first to admit that I was not excited about going to La Tavolta. Being a
food snob, when a group of friends suggested we go for dinner I looked online at the m
enu and to me there was nothing special and it seemed overpriced.  Im also not big on
ordering pasta when I go out. Alas, I was outnumbered. Thank goodness! I ordered the s
ea bass special. It was to die for. Cooked perfectly, seasoned perfectly, perfect port
ion. I can not say enough good things about this dish. When the server asked how it wa
s he seemed very proud of the dish and said, " doesn\'t she (the chef) do an incredibl
e job?" She does. \n\nMy hubby got the crab tortellini and also loved his. I heard "mm
mm this is so good" from all around the table. Our waiter was super nice and even gave
us free desserts because we were some of the last people in the restaurant. Service wa
s very slow and the place was PACKED but we had our jugs of wine and a large group wit
h good conversation so it didn\'t seem to bother anyone.\n\nSo-\n\nDo order the calama
ri and fried zucchini appetizers. Leave out the mussels. \n\nIf they have the sea bass
special, I highly recommend it. The chicken parm and crab tortellini were also very go
od and very big. The chicken Romano was a bit bland. The house salads were teeny. \n\n
Do make a reservation but still expect to wait for your food. Go with a large group of
people and plan for it to be loud. Don\'t go with a date unless you\'re fighting and d
on\'t feel like hearing anything they have to say.  Ask to sit in the side room if it
\'s available.'
```

Цей огляд у кращому випадку не є найкращим. З іншого боку, ми можемо побачити, що він говорить про піцу та італійську, що дає деяку впевненість у тому, що ми прийняли гарне рішення поєднати категорії pizza та italian.

Другий огляд

In [5]:
```
reviews.loc[1, 'text']
```

Out[5]:
```
"Wow. So surprised at the one and two star reviews!  We started with the most tender c
alamari. Although the marinara sauce was a bit bland, but a touch of salt made it just
right. My husband had the veal with peppers and said it was so delicious and tender. T
he mashed potatoes were perfect. I had the salmon Diablo which was also delicious. Our
salad was beautiful! Dressing was served on the salad and it was a nice amount. We end
ed our delicious meal with a piece of tiramisu. Our server Matt was right on!! Very pl
easant and knowledgeable about the menu. Our appetizer, salad and entrees were timed p
erfectly. I love salad and did not mind that my entree was served while I was still ea
ting it! No problem it let my dinner cool to just the right temp for me to eat it comf
ortably. \nI wonder sometimes if people just don't appreciate relaxing and taking time
to eat a wonderful and beautifully prepared meal.  A wonderful atmosphere. So relaxin
g. The chairs are super comfortable too!!! We will certainly be back. \nGive it a try.
  Don't  always go by the reviews. \nA bottle of Riesling, calamari app, two delicious
entrees and dessert for $92! \nWell with it."
```

Хороші та погані відгуки

Ми не хочемо читати всі відгуки в нашому наборі даних, щоб знайти хороші та погані. Очевидно, що 5 найкраще і 1 - це найгірше, але як їх згрупувати? Ми встановлюємо поріг посередині чи просто беремо верхній і нижній рейтинги? Давайте спочатку подивимось, як виглядає розподіл рейтингів.

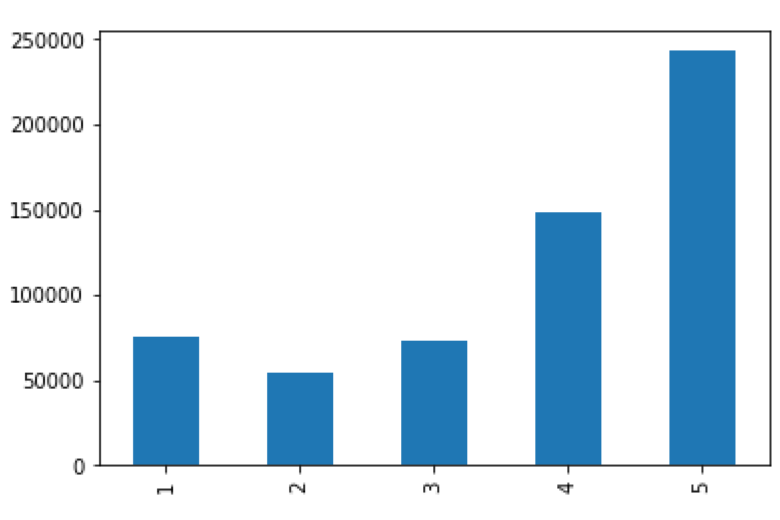
In [6]:
```
reviews['stars'].describe()
```

Out[6]:
```
count    594862.000000
mean          3.718837
std           1.403135
min           1.000000
25%           3.000000
50%           4.000000
75%           5.000000
max           5.000000
Name: stars, dtype: float64
```

In [7]:
```
star_counts = reviews['stars'].value_counts()
```

In [8]:
```
# task: create a bar plot of number of reviews for each star rating
# hint: sort_index may be useful to get your x-axis in the most intuitive order
star_counts.sort_index().plot(kind='bar')
```

Out[8]:
```
<AxesSubplot:>
```



Цей графік дає корисну наочність. Ми бачимо, що відгуки з трьома зірками або меншими - у нижній частині, а відгуки із п'ятьма зірками - у верхній частині. Ми можемо розглядати п'ять зірок як найкращі категорії, а три зірки і менше як «недобре».

Візуалізація оглядів

З аналізу розподілу рейтингів зіркових оглядів, нашим першим підходом для надання відповіді маркетинговій команді буде перегляд найпопулярніших слів у найкращих оглядах та порівняння їх із тими, що існують у найгірших оглядах.

Спочатку давайте окремо витягнемо хороші та погані відгуки.

In [9]:
```
# task, filter the 'text' column using the 'stars' column to create Series of good and
# call the results good_review_text and bad_review_text, respectively
# two lines of code here:
good_review_text = reviews[reviews.stars == 5]
bad_review_text = reviews[reviews.stars < 3]
```

Тепер ми можемо перевірити, скільки у нас рядків, і чи схожі вони на кількість відгуків, які ми очікували від розподілу рейтингів зірок.

In [10]:
```
good_review_text.shape
```

Out[10]:
```
(243057, 9)
```

In [11]:
```
bad_review_text.shape
```

Out[11]:
```
(130799, 9)
```

Зараз це насправді багато оглядів. Для наших цілей тут ми почнемо з простого вибору перших 1000 оглядів для кожної групи на основі припущення, що вони упорядковані випадковим чином. Також ми додамо невеликий уточнюючий крок перетворення символів у малі регістри, щоб не рахувати два рази, наприклад, "Bad" та "bad".

In [12]:
```
# task: select the first 1000 items in each of the good and bad review text Series and
# str.lower() method to convert characters to lower case. Save the results back in pla
# two lines of code here:
good_review_text = good_review_text.head(1000).text.str.lower()
bad_review_text = bad_review_text.head(1000).text.str.lower()
```

In [13]:
```
# Check first few good reviews
good_review_text.head()
```

Out[13]:
```
2     you can't really find anything wrong with this...
4     our family loves the food here. quick, friendl...
6     their pettuccine was fresh-made in the morning...
9     this place epitomizes the rumored transformati...
14    this place is quite possibly my favorite resta...
Name: text, dtype: object
```

In [14]:
```
# check first few bad reviews
bad_review_text.head()
```

Out[14]:
```
7     came here on a thursday night at 6:30 p.m. my ...
8     went here last weekend and was pretty disappoi...
11    th service here is very hit or miss... sometim...
12    i took my wife out for a birthday dinner with ...
16    normally, i give a restaurant at least 3 stars...
Name: text, dtype: object
```

Отже, ми розподілили наші огляди на групи, які ми вважаємо «чудовими» та «поганими». Наша проблема зараз полягає у тому, щоб зрозуміти, у чому полягає різниця між ними. Як ми інтерпретуємо чи візуалізуємо інформацію? Чудовим способом візуального перегляду такої оцінки є ознака того, як часто трапляються певні слова або словосполучення. Хороший вступ до створення wordclouds є тут у статті спільноти DataCamp. Очевидно, що необхідно встановити відповідну бібліотеку для того, щоб подальша робота була можливою.

In [15]:
```
# task: import WordCloud and STOPWORDS here
# one line of code here
from wordcloud import WordCloud, STOPWORDS
```

Спочатку нам потрібно зібрати огляди в єдину структуру для кожного хорошого та поганого відгуків для wordcloud.

In [16]:
```
# task: combine all the good and bad review text into a single string for each
# two lines of code here
good_text = ' '.join(good_review_text)
bad_text = ' '.join(bad_review_text)
```

Wordcloud з хороших слів

Тепер, нарешті, ми можемо створити wordcloud! Давайте розглянемо топ-50 слів з найкращих відгуків.

In [17]:
```
# task: generate a wordcloud of good review words, max 50 words
# one line of code here, call the result good_wordcloud
good_wordcloud = WordCloud(max_words=50).generate(good_text)
```

In [18]:
```
plt.imshow(good_wordcloud, interpolation='bilinear')
```

Out[18]:
```
<matplotlib.image.AxesImage at 0x7fde26db01f0>
```

Для маркетологів це зображення має велику цінність. Воно виглядає розумно. Багато хто із маркетингу буде зацікавлений прийняти це зображення таким, яким воно є, і використовувати його.

In [19]:
```
# task: use the to_file method for wordcloud to save the above image to send to market
# one line of code here
good_wordcloud.to_file("out/06_lab6_good_wordcloud.png")
```

Out[19]:
```
<wordcloud.wordcloud.WordCloud at 0x7fde2ba0a2e0>
```

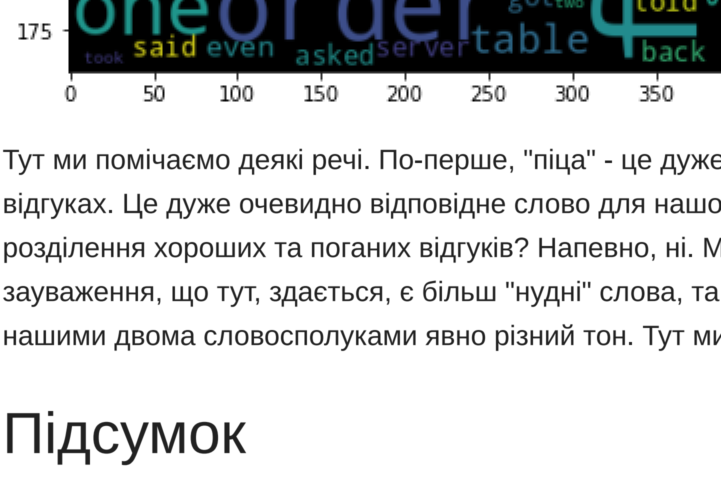
Wordcloud з поганих слів

Спробуємо дізнатися щось додаткове, переглянувши найкращі слова в поганих відгуках.

In [20]:
```
# task: generate a wordcloud of bad review words, max 50 words
# one line of code here, call the result bad_wordcloud
bad_wordcloud = WordCloud(max_words=50).generate(bad_text)
```

In [21]:
```
plt.imshow(bad_wordcloud, interpolation='bilinear')
```

Out[21]:
```
<matplotlib.image.AxesImage at 0x7fde26d4b340>
```



Тут ми помічаємо деякі речі. По-перше, "піца" - це дуже помітне слово, але це було і в найкращих відгуках. Це дуже очевидно відповідне слово для нашої цільової категорії, але чи корисне воно для розділення хороших та поганих відгуків? Напевно, ні. Ми також можемо почати робити цікаві зауваження, що тут, здається, є більш "нудні" слова, такі як "told", "said", "came" та "went". Між нашими двома словосполуками явно різний тон. Тут ми безумовно досягли чогось корисного.

Підсумок

Ми багато чого досягли за час виконання даного завдання. Перш за все ми пов'язали бізнес-проблему з набором даних та визначили, які саме дані нам потрібні. Далі ми ознайомилися з цими даними, щоб відповісти на важливе запитання: яка категорія продуктів харчування має цікавити наш бізнес. Тоді ми використали отримані знання, щоб витягти лише відповідні огляди з великого файлу, які в іншому випадку були б занадто великими для обробки. Ми також розділили окремі найкращі та найгірші відгуки, щоб подивитися на них окремо і чітко помітили, що між ними є різниця.

Отже, ми отримали корисний результат, але ми можемо зробити більше. Подальша обробка уточнює ту маркетингову інформацію, яку ми вже отримали.

Покращення набору слів wordcloud

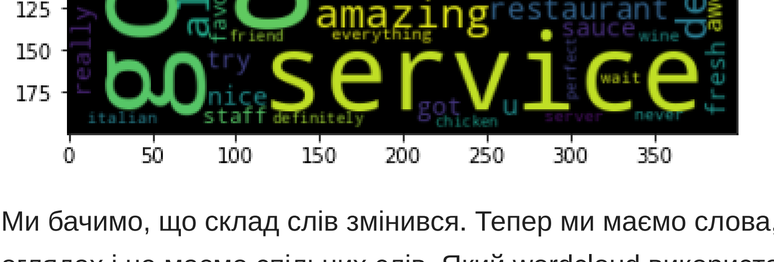
Ми помітили, що деякі слова, які мають часте використання в поганих оглядах, також наявні в хороших оглядах. Потрібно виключити їх із wordcloud, вказавши їх як стоп-слова. Додамо їх до списку стоп-слів за замовчуванням STOPWORDS.

In [22]:
```
bad_stopwords = set(['pizza', 'food', 'order', 'place'])
```

In [23]:
```
# task: create a set of stopwords and add the "bad" ones above to it
# two lines of code here:
stopwords = set(STOPWORDS)
stopwords |= bad_stopwords
```

In [24]:
```
# task: generate a better wordcloud of good review words, max 50 words
# one line of code here, call the result better_wordcloud
better_wordcloud = WordCloud(stopwords=stopwords, max_words=50).generate(good_text)
plt.imshow(better_wordcloud, interpolation='bilinear')
```

Out[24]:
```
<matplotlib.image.AxesImage at 0x7fde1494d940>
```



Ми бачимо, що склад слів змінився. Тепер ми маємо слова, які зустрічаються тільки в позитивних оглядах і не маємо спільних слів. Який wordcloud використовувати - це справа ваших маркетологів.

Післямова

Використання одиничних слів - це хороший початок, але слова, що виникають у послідовностях і самі послідовності можуть мати важливе значення. Розглядаючи хороші і погані огляди та аналізуючи лише окремі слова, ми втратимо, наприклад, словосполучення «highly recommended». Для підвищення точності оцінки необхідно використовувати словосполучення з двох, трьох і більше слів.