Documentación:
Proyecto 2 – Fase 1 --- Análisis del conjunto de datos de Yelp

1. Objetivo

El objetivo de este proyecto es analizar las reseñas de Yelp y obtener información útil sobre las calificaciones por estrellas y datos específicos de cada sector.
Para ello, se combinan los textos no estructurados de las reseñas de review.json con la información empresarial contenida en business.json.

---

2. Desarrollo del proyecto

2.1. Integración de Google Drive en Colab

- **¿Por qué?**
  Dado que los archivos están almacenados en Google Drive, fue necesario configurar el acceso a esta plataforma

  ¿Cómo?
  Mediante el siguiente código en Python:

  from google.colab import drive
  drive.mount ('/content/drive')

- **Resultado:**
  Google Drive se montó exitosamente, y los datos en la carpeta /content/drive/My Drive/Colab Notebooks/Proyecto 2/ fueron accesibles

2.2. Lectura del archivo review.json

- **¿Por qué?**
  El archivo review.json contiene textos de reseñas y calificaciones por estrellas, los cuales son los datos principales para el análisis
- **¿Cómo?**
  Dado que el archivo es muy grande (casi 7 millones de líneas), se leyó solo una parte (1 millón de líneas) para evitar problemas de memoria
- **Código Python:**

```python
import pandas as pd

file_path = "/content/drive/My Drive/Colab Notebooks/Proyecto 2/yelp_academic_dataset_
chunks = pd.read_json(file_path, lines=True, chunksize=1000000)
data = next(chunks)
```

- **Resultado:**

  1 millón de filas de review.json se cargaron exitosamente.

  El archivo contiene las siguientes columnas:

  - review_id: ID única de la reseña
  - user_id: ID del usuario
  - business_id: ID del negocio
  - stars: Calificación por estrellas (1–5)
  - useful, funny, cool: Valoraciones adicionales de otros usuarios
  - text: Texto no estructurado de la reseña
  - date: Fecha de la reseña

2.3. Se cargó el archivo business.json

- **¿Por qué?:** business.json contiene información sobre los negocios que se debe relacionar con las reseñas (por ejemplo, nombre, categoría, ciudad)
- **¿Cómo?**: Código en Python

  business_path = "/content/drive/My Drive/Colab Notebooks/Projekt 2 /yelp_academic_dataset_business.json"
  business_data = pd.read_json(business_path, lines=True)
  print(business_data.head())
  print("Spalten in business.json:", business_data.columns)

- **Resultado**:
  Los datos de la empresa se cargaron con éxito
  El archivo contiene columnas como:

  - **business_id**: ID única de la empresa (para vincular con las reseñas)
  - **name**: Nombre de la empresa
  - **categories**: Categorías o sectores (por ejemplo, restaurantes, cafeterías)
  - **city, state**: Ubicación
  - **stars**: Calificación promedio de estrellas

2.4. Categorías estandarizadas

- **¿Por qué?**
  - En la columna categories existían diferentes formas de escribir y ordenar las mismas categorías
    (por ejemplo, "Restaurante Mexicano" vs. "Mexicano Restaurante")
  - El objetivo era estandarizar estas categorías para poder analizarlas correctamente más adelante
- **¿Cómo?**

  Las categorías fueron procesadas de la siguiente manera:
  - Se convirtieron a minúsculas utilizando str.lower()
  - Se transformaron en listas separadas por comas utilizando str.split(', ')
  - Se ordenaron alfabéticamente para unificar el orden

Código Python

```
merged_data['categories_normalized'] = merged_data['categories'].dropna().str.lower().
    lambda x: ', '.join(sorted(x)) if isinstance(x, list) else x
)
```

Resultado:

- Categorías como "Restaurants, Bars" y "Bars, Restaurants" ahora se guardan como "bars, restaurants"
- Las categorías vacías se omitieron para evitar errores
- **Ejemplo:**
  - Antes: "Restaurants, Bars, Cafés"
  - Después: "bars, cafés, restaurants"

2.5. Número de evaluaciones para categorías específicas

¿Por qué?

- El objetivo era determinar el número de evaluaciones para 12 categorías específicas seleccionadas.
- Esto muestra con qué frecuencia estas categorías están presentes en los datos.

¿Cómo?

- Se creó una lista de categorías (specific_keywords), por ejemplo:
    - japanese, mexican, pizza, steakhouses, etc.
- Usando una combinación de filtros y conteos, se determinó el número de evaluaciones para cada categoría.

Código Python

Resultado:

- **Las categorías con más evaluaciones fueron:**
    - *American (new)* y *American (traditional)* (más de 140.000 evaluaciones)
- **Las categorías con menos evaluaciones fueron:**
    - *Pakistani* (5.252 evaluaciones) y *Korean* (7.766 evaluaciones)

2.6 Representación visual: Cantidad de evaluaciones por categoría

¿Por qué?

- El objetivo era mostrar de forma visual la distribución del número de evaluaciones para las 12 categorías seleccionadas, con el fin de comparar su popularidad.
- La representación permite identificar rápidamente categorías con muchas evaluaciones, como *American (new)*, o categorías con pocas evaluaciones, como *Pakistani*.

¿Cómo?

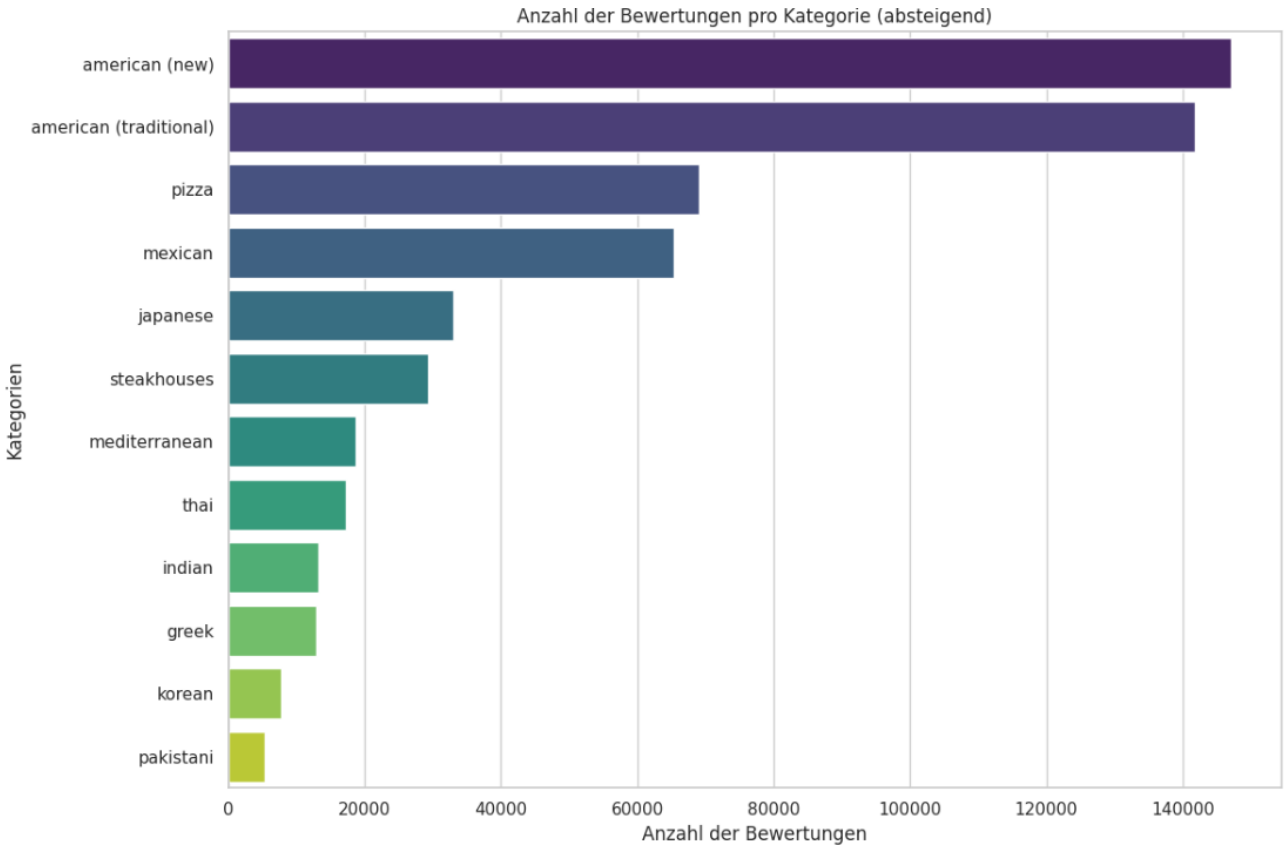
- El número de evaluaciones se ordenó y se representó en un gráfico de barras descendente.

Código Python

```python
import matplotlib.pyplot as plt

# Daten sortieren
sorted_counts = sorted(keyword_counts.items(), key=lambda x: x[1], reverse=True)
categories = [item[0] for item in sorted_counts]
counts = [item[1] for item in sorted_counts]

# Balkendiagramm
plt.figure(figsize=(10, 6))
plt.barh(categories, counts, color='skyblue')
plt.xlabel('Anzahl der Bewertungen')
plt.ylabel('Kategorien')
plt.title('Anzahl der Bewertungen pro Kategorie (absteigend)')
plt.gca().invert_yaxis()
plt.show()
```

Resultado:

- Las categorías con más valoraciones, como "american (new)" y "american (traditional)", se encuentran en la parte superior.
- Las categorías con menos valoraciones, como "pakistani", se encuentran en la parte inferior.



Conclusiones del diagrama:

- Categorías como "american (new)" y "american (traditional)" tienen, con diferencia, la mayor cantidad de valoraciones, lo que indica que estas categorías son las más frecuentemente visitadas o evaluadas por los clientes.
- Categorías como "pakistani" y "korean" tienen significativamente menos valoraciones, lo que podría indicar una menor popularidad o una oferta más especializada

Interpretación:

- Esta distribución podría reflejar diferencias regionales, una mayor cantidad de restaurantes estadounidenses o preferencias específicas de los clientes
- Los resultados podrían ser útiles para estrategias de marketing dirigidas o análisis adicionales, como investigar por qué algunas categorías reciben menos valoraciones

2.7. Análisis de calificación por estrellas según categorías

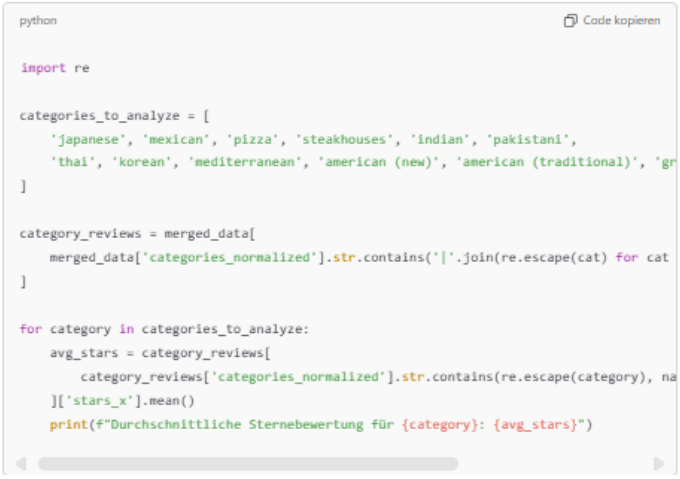
¿Por qué?

- El objetivo fue analizar cómo varían las calificaciones por estrellas entre diferentes categorías (por ejemplo, "mexicana", "japonesa", "pizza")
- Esto permite identificar categorías con alta o baja satisfacción del cliente

¿Cómo?

- Se filtraron las calificaciones para 12 categorías específicas:
    - *japonesa, mexicana, pizza, steakhouses, india, pakistaní, tailandesa, coreana, mediterránea, americana (nueva), americana (tradicional), griega*
- Se utilizaron métodos para manejar caracteres especiales en las categorías (e.g. *americana …*) mediante re.escape, evitando así advertencias o errores
- Se calculó la calificación promedio de estrellas para cada categoría

Código Python



Resultado:

- Se calcularon las calificaciones promedio por estrellas para las 12 categorías
- Resultado de Ejemplo:
    - japonés: 3.83
    - mexicano: 3.72
    - pizza: 3.68
    - griego: 3.99
    - americano (tradicional): 3.80

2.8. Representación visual: Calificación promedio por estrellas

¿Por qué?

- El objetivo era comparar qué categorías tienden a recibir mejores calificaciones (por ejemplo, "griego") y cuáles reciben peores (por ejemplo, "pizza").
- Este análisis muestra cómo varía la satisfacción del cliente entre categorías.

¿Cómo?

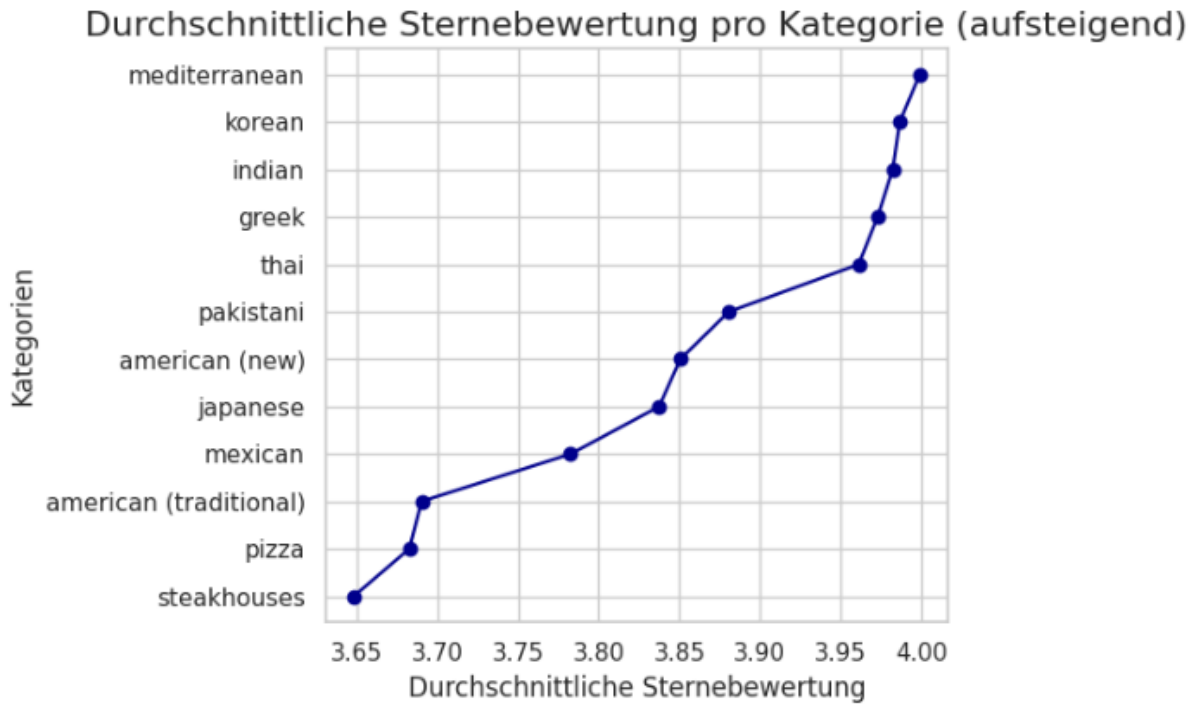
- Se calcularon las calificaciones promedio por estrellas para cada categoría y se representaron en un gráfico de barras.
- Las categorías fueron ordenadas de forma descendente según su calificación promedio.

Código Python

```python
import matplotlib.pyplot as plt
import re

# Durchschnittliche Sternebewertungen berechnen
average_stars = {}
for category in categories_to_analyze:
    avg_stars = category_reviews[
        category_reviews['categories_normalized'].str.contains(re.escape(category))
    ]['stars_x'].mean()
    average_stars[category] = avg_stars

# Daten sortieren
sorted_stars = sorted(average_stars.items(), key=lambda x: x[1], reverse=True)
categories = [item[0] for item in sorted_stars]
stars = [item[1] for item in sorted_stars]

# Balkendiagramm
plt.figure(figsize=(10, 6))
plt.barh(categories, stars, color='lightgreen')
plt.xlabel('Durchschnittliche Sternebewertung')
plt.ylabel('Kategorien')
plt.title('Durchschnittliche Sternebewertung pro Kategorie (absteigend)')
plt.gca().invert_yaxis()
plt.show()
```

Resultado:

- Categorías como "griega" o "mediterránea" tienen las calificaciones promedio más altas (cerca de 4 estrellas).
- Categorías como "pizza" o "asadores" están en el extremo inferior con calificaciones cercanas a 3,6



Hallazgos del diagrama:

- Categorías como "mediterranean" y "greek" obtienen las calificaciones promedio más altas (cerca de 4 estrellas), destacándose como las mejores
- Categorías como "pizza" y "steakhouses" tienen las calificaciones más bajas (aproximadamente entre 3,65 y 3,7 estrellas)
- La diferencia entre la calificación más alta y la más baja es relativamente pequeña, lo que indica que, en general, los clientes están satisfechos con los servicios

Interpretación:

- Calificaciones más altas en categorías como "mediterránea" podrían indicar una alta satisfacción del cliente y buena calidad
- Calificaciones más bajas en "pizza" o "steakhouses" podrían señalar posibles problemas, como el sabor, el servicio o expectativas que no se cumplieron
- Esto podría ser motivo para realizar investigaciones adicionales, por ejemplo, analizar los comentarios negativos más frecuentes en las reseñas

2.9. Análisis de la distribución de calificaciones por estrellas

¿Por qué?

- El objetivo era analizar la distribución de las calificaciones por estrellas (1 a 5 estrellas) en las diferentes categorías para obtener información sobre la satisfacción de los clientes

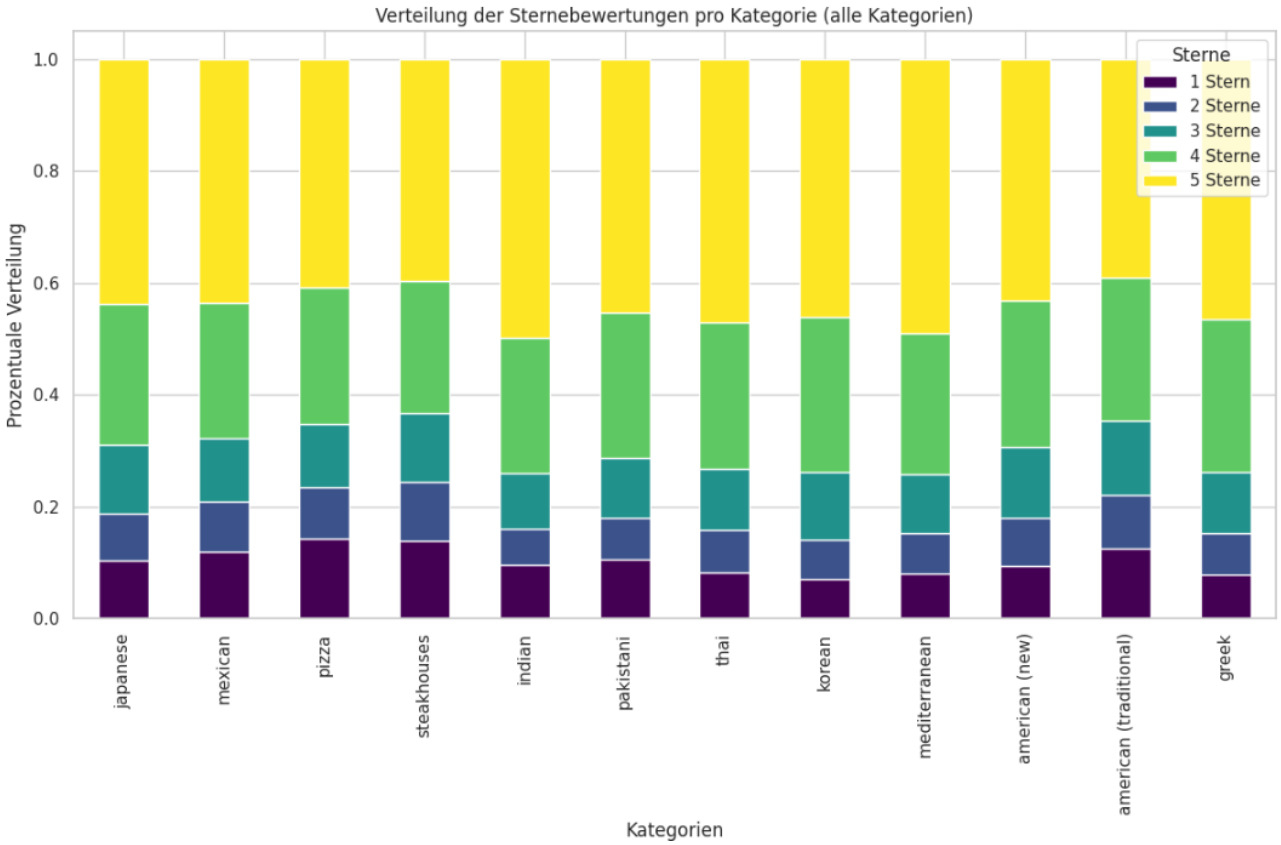
¿Cómo?

- Se contaron las calificaciones por estrellas para cada categoría y se normalizaron como una distribución porcentual
- Se utilizó un gráfico de barras apiladas para visualizar cómo se distribuyen las calificaciones por estrellas en las distintas categorías

Código Python

python Code kopieren

```python
# Verteilung der Sternebewertungen pro Kategorie
ratings_distribution = {}
for category in categories_to_analyze:
    category_data = merged_data[merged_data['categories_normalized'].str.contains(
    if not category_data.empty:
        star_counts = category_data['stars_x'].value_counts(normalize=True).sort_i
        ratings_distribution[category] = star_counts

# In DataFrame umwandeln
distribution_df = pd.DataFrame(ratings_distribution).fillna(0).T
distribution_df.columns = ['1 Stern', '2 Sterne', '3 Sterne', '4 Sterne', '5 Stern

# Gestapeltes Balkendiagramm
distribution_df.plot(kind='bar', stacked=True, figsize=(12, 8), colormap='viridis'
plt.title('Verteilung der Sternebewertungen pro Kategorie (alle Kategorien)')
plt.xlabel('Kategorien')
plt.ylabel('Prozentuale Verteilung')
plt.legend(title='Sterne')
plt.tight_layout()
plt.show()
```



Resultado:

- "Indian", "mediterranean" y "greek" tienen una proporción particularmente alta de valoraciones de 5 estrellas
- "Pizza" y "steakhouses" tienen una mayor proporción de valoraciones de 1 y 2 estrellas, lo que sugiere potencial de mejora
- En general, la proporción de valoraciones de 4 y 5 estrellas en casi todas las categorías es muy alta, lo que indica una satisfacción general de los clientes

2.10. Nubes de palabras: Palabras más frecuentes en las reseñas

¿Por qué?
Para obtener una visión rápida de los términos más comunes en las reseñas positivas y negativas. Esto ayuda a representar visualmente los aspectos clave de satisfacción e insatisfacción de los clientes

¿Cómo?

1. **Extracción de palabras principales de las reseñas:**
    - Se analizaron por separado las reseñas positivas (4 y 5 estrellas) y las negativas (1 y 2 estrellas)
    - Se utilizó una herramienta de *CountVectorizer* para contar las palabras más frecuentes
2. **Ajuste de la lista de palabras vacías:**
    - Palabras comunes pero irrelevantes para el análisis (por ejemplo, "solo", "muy") se eliminaron mediante una lista de palabras vacías personalizada
3. **Creación de nubes de palabras:**
    - Las 5 palabras principales de las reseñas positivas y negativas se representaron en nubes de palabras separadas
    - El tamaño de las palabras refleja su frecuencia

Código Python

```python
# Zusätzliche Stopwörter definieren
custom_negative_stop_words = ['did', 'ordered', 'came', 'just', 'good', 'like', 'said', 'd
custom_positive_stop_words = ['just', 'really', 've']

# Funktion zur Wortanalyse mit spezifischen Stopworten
def get_top_words_refined(texts, n=10, extra_stop_words=None):
    from sklearn.feature_extraction.text import ENGLISH_STOP_WORDS
    all_stop_words = list(ENGLISH_STOP_WORDS) + (extra_stop_words if extra_stop_words else
    vectorizer = CountVectorizer(stop_words=all_stop_words, max_features=1000)
    word_matrix = vectorizer.fit_transform(texts)
    word_counts = np.asarray(word_matrix.sum(axis=0)).flatten()
    word_indices = np.argsort(word_counts)[::-1][:n]
    top_words = [(vectorizer.get_feature_names_out()[i], word_counts[i]) for i in word_ind
    return top_words

# Häufigste Wörter berechnen
positive_words = get_top_words_refined(positive_reviews, n=5, extra_stop_words=custom_posi
negative_words = get_top_words_refined(negative_reviews, n=5, extra_stop_words=custom_nega

# Wortwolken erstellen
generate_wordcloud([word for word, _ in positive_words], [count for _, count in positive_w
generate_wordcloud([word for word, _ in negative_words], [count for _, count in negative_w
```

Resultado:

Reseñas positivas

- Palabras como "great", "food", "delicious", "service" y "place" dominan las reseñas positivas
- Estas palabras indican la gran importancia de la calidad de la comida y el servicio

Reseñas negativas

- Palabras como "food", "service", "place", "time" y "order" aparecen con frecuencia en las reseñas negativas
- Estos términos muestran posibles áreas problemáticas, como largos tiempos de espera y malas experiencias de servicio

![Nube de palabras positiva]
Una nube de palabras verde que muestra términos positivos como "great", "food" y "delicious"

![Nube de palabras negativa]
Una nube de palabras roja que destaca términos negativos como "food", "service" y "order"