

Análisis de Reseñas de Lugares

Moises León

3 de julio de 2023



En esta presentación, se explicará el código implementado en Python para el análisis de reseñas de lugares. El código realiza las siguientes tareas:

- Carga y procesa datos desde archivos CSV.
- Aplica Procesamiento de Lenguaje Natural (PLN) utilizando spaCy.
- Asigna colores a las reseñas según su calificación.
- Crea un mapa interactivo utilizando Folium para visualizar los lugares y sus calificaciones.

Funciones Importantes

```
import pandas as pd
import spacy
import folium
```

```
# Función para preprocesar el texto de las reseñas
```

```
def preprocess_text(text, nlp):
    # Implementación del PLN con spaCy
    ...
    return ' '.join(lemmas)
```

```
# Función para asignar color a las reseñas según su calificación
```

```
def asignar_color_calificacion(calificacion):
    # Asignación de colores según rangos de calificación
    ...
    return color
```

Carga y Procesamiento de Datos

```
# Leer y procesar los archivos CSV
detalles_df = pd.read_csv('detalles_lugares.csv')
reseñas_df = pd.read_csv('reseñas_lugares.csv')

# Identificar lugares evaluados sin duplicados
...

# Combinar información de detalles y reseñas
...

# Llenar valores faltantes
...

# Procesamiento de Lenguaje Natural
...
```

Creación del Mapa Interactivo

```
# Crear un mapa con Folium
```

```
mapa = folium.Map(  
    location=[lugares_evaluados_df['latitud'].mean(), lugares.  
    zoom_start=12  
)
```

```
# Agregar marcadores para los lugares en el mapa
```

```
for index, lugar in lugares_evaluados_df.iterrows():  
    if not pd.isna(lugar['latitud']) and not pd.isna(lugar['lon  
        # Crear el texto de la ventana emergente (popup) del m  
        popup_text = f"<b>{lugar['nombre']}</b><br>Calificació
```

```
# Agregar el marcador al mapa con el color asignado se
```

```
folium.Marker(  
    location=[lugar['latitud'], lugar['longitud']],  
    popup=popup_text,  
    tooltip=lugar['nombre']
```

Resultados y Conclusiones

```
# Guardar el mapa interactivo en un archivo HTML  
mapa.save('mapa_interactivo_lugares_evaluados.html')
```

```
# Guardar los datos de los lugares evaluados en un archivo CSV  
lugares_evaluados_df.to_csv('lugares_evaluados.csv', index=False)
```

El código ha creado un mapa interactivo que muestra los lugares evaluados y sus calificaciones, con marcadores de colores diferentes para cada calificación. También se ha guardado la información detallada de los lugares evaluados en un archivo CSV.

Mapa interactivo

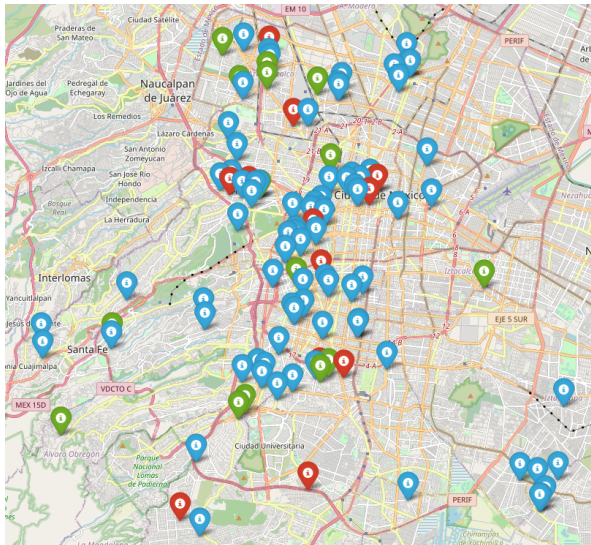


Figura: Mapa interactivo

Mapa interactivo

