Map visualisation

July 21, 2023

Map visualisation

Prepare data of sentiments as a GeoJSON FeatureCollection for using on a OpenStreetMap with the vanilla-js-web-component-leaflet-geojson to look like this.

Setup

[ ]:
```
!pip install pandas
```

Loading the data

The data to be loaded are related to the results of 'sentimientos al paso' and are the following: - Associated sentiment using OpenAI - Associated sentiments using the Python library pysentimiento - Associated sentiments using the Python library twitter-XLM-roBERTa-base for Emotion Analysis - Location data

[2]:
```
import pandas as pd

openai_and_pysentimientos_url = 'https://github.com/migupl/sentimientos-al-paso/
 ↪raw/main/notebooks/output/versosalpaso_robertuito-sentiment-analysis.csv'
openai_and_pysentimientos = pd.read_csv(openai_and_pysentimientos_url, sep=';',␣
 ↪encoding='utf-8')
openai_and_pysentimientos.columns
```

[2]:
```
Index(['Unnamed: 0.1', 'Unnamed: 0', 'id', 'latitud', 'longitud', 'autor',
       'barrio', 'verso', 'direccion', 'openai_sentiment', 'quarter',
       'district', 'city', 'robertuito_sentiment',
       'robertuito_sentiment_probas'],
      dtype='object')
```

[3]:
```
sentiments_with_location = openai_and_pysentimientos[
    ['verso', 'autor', 'latitud', 'longitud', 'district', 'openai_sentiment',␣
 ↪'robertuito_sentiment']
]
```

[4]:
```
no_of_districts = len(sentiments_with_location.district.unique())
print(f'There are {no_of_districts} districts')
```

```
There are 21 districts
```

[5]:
```
twitter_xlm_url = 'https://github.com/migupl/sentimientos-al-paso/raw/main/
 ↪notebooks/output/versosalpaso_twitter-XLM-roBERTa-base.csv'
twitter_xlm = pd.read_csv(twitter_xlm_url, sep=';', encoding='utf-8')
twitter_xlm.columns
```

[5]:
```
Index(['Unnamed: 0.1', 'Unnamed: 0', 'id', 'latitud', 'longitud', 'autor',
       'barrio', 'verso', 'direccion', 'openai_sentiment', 'quarter',
       'district', 'city', 'twitter-xml_sentiment', 'twitter-xml_anger',
       'twitter-xml_disgust', 'twitter-xml_fear', 'twitter-xml_joy',
       'twitter-xml_sadness', 'twitter-xml_surprise', 'twitter-xml_others',
       'twitter-xml_as_positive', 'twitter-xml_as_neutral',
       'twitter-xml_as_negative'],
      dtype='object')
```

Both of the dataframes have the same origin

[ ]:
```
sentiments_with_location['twitter-xml_sentiment'] =␣
 ↪twitter_xlm['twitter-xml_sentiment']
```

Define GeoJSON Feature

[7]:
```
import json

def get_heart_colors(sentiments: list[str]) -> str:
    unique_sentiments = sorted(set(sentiments))

    colors_by_sentiment = {'negative': 'red', 'neutral': 'yellow', 'positive':␣
 ↪'green'}
    colors = []
    for sentiment in unique_sentiments:
        colors.append(colors_by_sentiment[sentiment])

    colors_str = '-'.join(colors)
    return colors_str

def geojson_feature(latitude: float, longitude: float, author: str, verse: str,␣
 ↪sentiments: list[str]) -> json:
    color_labels = get_heart_colors(sentiments)
    return {
        'type': "Feature",
        'geometry': {
            'type': "Point",
            'coordinates': [longitude, latitude]
        },
        'properties': {
            'popupContent': f'<strong>{verse}</strong><br>- <cite>{author}</
 ↪cite>',
            'icon': {
```

```
                "iconUrl": f'https://raw.githubusercontent.com/migupl/
↪svg-vectors-and-icons/main/heart-like/heart-{color_labels}.png',
                "iconSize": [41, 41],
                "iconAnchor": [20, 41],
                "popupAnchor": [1, -34]
            },
        }
    }
```

Prepare the GeoJSON's FeatureCollection by district

[8]:
```
sentiments_by_district = sentiments_with_location.groupby('district')
```

[9]:
```
geojson = {}
for name, group in sentiments_by_district:
    features = []
    for index, row in group.iterrows():
        sentiments = [row.openai_sentiment, row.robertuito_sentiment,␣
↪row['twitter-xml_sentiment']]
        feature = geojson_feature(row.latitud, row.longitud, row.autor, row.
↪verso, sentiments)
        features.append(feature)

    geojson[name] = {
        'type': "FeatureCollection",
        'features': features
    }

keys = geojson.keys()
keys
```

[9]:
```
dict_keys(['Arganzuela', 'Barajas', 'Carabanchel', 'Centro', 'Chamartín',
'Chamberí', 'Ciudad Lineal', 'Fuencarral-El Pardo', 'Hortaleza', 'Latina',
'Moncloa-Aravaca', 'Moratalaz', 'Puente de Vallecas', 'Retiro', 'Salamanca',
'San Blas - Canillejas', 'Tetuán', 'Usera', 'Vicálvaro', 'Villa de Vallecas',
'Villaverde'])
```

[10]:
```
no_of_keys = len(keys)
assert 21, no_of_keys
```

Save JSON for later use

[11]:
```
js_content = f"""
const data = {geojson};
export {{ data }}
"""
```

[12]:
```
js_file_path = './output/sentiments_by_district_geo.js'
with open(js_file_path, 'w') as f:
    f.write(js_content)
```