

Universidad Nacional Agraria La Molina



CURSO: Lenguaje de Programación 2

TÍTULO: Análisis de Publicidades de Videojuegos en Reddit

INTEGRANTES:

- Gervasi Rodriguez, Mateo
- Peraltilla Parraguez, Angel
- Morales Carlos, Mauricio

Introducción al uso de la API de Reddit para Extracción de Datos

La API de Reddit, integrada a través de la biblioteca PRAW en Python, proporciona acceso programático a los contenidos y datos disponibles en Reddit. Reddit es una plataforma social muy popular que alberga una gran variedad de comunidades (llamadas subreddits) dedicadas a diversos temas, incluidos los videojuegos.

Propósito y Funcionalidad

El propósito principal de utilizar la API de Reddit en conjunción con PRAW es la extracción automatizada de datos específicos de Reddit, como publicaciones, comentarios y detalles asociados, basados en criterios de búsqueda predefinidos. En el contexto de análisis de videojuegos y publicidades:

- **Extracción de Publicaciones:** Permite obtener publicaciones específicas relacionadas con términos clave como "advertisement", "ad", "marketing" o "promo" en subreddits relevantes como r/gaming.
- **Recopilación de Datos:** Facilita la recolección de información como títulos de publicaciones, autores, puntajes (score), número de comentarios, URLs y fechas de creación.

Configuración y Uso

Para utilizar la API de Reddit con PRAW, es necesario configurarla con credenciales específicas de la aplicación (cliente ID y cliente secreto), junto con un agente de usuario único que identifica tu aplicación para Reddit.

Ejemplo Práctico

A continuación se muestra un ejemplo de cómo configurar y utilizar la API de Reddit para extraer datos de publicaciones relacionadas con publicidades en videojuegos desde el subreddit r/gaming:

```
import praw
import csv
import datetime

# Configuración de La API de Reddit
reddit = praw.Reddit(
    client_id='46K3g9wdgnG6N-EULyNyCA',
    client_secret='Wu0ex80UAuqV9fv6xkN2wkCIZ-NR6Q',
    user_agent='Traditional-Can3255'
)

# Define el subreddit y el término de búsqueda
subreddit_name = 'gaming' # Puedes cambiar a otros subreddits relacionados con videojuegos
search_term = 'advertisement OR ad OR marketing OR promo' # Términos de búsqueda para encontrar publicidades

# Extrae publicaciones relacionadas con videojuegos
subreddit = reddit.subreddit(subreddit_name)
posts = subreddit.search(search_term, sort='relevant', time_filter='all', limit=100) # Cambia el límite según tus necesidades

# Archivo CSV para guardar Los datos extraídos
csv_file = 'video_game_advertisements.csv'

# Abrir el archivo CSV para escribir datos
with open(csv_file, mode='w', newline='', encoding='utf-8') as file:
    writer = csv.writer(file)
    writer.writerow(['Title', 'Author', 'Score', 'Comments', 'URL', 'Date']) # Cabeceras de columnas

for post in posts:
    # Extraer datos de cada publicación
    title = post.title
    author = post.author.name if post.author else 'Unknown'
    score = post.score
    comments = post.num_comments
    url = post.url
    date = datetime.datetime.fromtimestamp(post.created_utc).strftime('%Y-%m-%d %H:%M:%S')

    # Escribir los datos en el archivo CSV
    writer.writerow([title, author, score, comments, url, date])

print(f'Datos extraídos y guardados en {csv_file}')
```

Datos extraídos y guardados en video_game_advertisements.csv

Delimiter: , v						
	Title	Author	Score	Comments	URL	Date
1	iece of advertising history.	InvadingBacon	1844	35	edd.it/1dqjd4jkn6qc1.jpeg	2024-03-23 20:14:08
2	I this advertisement today	Lxlith_s	14756	1214	reddit.com/gallery/u72rek	2022-04-19 06:19:51
3	hent from the Netherlands	DevilUnknown17	9350	637	//i.redd.it/cjerjwasljz81.jpg	2022-05-14 20:29:11
4	ox • 2004 • Advertisement	TurboTBag	5226	440	i.redd.it/9qo9rdzs7jja1.jpg	2023-02-21 02:02:26
5	o-game Advertisement...	Cooper3b	25929	625	i.imgur.com/R7gHPYp.jpg	2017-05-30 00:34:51
6	/ideogame Advertisement	Velock0009	9094	371	p://m.imgur.com/9Aiz4U2	2016-03-04 09:50:20
7	advertisement from 1990.	jdmorgenstern	60238	2947	edd.it/0hav3h0fgmz81.jpg	2022-05-15 06:04:27
8	This ad is just... beautiful	csiramokus	48389	536	i.imgur.com/pDqoLVQ.jpg	2017-04-15 04:26:12
9	g PSVITA advertisement.	jabnes	2604	110	add.it/gax387wcu6v81.jpg	2022-04-22 21:17:32
10	or Gameboy Advance SP	newtonic	9813	296	'edd.it/kep39rvay2a81.jpg	2022-01-06 09:45:42
11	a mobile gardening game	Ghulam_Jewel	22183	758	aglarngindochinahogdeer	2019-08-08 16:55:02
12	Boy Advance Walmart ad	sworedmagic	21803	660	add.it/rso02c11p9sc1.jpeg	2024-04-03 08:35:55
13	Great advertising	lostti311	21673	458	redd.it/q0lrmxjw5xc61.jpg	2021-01-22 12:32:22
14	2002 Sims 2 ad	sworedmagic	22076	583	i.redd.it/blq5est5cerb1.jpg	2023-09-30 08:39:45
15	3D • [MS-DOS AD • 1996]	TurboTBag	3223	503	redd.it/rvmnz4i10rx191.jpg	2022-05-26 22:11:38
16	vertisement ever for WoW	silentez	15647	466	add.it/55uosm77rug01.jpg	2018-02-17 17:50:38
17	ting in Sydney (Australia)	evilthing	14319	310	redd.it/tvxp6qao9vb1.jpg	2023-10-19 21:15:28
18	rtisements in a \$60 game	AmazinglyMagicToast	39241	4398	//i.imgur.com/wspBuQt.jpg	2016-06-03 07:33:43
19	o advertise memory cards	martinfisherman	42782	2088	//i.redd.it/s7uasiqp54y.jpg	2016-12-20 11:57:15

```

import praw
import csv
import datetime
import pandas as pd
# Leer datos desde el archivo CSV
data = pd.read_csv('video_game_advertisements.csv')

# Convertir la columna 'Date' a tipo datetime y extraer el año
data['Date'] = pd.to_datetime(data['Date'])
data['Year'] = data['Date'].dt.year

# Calcular el promedio de puntaje y comentarios por año
yearly_stats = data.groupby('Year').agg({'Score': 'mean', 'Comments': 'mean'}).reset_index()
yearly_stats.rename(columns={'Score': 'Average_Score', 'Comments': 'Average_Comments'}, inplace=True)

# Archivo CSV para guardar los promedios por año
yearly_csv_file = 'yearly_video_game_ad_stats.csv'

# Guardar los datos en un nuevo archivo CSV
yearly_stats.to_csv(yearly_csv_file, index=False, encoding='utf-8')

print(f'Datos de promedios por año guardados en {yearly_csv_file}')

```

Datos de promedios por año guardados en yearly_video_game_ad_stats.csv

Delimiter:

	Year	Average_Score	Average_Comments
1	2009	558.0	357.0
2	2011	4233.0	1787.0
3	2012	1590.0	357.0
4	2013	1915.5	327.5
5	2014	8914.538461538461	814.0
6	2015	13623.57142857143	993.2857142857143
7	2016	20836.125	1577.125
8	2017	17504.5	503.5
9	2018	31590.14285714286	676.2857142857143
10	2019	48276.666666666664	1192.6666666666667
11	2020	29540.75	588.0
12	2021	28951.0	653.6
13	2022	19040.866666666665	759.1333333333333
14	2023	10679.3	764.3
15	2024	10899.5	685.8

```

import pandas as pd
import matplotlib.pyplot as plt
import matplotlib
import os

# Leer datos desde el archivo CSV de promedios por año
yearly_stats = pd.read_csv('yearly_video_game_ad_stats.csv')

# Gráfico de distribución de puntajes
plt.figure(figsize=(10, 5))
plt.hist(data['Score'], bins=30, edgecolor='black')
plt.xlabel('Score')
plt.ylabel('Number of Posts')
plt.title('Distribución de Puntajes de Publicaciones de Videojuegos')
plt.tight_layout()
plt.savefig('score_distribution.png') # Guardar el gráfico como una imagen
plt.close()

```

```

# Crear gráfico de líneas para promedios por año
plt.figure(figsize=(10, 6))
plt.plot(yearly_stats['Year'], yearly_stats['Average_Score'], marker='o', linestyle='-', color='blue', label='Promedio de Puntajes')
plt.plot(yearly_stats['Year'], yearly_stats['Average_Comments'], marker='o', linestyle='-', color='green', label='Promedio de Comentarios')
plt.xlabel('Año')
plt.ylabel('Promedio')
plt.title('Promedio de Puntajes y Comentarios por Año')
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.savefig('yearly_stats.png') # Guardar el gráfico como una imagen
plt.close()

```

```

# Generar contenido HTML
html_content = f"""
<!DOCTYPE html>
<html lang="es">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Análisis de Publicidades de Videojuegos en Reddit</title>
    <style>
        body {{
            font-family: Arial, sans-serif;
            background-color: #f4f4f4;
            color: #333;
            margin: 0;
            padding: 20px;
        }}
        h1 {{
            color: #4CAF50;
        }}
        .container {{
            max-width: 900px;
            margin: 0 auto;
            background-color: #fff;
            padding: 20px;
            border-radius: 8px;
            box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
        }}
        table {{

```

```

    table {{
        width: 100%;
        border-collapse: collapse;
        margin-bottom: 20px;
    }}
    th, td {{
        padding: 10px;
        text-align: left;
        border-bottom: 1px solid #ddd;
    }}
    th {{
        background-color: #4CAF50;
        color: #fff;
    }}
    .chart {{
        text-align: center;
    }}
    }}
</style>
</head>
<body>
    <div class="container">
        <h1>Análisis de Publicidades de Videojuegos en Reddit</h1>
        <p>Este informe presenta un análisis de las publicidades de videojuegos en Reddit, basado en las publicaciones del subreddit <strong>{subreddit_n}</strong>.</p>

        <h2>Estadísticas Básicas</h2>
        <table>
            <tr>

```

```

                <tr>
                    <th>Estadística</th>
                    <th>Valor</th>
                </tr>
                <tr>
                    <td>Promedio de Puntajes</td>
                    <td>{stats['Score']['mean']:.2f}</td>
                </tr>
                <tr>
                    <td>Desviación Estándar de Puntajes</td>
                    <td>{stats['Score']['std']:.2f}</td>
                </tr>
                <tr>
                    <td>Máximo Puntaje</td>
                    <td>{stats['Score']['max']}</td>
                </tr>
                <tr>
                    <td>Mínimo Puntaje</td>
                    <td>{stats['Score']['min']}</td>
                </tr>
                <tr>
                    <td>Promedio de Comentarios</td>
                    <td>{stats['Comments']['mean']:.2f}</td>
                </tr>
                <tr>
                    <td>Desviación Estándar de Comentarios</td>
                    <td>{stats['Comments']['std']:.2f}</td>
                </tr>
            </table>

```

```
<h2>Distribución de Puntajes</h2>
<div class="chart">
    
</div>

<h2>Promedio de Puntajes y Comentarios por Año</h2>
<div class="chart">
    
</div>
</div>
</body>
</html>
"""

# Guardar el contenido HTML en un archivo
with open('analysis_report.html', 'w', encoding='utf-8') as file:
    file.write(html_content)

print('El archivo HTML del análisis ha sido creado como "analysis_report.html".')
```

Análisis de Publicidades de Videojuegos en Reddit

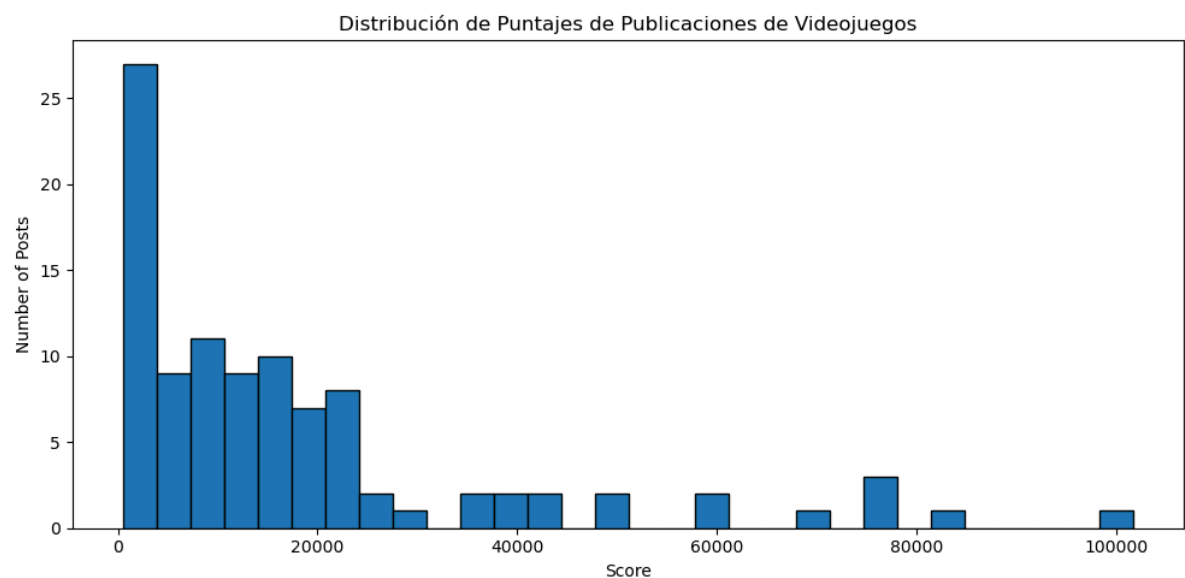
Este informe presenta un análisis de las publicidades de videojuegos en Reddit, basado en las publicaciones del subreddit **gaming** con términos de búsqueda relacionados con anuncios.

Estadísticas Básicas

Estadística	Valor
Promedio de Puntajes	16543.29
Desviación Estándar de Puntajes	13520.83
Máximo Puntaje	48275.0
Mínimo Puntaje	555.0
Promedio de Comentarios	802.41
Desviación Estándar de Comentarios	428.54

... / ...

Distribución de Puntajes



Promedio de Puntajes y Comentarios por Año

