

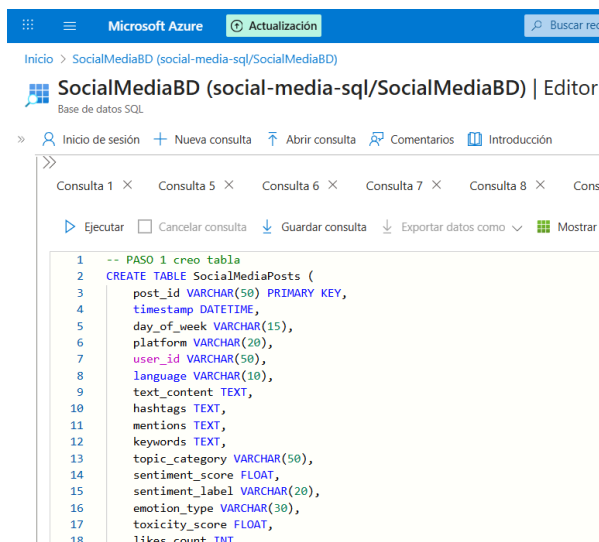
Creacion del Dataset en Azure SQL Cloud

Base de Datos y Tablas en Azure SQL

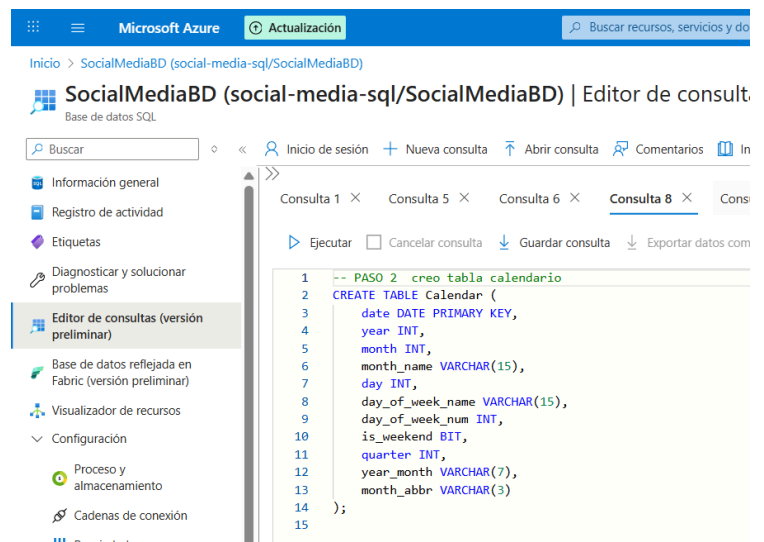
Se implementó una base de datos en Azure SQL Database bajo el nombre **SocialMediaDB**, como parte de una arquitectura moderna de almacenamiento en la nube.

Dentro de esta base se crearon:

- La tabla principal **SocialMediaPosts**, compuesta por 43 columnas, que consolidan datos relevantes de publicaciones en redes sociales, incluyendo métricas de interacción, emociones, sentimiento, hashtags, menciones y datos temporales.
- La tabla auxiliar **Calendar**, diseñada para facilitar análisis temporales y permitir relaciones sólidas entre fechas.



```
1 -- PASO 1 creo tabla
2 CREATE TABLE SocialMediaPosts (
3     post_id VARCHAR(50) PRIMARY KEY,
4     timestamp DATETIME,
5     day_of_week VARCHAR(15),
6     platform VARCHAR(20),
7     user_id VARCHAR(50),
8     language VARCHAR(10),
9     text_content TEXT,
10    hashtags TEXT,
11    mentions TEXT,
12    keywords TEXT,
13    topic_category VARCHAR(50),
14    sentiment_score FLOAT,
15    sentiment_label VARCHAR(20),
16    emotion_type VARCHAR(30),
17    toxicity_score FLOAT,
18    likes count INT,
```



```
1 -- PASO 2 creo tabla calendario
2 CREATE TABLE Calendar (
3     date DATE PRIMARY KEY,
4     year INT,
5     month INT,
6     month_name VARCHAR(15),
7     day INT,
8     day_of_week_name VARCHAR(15),
9     day_of_week_num INT,
10    is_weekend BIT,
11    quarter INT,
12    year_month VARCHAR(7),
13    month_abbr VARCHAR(3)
14 );
15
```

Carga de datos desde Python

La importación de datos se realizó utilizando Visual Studio Code con Python y las librerías `pandas` y `pyodbc`.

Detalles técnicos del proceso:

- Se estableció una conexión segura con Azure SQL Database.
- Se leyó un archivo `.csv` desde el escritorio con codificación UTF-8, para evitar errores de caracteres especiales.
- Las columnas booleanas fueron transformadas a enteros (0 y 1) ya que SQL Server no admite directamente valores booleanos.
- Los valores `NaN` se convirtieron en `None` para ser insertados como `NULL`.
- Se recorrieron las filas del DataFrame y se insertaron en la tabla `SocialMediaPosts` utilizando 43 marcadores de parámetro (`?`), uno por cada columna.
- Se confirmó la transacción y se cerró la conexión.

```
Generate + Code + Markdown | Run All Restart Clear All Outputs View data Jupyter

import pandas as pd
import pyodbc

# Conexión a Azure SQL Database
conn = pyodbc.connect(
    'DRIVER={ODBC Driver 17 for SQL Server};'
    'SERVER=social-media-sql.database.windows.net;'
    'DATABASE=SocialMediaBD;'
    'UID=ROJO;'
    'PWD=aaafgh'
)
cursor = conn.cursor()

# Cargar el CSV local
df = pd.read_csv('C:/Users/Bruger/Desktop/Proyecto Redes/Dataset REDES/final_social_media_posts.csv', encoding='utf-8')

# Convertir booleanos a enteros
df['is_weekend'] = df['is_weekend'].astype(int)
df['has_hashtags'] = df['has_hashtags'].astype(int)
df['has_mentions'] = df['has_mentions'].astype(int)

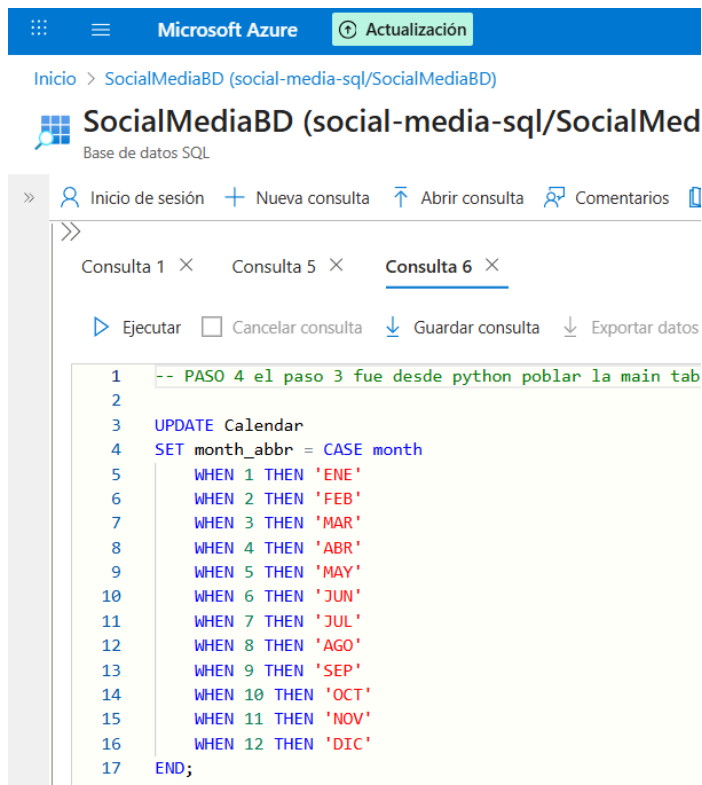
# Reemplazar NaN por None (para que SQL los acepte)
df = df.where(pd.notnull(df), None)

# Insertar datos fila por fila
for index, row in df.iterrows():
    try:
        cursor.execute("""
            INSERT INTO SocialMediaPosts VALUES (
                ?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,
                ?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,
            )
        """, tuple(row))
    except Exception as e:
        print(f"⚠ Error en fila {index}: {e}", flush=True)
```

Enriquecimiento temporal con tabla calendario

A la tabla **Calendar** se le incorporó una columna adicional con la abreviatura del mes (**month_abbr**) y se establecieron relaciones clave entre **post_date** (fecha del post) y **date** en el calendario.

Este paso fue fundamental para realizar análisis temporales detallados y conectar ambas tablas mediante una clave foránea.



```
1 -- PASO 4 el paso 3 fue desde python poblar la main tab
2
3 UPDATE Calendar
4 SET month_abbr = CASE month
5     WHEN 1 THEN 'ENE'
6     WHEN 2 THEN 'FEB'
7     WHEN 3 THEN 'MAR'
8     WHEN 4 THEN 'ABR'
9     WHEN 5 THEN 'MAY'
10    WHEN 6 THEN 'JUN'
11    WHEN 7 THEN 'JUL'
12    WHEN 8 THEN 'AGO'
13    WHEN 9 THEN 'SEP'
14    WHEN 10 THEN 'OCT'
15    WHEN 11 THEN 'NOV'
16    WHEN 12 THEN 'DIC'
17 END;
```



```
1 -- PASO 5 Esto crea una nueva columna para guardar la fecha
2 --sin hora y poder vincularla con la tabla Calendar
3 ALTER TABLE SocialMediaPosts
4 ADD post_date DATE;
5
6
7 -- PASO 6 Transformar la fecha con hora (timestamp) en una fecha simple (YYYY-MM-DD)
8 --y la guarda en la nueva columna post_date
9
10 UPDATE SocialMediaPosts
11 SET post_date = CAST(timestamp AS DATE);
12
13 --PASO 7: Crear la clave foranea entre SocialMediaPosts y Calendar
14
15 ALTER TABLE SocialMediaPosts
16 ADD CONSTRAINT FK_Posts_Calendar
17 FOREIGN KEY (post_date)
18 REFERENCES Calendar(date);
19
```

Análisis exploratorio de datos (EDA) en Azure Data Studio

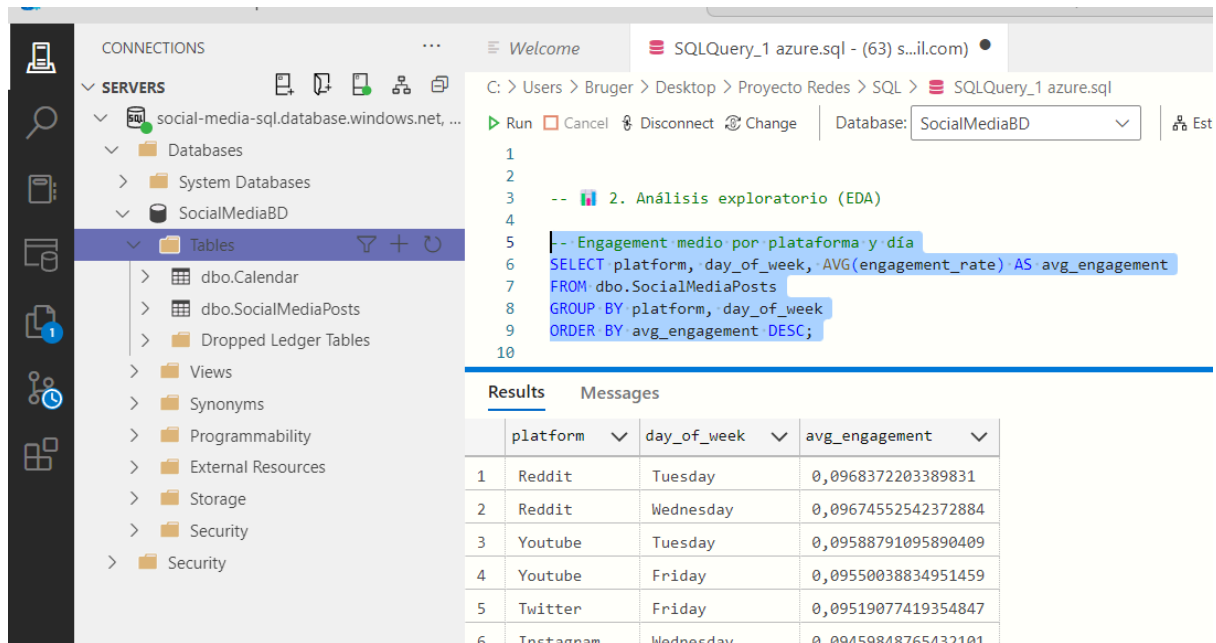
Complementariamente al análisis realizado en Python, se ejecutaron consultas directamente en Azure Data Studio para obtener insights a partir de SQL.

1. Engagement por plataforma y día

Esta consulta permitió identificar qué combinaciones de plataforma y día de la semana generan mayor engagement promedio.

Los resultados mostraron que Reddit y YouTube, los días martes y viernes, presentan los niveles más altos de interacción, seguidos por Twitter e Instagram.

Esta información es clave para optimizar la programación de publicaciones y maximizar el alcance y la participación de los usuarios.



The screenshot shows the Azure Data Studio interface. On the left, the 'CONNECTIONS' pane shows a server named 'social-media-sql.database.windows.net, ...' with a database 'SocialMediaBD' and a table 'dbo.SocialMediaPosts'. The main editor shows a SQL query for an exploratory data analysis (EDA) to find the average engagement rate by platform and day of the week. The query is as follows:

```
-- 2. Análisis exploratorio (EDA)
-- Engagement medio por plataforma y día
SELECT platform, day_of_week, AVG(engagement_rate) AS avg_engagement
FROM dbo.SocialMediaPosts
GROUP BY platform, day_of_week
ORDER BY avg_engagement DESC;
```

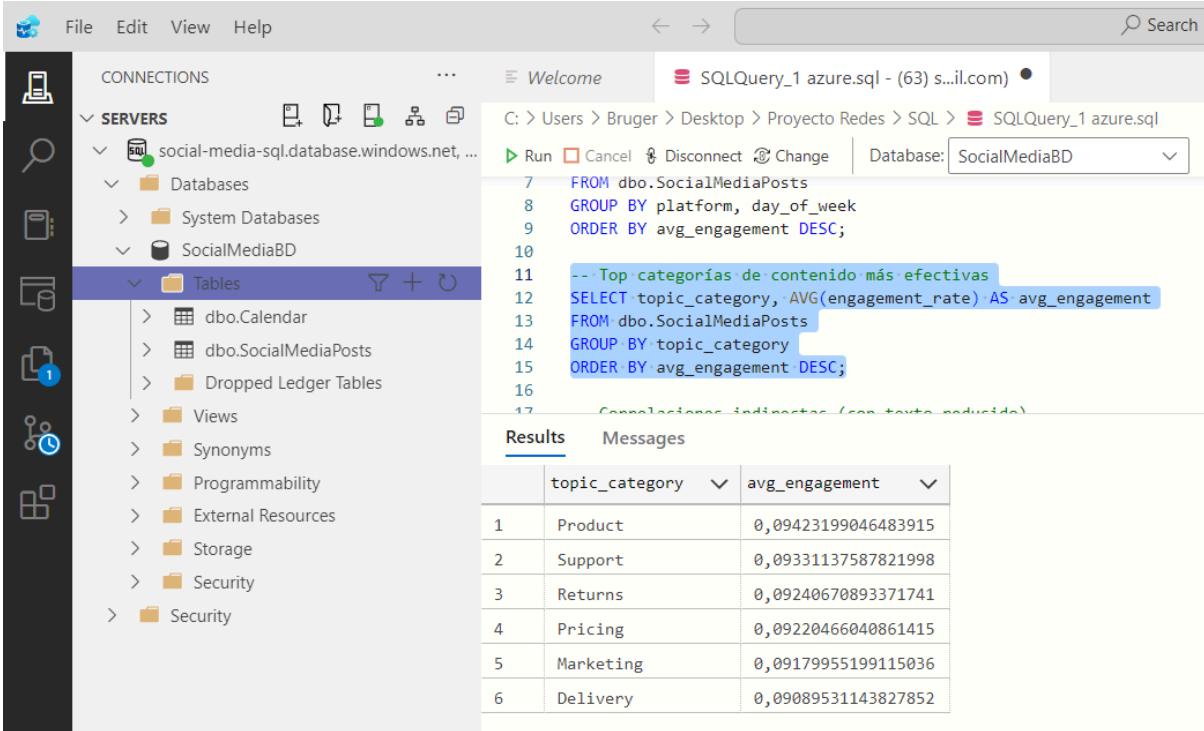
The 'Results' pane shows the following data:

	platform	day_of_week	avg_engagement
1	Reddit	Tuesday	0,0968372203389831
2	Reddit	Wednesday	0,09674552542372884
3	Youtube	Tuesday	0,09588791095890409
4	Youtube	Friday	0,09550038834951459
5	Twitter	Friday	0,09519077419354847
6	Instagram	Wednesday	0,09459848765432101

2. Categorías de contenido más efectivas

Se evaluó qué tipo de contenido genera mayor interacción en redes sociales.

Las publicaciones relacionadas con productos, soporte y devoluciones obtuvieron los niveles más altos de engagement promedio. Esto sugiere que los usuarios responden más activamente a contenidos funcionales y orientados a la experiencia del cliente, en comparación con temas promocionales como marketing o precios.



The screenshot shows the SQL Server Enterprise Manager interface. On the left, the 'SERVERS' tree is expanded to show the 'SocialMediaBD' database and its 'Tables' folder. The 'dbo.SocialMediaPosts' table is selected. The main pane displays a SQL query in the 'SQLQuery_1 azure.sql' file. The query is as follows:

```
7 FROM dbo.SocialMediaPosts
8 GROUP BY platform, day_of_week
9 ORDER BY avg_engagement DESC;
10
11 -- Top categorías de contenido más efectivas
12 SELECT topic_category, AVG(engagement_rate) AS avg_engagement
13 FROM dbo.SocialMediaPosts
14 GROUP BY topic_category
15 ORDER BY avg_engagement DESC;
16
17 -- Correlaciones indirectas (con texto reduido)
```

Below the query, the 'Results' tab is active, showing a table with two columns: 'topic_category' and 'avg_engagement'. The results are as follows:

	topic_category	avg_engagement
1	Product	0,09423199046483915
2	Support	0,09331137587821998
3	Returns	0,09240670893371741
4	Pricing	0,09220466040861415
5	Marketing	0,09179955199115036
6	Delivery	0,09089531143827852

3. Correlaciones indirectas: longitud del texto, sentimiento y engagement

Esta consulta busca explorar **relaciones indirectas** entre la longitud del texto y otras métricas de impacto. Por ejemplo:

- ¿Los textos más cortos o más largos tienden a tener mejor engagement?
- ¿Hay alguna relación entre el sentimiento del texto y la cantidad de likes?
- ¿Los textos positivos generan más interacción?

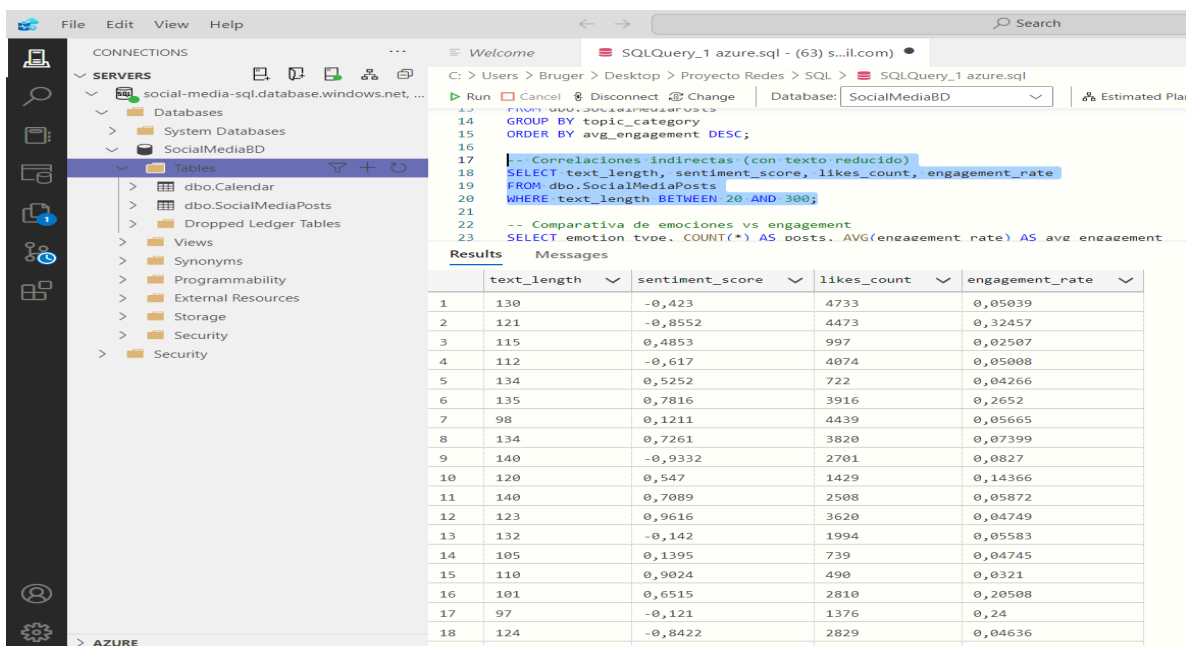
Se analizó si existen relaciones indirectas entre la longitud del texto, el sentimiento, la cantidad de likes y el engagement.

Se filtraron los textos con una longitud entre 20 y 300 caracteres, representando el rango más común en redes sociales.

Resultados observados:

- No hay una relación lineal clara entre la longitud del texto y el engagement.
- Algunas publicaciones con sentimiento negativo elevado (por ejemplo, -0.8552 y -0.9332) lograron tasas de engagement sorprendentemente altas (0.32457 y 0.0827), lo cual sugiere que el contenido polémico o emocionalmente cargado puede generar más interacción.
- También se observaron publicaciones con sentimiento positivo alto (0.7308, 0.7816) y engagement elevado (0.16573, 0.2652), lo que indica que el tono positivo también puede ser efectivo.

En resumen, tanto el contenido emocionalmente positivo como negativo pueden generar alto engagement, dependiendo del contexto y la audiencia.



The screenshot shows a SQL Server Enterprise Manager interface. On the left, the 'SERVERS' tree is expanded to show the 'SocialMediaBD' database. The 'Tables' folder is selected, and 'dbo.SocialMediaPosts' is highlighted. The main pane displays a SQL query and its results.

Query:

```

14 GROUP BY topic_category
15 ORDER BY avg_engagement DESC;
16
17 -- Correlaciones indirectas (con texto reducido)
18 SELECT text_length, sentiment_score, likes_count, engagement_rate
19 FROM dbo.SocialMediaPosts
20 WHERE text_length BETWEEN 20 AND 300;
21
22 -- Comparativa de emociones vs engagement
23 SELECT emotion tvpe, COUNT(*) AS posts, AVG(engagement_rate) AS avg_engagement

```

Results:

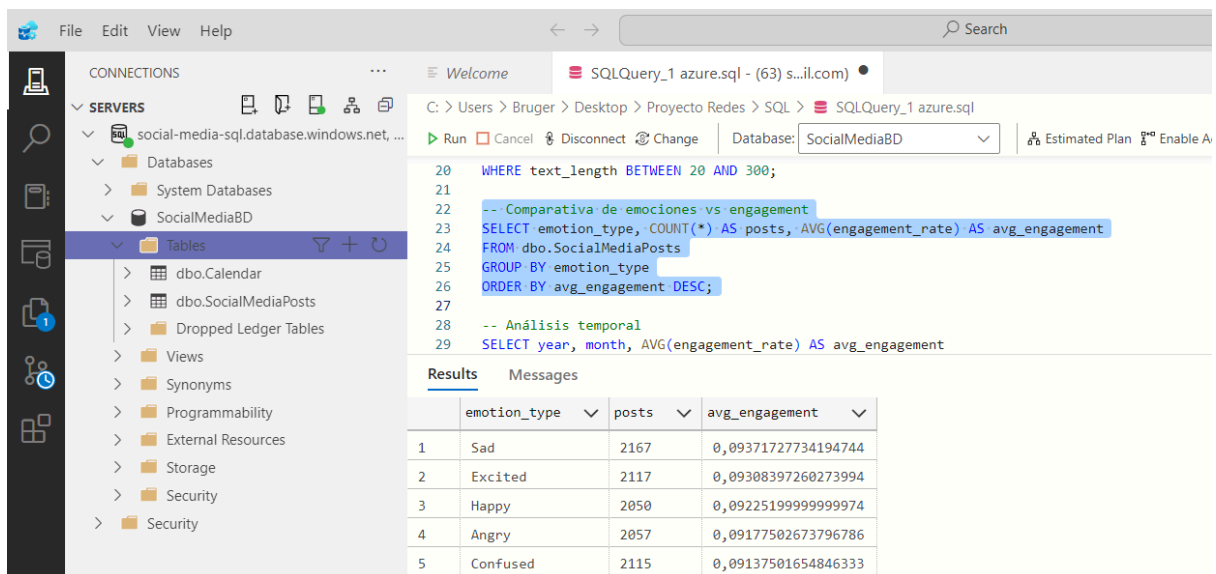
	text_length	sentiment_score	likes_count	engagement_rate
1	130	-0,423	4733	0,05039
2	121	-0,8552	4473	0,32457
3	115	0,4853	997	0,02507
4	112	-0,617	4074	0,05008
5	134	0,5252	722	0,04266
6	135	0,7816	3916	0,2652
7	98	0,1211	4439	0,05665
8	134	0,7261	3820	0,07399
9	140	-0,9332	2701	0,0827
10	120	0,547	1429	0,14366
11	140	0,7089	2508	0,05872
12	123	0,9616	3620	0,04749
13	132	-0,142	1994	0,05583
14	105	0,1395	739	0,04745
15	110	0,9024	490	0,0321
16	101	0,6515	2810	0,20508
17	97	-0,121	1376	0,24
18	124	-0,8422	2829	0,04636

4. Engagement según emoción expresada

Esta consulta analizó cómo varía el engagement promedio según el tipo de emoción expresada en las publicaciones.

Las publicaciones clasificadas como "Sad" obtuvieron el mayor engagement promedio, seguidas por emociones positivas como "Excited" y "Happy". Emociones como "Angry" y "Confused" también mostraron niveles de engagement relativamente altos.

Esto refuerza la hipótesis de que el contenido emocionalmente intenso, tanto negativo como positivo, tiende a captar mayor atención e interacción en redes sociales.



The screenshot shows the SQL Server Enterprise Manager interface. The left pane displays the 'SocialMediaBD' database structure, including tables like 'dbo.Calendar' and 'dbo.SocialMediaPosts'. The right pane shows a SQL query being executed against the 'SocialMediaBD' database. The query is a SELECT statement that filters posts by text length and calculates the average engagement rate for different emotion types, ordered by engagement rate in descending order.

```
20 WHERE text_length BETWEEN 20 AND 300;
21
22 -- Comparativa de emociones vs engagement
23 SELECT emotion_type, COUNT(*) AS posts, AVG(engagement_rate) AS avg_engagement
24 FROM dbo.SocialMediaPosts
25 GROUP BY emotion_type
26 ORDER BY avg_engagement DESC;
27
28 -- Análisis temporal
29 SELECT year, month, AVG(engagement_rate) AS avg_engagement
```

The 'Results' tab shows the following data:

	emotion_type	posts	avg_engagement
1	Sad	2167	0,09371727734194744
2	Excited	2117	0,09308397260273994
3	Happy	2050	0,09225199999999974
4	Angry	2057	0,09177502673796786
5	Confused	2115	0,09137501654846333

5. Análisis temporal del engagement

El análisis temporal reveló que el mes con mayor engagement promedio fue septiembre de 2024 (0.0965), seguido por agosto de 2024 (0.0951) y marzo de 2025 (0.0940). El mes con menor engagement fue junio de 2024 (0.0863).

Estos resultados pueden servir como referencia para identificar temporadas de mayor efectividad en las campañas de contenido.

SERVERS

social-media-sql.database.windows.net, ...

Databases

System Databases

SocialMediaBD

Tables

dbo.Calendar

dbo.SocialMediaPosts

Dropped Ledger Tables

Views

Synonyms

Programmability

External Resources

Storage

Security

Security

C: > Users > Brugger > Desktop > Proyecto Redes > SQL > SQLQuery_1 azure.sql

Run
Cancel
Disconnect
Change
Database: SocialMediaBD

```

26 ORDER BY avg_engagement DESC;
27
28 -- Análisis temporal
29 SELECT year, month, AVG(engagement_rate) AS avg_engagement
30 FROM dbo.SocialMediaPosts
31 GROUP BY year, month
32 ORDER BY avg_engagement DESC;
33
34

```

Results Messages

	year	month	avg_engagement
1	2024	9	0,0964816433566433
2	2024	8	0,09513389965792482
3	2025	2	0,09412023456790124
4	2025	3	0,09395032222222209
5	2024	5	0,09375899113082024
6	2024	10	0,09288927631578948
7	2025	1	0,09285789177001132
8	2024	7	0,0927078587962963
9	2024	11	0,09097673289183218
10	2025	4	0,09095317912218281
11	2024	12	0,08934618778280533
12	2024	6	0,08628399768250296

