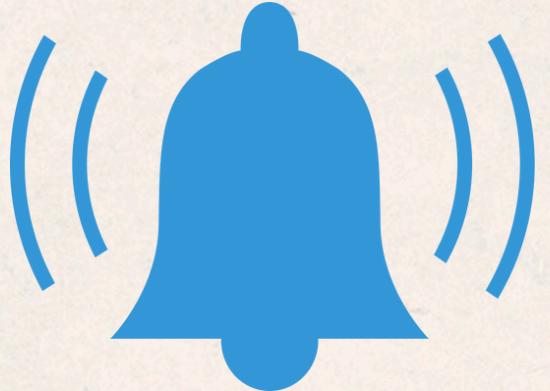
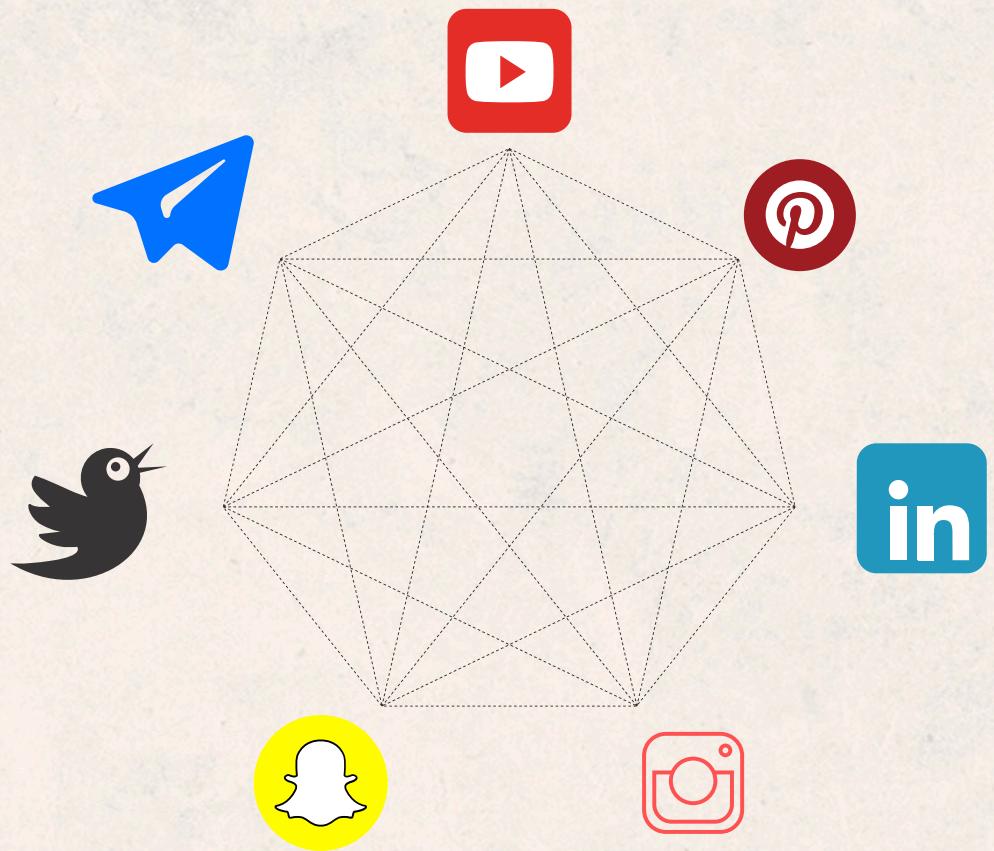
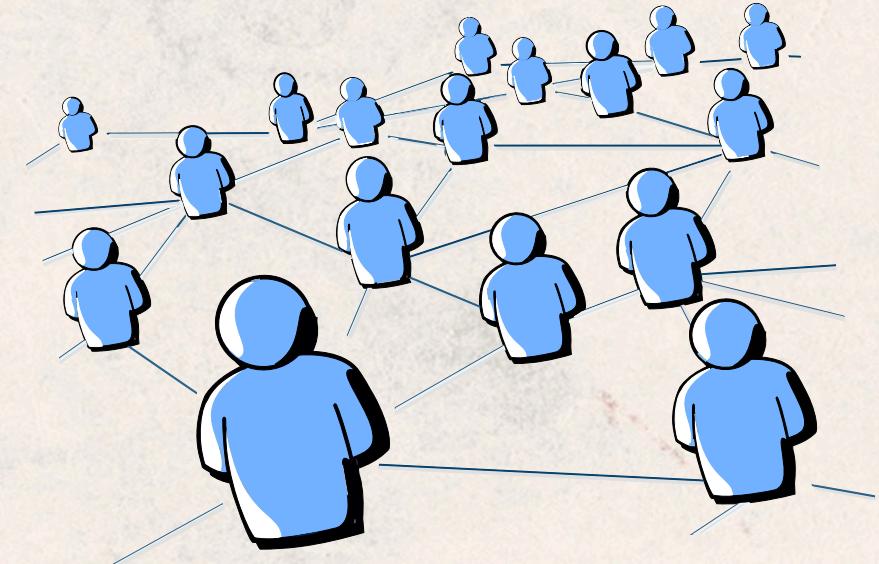
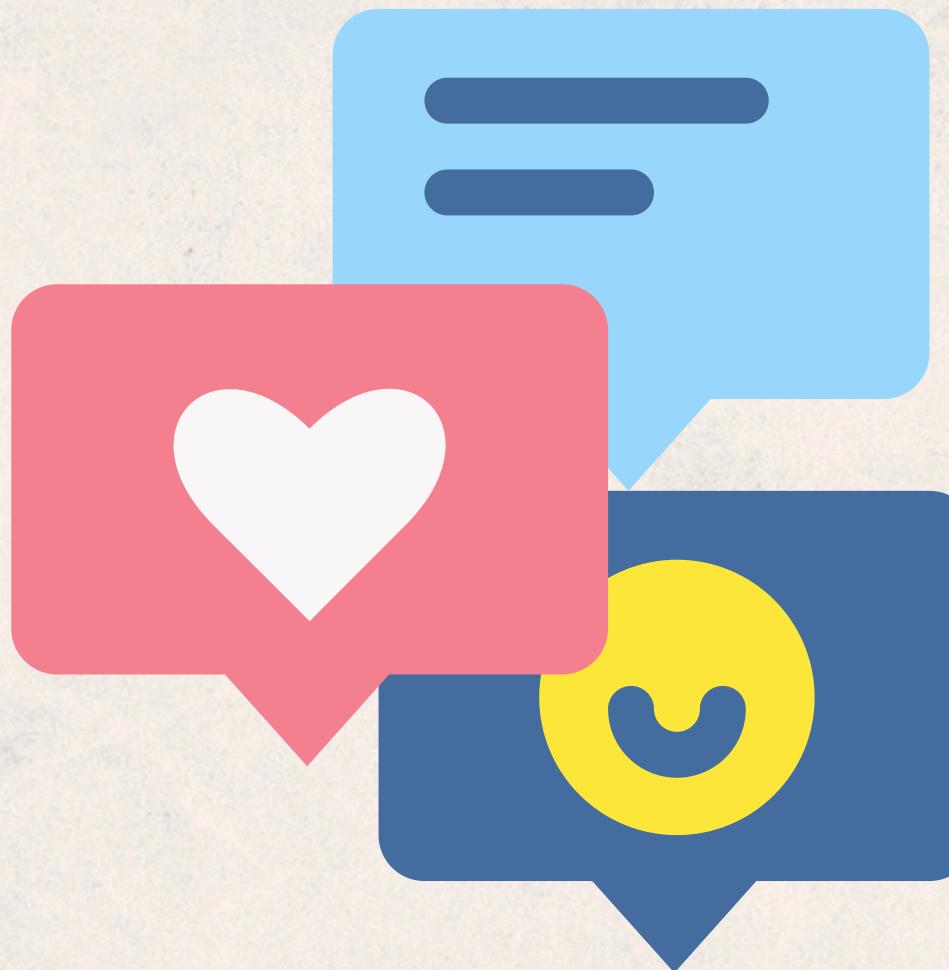




SOCIAL MEDIA SENTIMENT ANALYSIS



INTRODUCTION



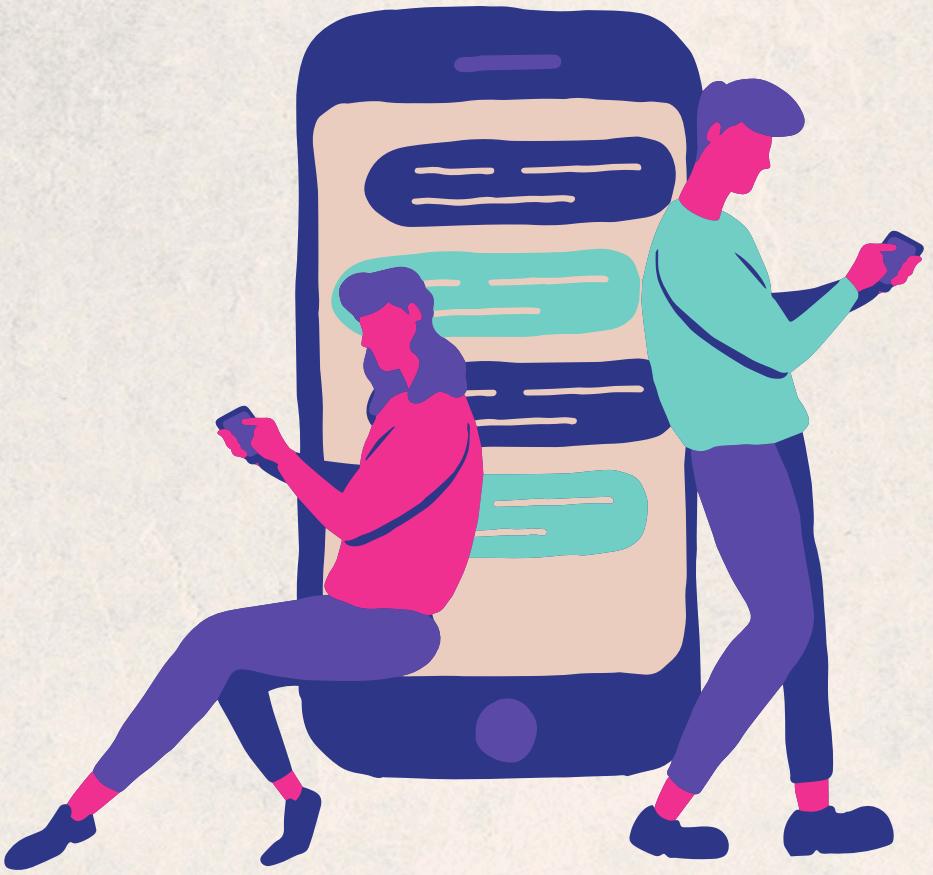
Social media has become a crucial platform for individuals and brands to share thoughts, opinions, and engage with audiences worldwide. With the ever-growing volume of content generated on platforms like Twitter, Facebook, and Instagram, analyzing user sentiment and engagement trends has become essential for understanding public perception and driving strategic decisions.

This project focuses on analyzing social media posts, user interactions, and trending hashtags to uncover sentiment patterns and engagement metrics. Using SQL, I developed a comprehensive data analysis pipeline that reveals insights into user behavior, content effectiveness, and platform-specific trends.

OBJECTIVES

- Sentiment Analysis: Evaluate the emotional tone (positive, negative, neutral) of posts and comments to gauge public opinion.
- Hashtag Performance: Identify trending hashtags and their impact on sentiment, brand visibility, and user engagement.
- User Engagement Insights: Analyze how users interact with content through likes, shares, and comments across different platforms.
- Content Strategy: Understand what types of posts generate the most interaction and what timeframes see peak engagement.

This analysis provides valuable insights into the dynamics of social media, allowing brands and individuals to better tailor their content and improve audience engagement.



-- Compare Sentiment Scores for Different Social Media Platforms:

MySQL Workbench

Local instance MySQL80 (social_...x) unconnected Local instance MySQL80 (soc... x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- school
- school2
- school3
- social_media_analy** (selected)
- Tables
- Views
- Stored Procedures
- Functions
- spotify
- student_mentalhealth
- sys
- titanic

Design

Elements

Text

Brands

Uploads

Draw

Project

Query 1 Q1 Q2 Q3 Q4 Q5 Q6

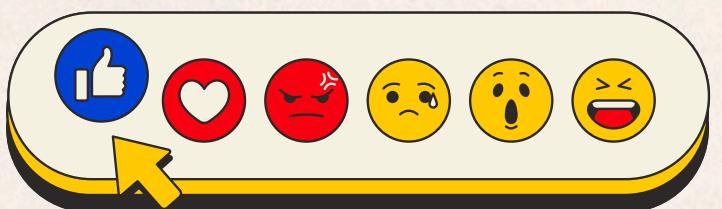
2 • 2 • SELECT p.platform, AVG(
3 CASE
4 WHEN p.sentiment = 'Positive' THEN 1
5 WHEN p.sentiment = 'Neutral' THEN 0
6 WHEN p.sentiment = 'Negative' THEN -1
7 ELSE NULL
8 END
9) AS avg_sentiment_score
10 FROM posts p
11 GROUP BY p.platform;

Administration Schemas

No object selected

Result Grid | Filter Rows: Export: Wrap Cell Content:

platform	avg_sentiment_score
Twitter	-0.1111
Facebook	0.6250
Instagram	0.5000



-- CALCULATE THE AVERAGE SENTIMENT SCORE FOR EACH USER:



MySQL Workbench

Local instance MySQL80 (social_...x) unconnected Local instance MySQL80 (soc... x

File Edit View Query Database Server Tools Scripting Help

SQL SQL+ Data Editor Database Browser

Navigator Query 1 Q1 Q2 x

SCHEMAS Filter objects

- school
- school2
- school3
- social_media_analy** (selected)
- Tables
- Views
- Stored Procedures
- Functions
- spotify
- student_mentalhealth
- sys
- titanic

Design Elements Text Branding Information

No object selected

```
3 • SELECT
4     users.user_id,
5     users.username,
6     AVG(CASE
7         WHEN comments.sentiment = 'Positive' THEN 1
8         WHEN comments.sentiment = 'Neutral' THEN 0
9         WHEN comments.sentiment = 'Negative' THEN -1
10        ELSE NULL
11    END) AS avg_sentiment_score
12
13    FROM
14        comments
15        JOIN
16            users ON comments.user_id = users.user_id
17        GROUP BY user_id
18        ORDER BY user_id;
```

Result Grid Filter Rows: Export: Wrap Cell Content: Read Only

	user_id	username	avg_sentiment_score
▶	1	user1	0.6364
	2	user2	0.8333
	3	user3	0.8182
	4	user4	0.8182
	5	user5	0.8000

Result 1 x Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	19:09:21	SELECT hashtag, frequency, sentiment FROM hashtags ORDER BY frequency DESC , sentiment LIMIT 5	5 row(s) returned	0.016 sec / 0.000 sec
2	19:15:18	SELECT users.user_id, users.username, AVG(CASE WHEN comments.sentiment = 'Positive' TH...	5 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

-- FIND THE MOST ENGAGING POST BASED ON LIKES, SHARES, AND COMMENTS:

MySQL Workbench

Local instance MySQL80 (social_...x) unconnected Local instance MySQL80 (soc... x

File Edit View Query Database Server Tools Scripting Help

Navigator: Query 1 Q1 Q2 Q3 | Limit to 1000 rows

SCHEMAS: school, school2, school3, social_media_analy, spotify, student_mentalhealth, sys, titanic

3 • SELECT

4 post_id,
5 content,
6 (posts.likes + posts.shares + posts.comments) AS total_engagement
7 FROM
8 posts
9 ORDER BY total_engagement DESC
10 LIMIT 1;

No object selected

Result Grid | Filter Rows: Export: Wrap Cell Content: Fetch rows: Result 1 x Read Only

post_id	content	total_engagement
44	Happy to see my favorite band perform live!	420

Action Output

#	Time	Action	Message	Duration / Fetch
2	19:15:18	SELECT users.user_id, users.username, AVG(CASE WHEN comments.sentiment = 'Positive' THEN 1 ELSE 0 END) AS positive_ratio FROM users JOIN comments ON users.user_id = comments.user_id WHERE users.username = 'JohnDoe' GROUP BY users.user_id ORDER BY positive_ratio DESC LIMIT 1;	5 row(s) returned	0.000 sec / 0.000 sec
3	19:17:46	SELECT post_id, content, (posts.likes + posts.shares + posts.comments) AS total_engagement FROM posts ORDER BY total_engagement DESC LIMIT 1;	1 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

83°F Mostly cloudy Search ENG IN 19:18 21-08-2024 PRE



-- Analyze the Relationship Between Post Frequency and Sentiment:



MySQL Workbench

Local instance MySQL80 (social_...) unconnected Local instance MySQL80 (soc...)

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

- school
- school2
- school3
- social_media_analy** (selected)
- Tables
- Views
- Stored Procedures
- Functions
- spotify
- student_mentalhealth
- sys
- titanic

Design

Elements

Text

Brands

Administration Schemas

Information

No object selected

Upload

Draw

Project

Apps

Object Info Session

Query 1 Q1 Q2 Q3 Q4

```
3 •  SELECT h.hashtag, COUNT(p.post_id) AS post_count, AVG(
4   CASE
5     WHEN p.sentiment = 'Positive' THEN 1
6     WHEN p.sentiment = 'Neutral' THEN 0
7     WHEN p.sentiment = 'Negative' THEN -1
8     ELSE NULL
9   END
10 ) AS avg_sentiment_score
11 FROM hashtags h
12 JOIN posts p ON h.post_id = p.post_id
13 GROUP BY h.hashtag;
```

Result Grid | Filter Rows: Export: Wrap Cell Content: Result 1 Read Only

hashtag	post_count	avg_sentiment_score
#sunnyday	1	1.0000
#goodvibes	1	1.0000
#traffic	1	-1.0000
#movienight	1	1.0000
#tired	1	0.0000
#workout	1	1.0000
#brunch	1	1.0000
#rain	1	0.0000

Action Output

#	Time	Action	Message	Duration / Fetch
3	19:17:46	SELECT post_id, content, (posts.likes + posts.shares + posts.comments) AS total_engagement FRO...	1 row(s) returned	0.000 sec / 0.000 sec
4	19:20:15	SELECT h.hashtag, COUNT(p.post_id) AS post_count, AVG(CASE WHEN p.sentiment = 'Positive' ...	20 row(s) returned	0.000 sec / 0.000 sec

-- Identify the Top 5 Hashtags by Sentiment and Frequency:

The screenshot shows the MySQL Workbench interface with a query editor and results grid.

Query Editor:

```
1
2 -- Identify the Top 5 Hashtags by Sentiment and Frequency:
3 • SELECT
4     hashtag, frequency, sentiment
5 FROM
6     hashtags
7 ORDER BY frequency DESC , sentiment
8 LIMIT 5;
```

Results Grid:

hashtag	frequency	sentiment
#music	310	Positive
#concert	300	Positive
#beachtrip	290	Positive
#adventure	250	Positive
#travel	250	Positive

Output Log:

#	Time	Action	Message	Duration / Fetch
1	19:09:21	SELECT	hashtag,frequency,sentiment FROM hashtags ORDER BY frequency DESC , sentiment LIMIT 5 5 row(s) returned	0.016 sec / 0.000 sec

System Tray:

- 83°F Mostly cloudy
- Search bar
- Taskbar icons: File Explorer, Microsoft Edge, ChatGPT, Google Chrome, Task View, Task Manager, Power User Menu, File Explorer, Task View, Task Manager, Power User Menu
- System icons: Volume, Battery, Network, Language (ENG IN), Date (21-08-2024), Time (19:11)

- Find Users with the Most Negative Sentiments in Comments:



File Edit View Query Database Server Tools Scripting Help

SQL SQL+ Database Object Navigator Query 1 Q1 Q2 Q3 Q4 Q5

Design Element Text Brand Upload Draw

Schemas

Filter objects

- school
- school2
- school3
- social_media_analy** ▶
 - Tables
 - Views
 - Stored Procedures
 - Functions
- spotify
- student_mentalhealth
- sys
- titanic

Administration Schemas

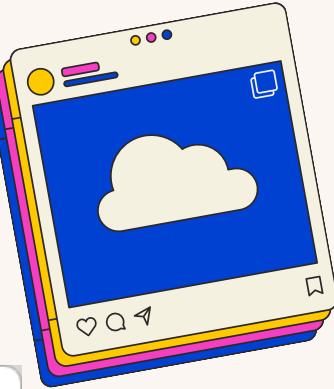
No object selected

```
3 • SELECT
4     users.username,
5     users.user_id,
6     COUNT(comments.comment_id) AS negative_comments
7 FROM
8     users
9     JOIN
10    comments ON users.user_id = comments.user_id
11 WHERE
12    sentiment = 'Negative'
13 GROUP BY users.username , users.user_id
14 ORDER BY negative_comments DESC
15 LIMIT 5;
```

Result Grid Filter Rows: Export: Wrap Cell Content:

	username	user_id	negative_comments
▶	user1	1	1
	user4	4	1

-- Analyze the Impact of Hashtags on Post Sentiment:



MySQL Workbench

Local instance MySQL80 (social_... unconnected Local instance MySQL80 (soc...)

File Edit View Query Database Server Tools Scripting Help

Navigator Query 1 Q1 Q2 Q3 Q4 Q5 Q6 Q7

Limit to 1000 rows

SCHEMAS

Filter objects

school school2 school3 social_media_analy: Tables Views Stored Procedures Functions spotify student_mentalhealth sys titanic

Administration Schemas

No object selected

```
2 • SELECT h.hashtag, AVG(
3   CASE
4     WHEN p.sentiment = 'Positive' THEN 1
5     WHEN p.sentiment = 'Neutral' THEN 0
6     WHEN p.sentiment = 'Negative' THEN -1
7     ELSE NULL
8   END
9 ) AS avg_post_sentiment
10 FROM hashtags h
11 JOIN posts p ON h.post_id = p.post_id
12 GROUP BY h.hashtag
13 ORDER BY avg_post_sentiment DESC;
14
```

Result Grid | Filter Rows: Export: Wrap Cell Content: Result 1 Read Only

hashtag	avg_post_sentiment
#sunnyday	1.0000
#goodvibes	1.0000
#movienight	1.0000
#workout	1.0000
#brunch	1.0000
#concert	1.0000
#weekend	1.0000
#adventure	1.0000
#travel	1.0000

Analyze the Sentiment Distribution Across Different Age Groups:



MySQL Workbench

Local instance MySQL80 (social_... X unconnected X Local instance MySQL80 (soc... X

File Edit View Query Database Server Tools Scripting Help

Navigator Query 1 Q1 Q2 Q3 Q4 Q5 Q6 Q7 Q8*

SCHEMAS Filter objects

- school
- school2
- school3
- social_media_analy** (selected)
- Tables
- Views
- Stored Procedures
- Functions
- spotify
- student_mentalhealth
- sys
- titanic

No object selected

```
2 •  SELECT
3   CASE
4     WHEN u.age < 20 THEN 'Under 20'
5     WHEN u.age BETWEEN 20 AND 29 THEN '20-29'
6     WHEN u.age BETWEEN 30 AND 39 THEN '30-39'
7     ELSE '40 and above'
8   END AS age_group,
9   AVG(
10    CASE
11      WHEN p.sentiment = 'Positive' THEN 1
12      WHEN p.sentiment = 'Neutral' THEN 0
13      WHEN p.sentiment = 'Negative' THEN -1
14      ELSE NULL
15    END
16  ) AS avg_sentiment_score
17  FROM users u JOIN posts p ON u.user_id = p.user_id
18  GROUP BY age_group ORDER BY avg_sentiment_score DESC;
```

Result Grid | Filter Rows: _____ | Export: _____ | Wrap Cell Content: _____

	age_group	avg_sentiment_score
▶	30-39	0.5294
	20-29	0.2121

-- Find the Most Popular Hashtags Associated with Positive Sentiment:

The screenshot shows the MySQL Workbench interface with a query editor window. The query is as follows:

```
2 •  SELECT h.hashtag, COUNT(h.post_id) AS hashtag_usage, AVG(
3   CASE
4     WHEN p.sentiment = 'Positive' THEN 1
5     WHEN p.sentiment = 'Neutral' THEN 0
6     WHEN p.sentiment = 'Negative' THEN -1
7     ELSE NULL
8   END
9 ) AS avg_sentiment_score
10 FROM hashtags h
11 JOIN posts p ON h.post_id = p.post_id
12 GROUP BY h.hashtag
13 HAVING avg_sentiment_score > 0.7
14 ORDER BY hashtag_usage DESC
15 LIMIT 5;
16
```

The result grid shows the following data:

hashtag	hashtag_usage	avg_sentiment_score
#sunnyday	1	1.0000
#goodvibes	1	1.0000
#movienight	1	1.0000
#workout	1	1.0000
#brunch	1	1.0000

FOLLOW

-- Track Engagement Trends Based on Hashtag Sentiment Over Time:

The screenshot shows a SQL development environment with the following interface elements:

- Toolbar:** File, Edit, View, Query, Database, Server, Tools, Scripting, Help.
- Navigator:** Schemas (school, school2, school3, social_media_analytic, spotify, student_mentalhealth, sys, titanic).
- Query Editor:** Shows a complex SQL SELECT statement. The statement uses a CASE expression to calculate an average sentiment score based on post sentiment ('Positive', 'Neutral', 'Negative'). It joins hashtags, posts, and engagement tables, groups by hashtag and post date, and orders the results by post date and average sentiment score in descending order.

```
2 •  SELECT h.hashtag, p.post_date, COUNT(e.engagement_id) AS total_engagements, AVG(
3   CASE
4     WHEN p.sentiment = 'Positive' THEN 1
5     WHEN p.sentiment = 'Neutral' THEN 0
6     WHEN p.sentiment = 'Negative' THEN -1
7     ELSE NULL
8   END
9 ) AS avg_sentiment_score
10 FROM hashtags h
11 JOIN posts p ON h.post_id = p.post_id
12 LEFT JOIN engagement e ON p.post_id = e.post_id
13 GROUP BY h.hashtag, p.post_date
14 ORDER BY p.post_date, avg_sentiment_score DESC;
```

- Result Grid:** Displays the query results in a tabular format. The columns are hashtag, post_date, total_engagements, and avg_sentiment_score. The data shows various hashtags and their engagement statistics.

hashtag	post_date	total_engagements	avg_sentiment_score
#sunnyday	2023-08-01	2	1.0000
#goodvibes	2023-08-01	2	1.0000
#traffic	2023-08-02	2	-1.0000
#movienight	2023-08-03	2	1.0000
#tired	2023-08-04	2	0.0000
#workout	2023-08-05	1	1.0000
#brunch	2023-08-06	1	1.0000
#rain	2023-08-07	1	0.0000
#concert	2023-08-09	1	1.0000

- Bottom Right:** A large black outline of a thumbs-up icon is overlaid on the interface.

THANKYOU

