**Instagram Data Warehouse Implementation Report**

**1. Assignment Overview**

The design and deployment of the Instagram Data Warehouse leverage Databricks and Delta Lake for ETL tasks and analytics functions. The data acquisition process includes structured extraction from various sources, transformations into a star schema structure, and loading for analytical querying and business intelligence insights.

**2. Data Sources**

The dataset originates from simulated Instagram interactions and engagement data, including user activities such as likes, comments, and follows. The raw data analysis consists of millions of records presenting information about user profiles, photos, engagement metrics, and social interactions.

**Dataset Overview**

The Instagram data warehouse collects and structures user interaction data to analyze engagement trends and performance. The dataset includes:

* **Users Data**: Profile details, account creation date, and activity logs.
* **Photos Data**: Image metadata, upload timestamps, and associated tags.
* **Engagement Data**: Like, comment, and follow actions.

**Potential Use Cases**

1. **User Engagement Analysis**
   * Identify the most active users and their engagement levels.
   * Track social influence by analyzing followers and interactions.
2. **Content Performance Benchmarking**
   * Measure the most liked and commented photos over time.
   * Determine content trends based on tags and interactions.
3. **Influencer Identification**
   * Discover users with the highest engagement scores.
   * Analyze relationships between engagement and follower growth.
4. **Predictive Engagement Modeling**
   * Utilize machine learning to predict trending content.
   * Forecast user behavior based on historical activity.
5. **Advertising and Monetization Insights**
   * Analyze user engagement with sponsored content.
   * Optimize advertisement targeting based on interaction patterns.

**Dataset Files**

The following datasets were used:

|  |  |
| --- | --- |
| **Dataset File Name** | **Description** |
| users.csv | Contains user profile details, including usernames, creation dates, and activity logs. |
| photos.csv | Metadata about uploaded photos, including URLs, timestamps, and associated tags. |
| likes.csv | Records of user interactions in the form of likes. |
| comments.csv | Contains user comments and associated timestamps. |
| follows.csv | Tracks users follow actions and relationships. |
| tags.csv | Stores hashtags and labels associated with photos. |

A full data dictionary is provided in the README file accompanying this report.

**3. Normalized Database**

The normalized database contains multiple tables linked by foreign keys to maintain referential integrity and optimize storage. The primary relationships between tables are as follows:

* **Users Table**: Stores user profiles and activity logs.
* **Photos Table**: Stores metadata about uploaded content.
* **Likes, Comments, and Follows Tables**: Track engagement actions.
* **Tags Table**: Links hashtags to photos.

**4. ETL Process Implementation**

**ETL Steps:**

1. **Extract:** Data is loaded from CSV files in Azure Blob Storage.
2. **Transform:** Data is cleaned, formatted, and structured into a normalized schema.
3. **Load:** Normalized data is inserted into staging tables and later transformed into dimensional tables and fact tables.

**ETL Challenges and Solutions:**

* **Handling Missing Data:** Null values are either imputed or removed.
* **Data Deduplication:** Ensures unique records for each user and interaction.
* **Surrogate Keys:** Implemented in dimension tables for consistency.
* **Slowly Changing Dimensions (SCD Type 2):** Tracks historical changes in user and photo data.

**ETL Execution Logs:**

Screenshots and logs of successful execution are included in the README file.

**5. Data Warehouse Design: Star Schema**

**Dimension Tables:**

|  |  |
| --- | --- |
| Table Name | Description |
| dim\_users | Stores user attributes such as username, account creation date, and engagement metrics. |
| dim\_photos | Contains metadata about uploaded photos, including URLs and timestamps. |
| dim\_tags | Stores hashtag and label associations for content. |
| dim\_interaction\_type | Defines different types of engagement (likes, comments, follows). |

**Fact Tables:**

|  |  |
| --- | --- |
| Table Name | Description |
| fact\_interactions | Stores all user engagement data, including likes, comments, and follows. |

**Star Schema Design**

The fact table **fact\_interactions** is linked to multiple dimension tables to enable efficient querying and reporting. The structure is optimized for aggregating engagement metrics.

**6. Analytical Queryig and Business Insights**

**Query 1: Identify Top 5 Most Liked Photos**

**Business Insight:** Helps identify content trends and popular posts.

SELECT p.photo\_url, u.username AS uploaded\_by, COUNT(f.interaction\_type\_sk) AS total\_likes

FROM fact\_interactions f

JOIN dim\_photos p ON f.photo\_sk = p.sk\_id

JOIN dim\_users u ON p.user\_id = u.user\_id

WHERE f.interaction\_type\_sk = (SELECT interaction\_type\_sk FROM dim\_interaction\_type WHERE interaction\_type = 'like')

GROUP BY p.photo\_url, u.username

ORDER BY total\_likes DESC

LIMIT 5;

**A graph of a number of likes

AI-generated content may be incorrect.**

**Query 2: Identify Top 10 Most Engaging Users**

**Business Insight:** Determines users with the highest interaction levels.

SELECT ua.username, ua.total\_interactions, ue.received\_interactions,

(ua.total\_interactions + COALESCE(ue.received\_interactions, 0)) AS engagement\_score

FROM (

SELECT u.username, COUNT(f.user\_sk) AS total\_interactions

FROM fact\_interactions f

JOIN dim\_users u ON f.user\_sk = u.sk\_id

GROUP BY u.username

) ua

LEFT JOIN (

SELECT u.username, COUNT(f.followed\_user\_sk) AS received\_interactions

FROM fact\_interactions f

JOIN dim\_users u ON f.followed\_user\_sk = u.sk\_id

WHERE f.followed\_user\_sk IS NOT NULL

GROUP BY u.username

) ue

ON ua.username = ue.username

ORDER BY engagement\_score DESC

LIMIT 10;

**A green bar graph with numbers

AI-generated content may be incorrect.**

**Query 3: Identify Top 10 Most Followed Users**

**Business Insight:** Highlights social influencers based on follower count.

SELECT u.username, COUNT(f.followed\_user\_sk) AS total\_followers

FROM fact\_interactions f

JOIN dim\_users u ON f.followed\_user\_sk = u.sk\_id

WHERE f.interaction\_type\_sk = 3

GROUP BY u.username

ORDER BY total\_followers DESC

LIMIT 10;

**A purple bar graph with white lines

AI-generated content may be incorrect.**

**7. Conclusion and Future Improvements**

This ETL pipeline effectively processes Instagram engagement data, enabling scalable analytics and business insights. The implementation follows best practices, including normalized staging, star schema modeling, and optimized querying.

**Future Enhancements:**

1. **AI-Driven Insights:** Implement sentiment analysis on comments.
2. **Real-Time Analytics:** Introduce Apache Kafka for streaming data processing.
3. **Enhanced Visualization:** Build interactive dashboards with Power BI or Tableau.
4. **Predictive Modeling:** Use machine learning to forecast engagement trends.

The Instagram Data Warehouse provides a scalable, high-performance analytical solution that can be further enhanced for deeper insights into user behavior and social interactions.