

Project of Decision Support Systems

Module II

Anna Monreale, Cristiano Landi

Introduction

This project consists of several assignments that must be completed in sequence. First, you have to analyze the data, design the database, and populate it. Next, you have to solve problems on the database you created using SQL Server Integration Services (SSIS). Minimize the use of SQL commands within nodes and rely on native SSIS components whenever possible¹. Then, you have to create a datacube based on your database and use it to answer business questions. Document the datacube creation process in your report and write MultiDimensional eXpressions (MDX) queries in SQL Server Management Studio (SSMS). Finally, you have to design interactive dashboards. We recommend using the tool demonstrated in class, but you may use alternatives such as [MicroStrategy](#), provided the dashboard is interactive and can connect to the database. In the report, include a screenshot and a short explanation of why each dashboard is analytically interesting.

In the following, you can find a set of incremental assignments, each with a brief description of what you are required to produce and which tools you can use for the task.

This year's project focuses on analyzing data from Truth Social². Attached to this document, you can find the following files³:

- **truth.csv**: contains the main dataset, including details about Truths (posts).
- **users.xml**: contains information about Truth Social accounts.
- **follows.json**: contains, for most users, information about their followers and the accounts they follow.
- **external_urls.json**, **hashtags.json**, **media.json**, and **quotes.json**: contain additional information about Truths (posts).

More importantly, you also have access to the sequences of events associated with each post, including truth (post) creation, likes, re-truths (reposts), and replies. Each of these events represents an individual action occurring at a specific point in time. Due to the high dimensionality and volume of the events dataset, we have uploaded it to the course server. A small subsample can be found in the **subsample_events.xml** file. **You must design and implement a data warehouse for analyzing these events.**

¹SQL commands run only on the server, while SSIS nodes distribute operations between server and client

²<https://truthsocial.com/>

³Data extracted from <https://arxiv.org/pdf/2303.11240> and integrated with https://github.com/kashish-s/TruthSocial_2024ElectionInitiative

Using Python

All tasks must be completed without using any pandas-like package, unless explicitly specified otherwise.

Assignment 1: data preparation and cleaning

Explore and understand the data at your disposal. For the events dataset, use the subsample we provided in **subsample_events.xml**. Are there any data quality issues, such as missing values, outliers, duplicate entries, or other issues? If so, can they be treated *easily*? Can you integrate external data or infer additional information with reasonable effort, for example, spatial information related to locations mentioned in a post? Can you derive further information from the provided data, for instance, from textual fields?

On the DidaWiki page, you can find a link to download a small dataset in the arff format that must be integrated into the main corpus (Data retrieved from <https://github.com/stiles/trump-truth-social-archive> on September 24)

For this assignment, you may use any software or package you prefer.

Assignment 2: post profiling

Using the information collected in the previous assignment, define a set of meaningful categories for the posts based on their topics and/or on other relevant information available in the dataset. The categorization can be performed either manually or by applying data mining techniques. The resulting categories should be clearly defined and interpretable.

For this assignment, you may use any software or package you prefer.

Assignment 3: DW Schema

It is now time to move from operational databases to analysis-oriented data warehouses. When designing the DW schema, consider the events associated with posts as the facts to be analyzed.

Next, create two copies of the data warehouse schema in the database named *Group_ID_DB* (e.g., *Group_01_DB*), as specified in the email containing your group credentials. Specifically, you have to create: one schema where table names have the suffix `_python` and one schema where table names have the suffix `_IS`.

You may use both Python and SQL Server Management Studio to design and create the data warehouse.

Assignment 4: Data preparation

Write a Python program that splits the data into separate csv files, one per table in the data warehouse proposed in the previous step.

Assignment 5: data uploading with Python

Write a Python program that populates the *Group_ID_DB* database according to the data warehouse schema, using the csv files prepared in Assignment 5. You must populate only the tables ending with `_python`.

Please note that this operation may take some time. Design your code accordingly, taking efficiency and scalability into account.

Assignment 6: data gathering

In the server, you can find the complete set of events. Write a Python script to download at least 5% of the data from the complete dataset and save it in a JSONfile. The data can be found in the *ProjectAdditionalData2627* database in the table `events`.

Using Microsoft SQL Server Integration Services

Assignment 7

Using the csv files from Assignment 5, use SSIS to load the data into your group database on the server, populating the schema with tables ending with `_IS` except for the table (or tables) regarding events. This table (or tables) must be populated with the data in the `events` table of the *ProjectAdditionalData2627* database, applying any potential preprocessing identified in assignment 1 using SSIS.

At this point, you should have two fact tables along with their corresponding (duplicated) dimension tables. From now on, **perform all assignments using the fact table that contains all the data**⁴.

Assignment 8

For each user, compute the *night-day score*, defined as the ratio between the number of post released during the night (8pm to 8am) and those released during the day (8am to 8pm). Also, compute the same score using the number of likes.

Assignment 9

For each post category (see Assignment 2) and each month, compute the ratio between the cumulative number of likes for that category and the total number of posts, including reposted ones.

Assignment 10

For each post category (see Assignment 2), compute the ratio of the average number of likes for that category to the average number of likes across all other categories.

⁴We suggest using the smaller fact table to verify the correctness of your results manually

Assignment 11

For each user and category, compute the following statistics:

1. *Trending percentage*: the percentage of trending posts, i.e., posts where the number of likes exceeds the user's average by at least one standard deviation.
2. *Trending factor*: the difference between the number of trending posts and flopping posts, i.e., posts where the number of likes falls below the user's average by at least one standard deviation. This value must be normalized such that -1 indicates only flopping posts, while 1 indicates only trending posts.

Edge cases must be considered and documented in the report. Additionally, save the same statistics in a separate file, only for posts that reposted content from Donald Trump.

Assignment 12

Based on the analytical results of the previous queries and the insights from the data understanding phase, define an **interesting** query and answer it using SSIS. Focus on one or a few accounts with a high number of followers. In your report, explain why this query is interesting.

Assignment 13

Based on the analytical results of the previous queries and the insights from the data understanding phase, define an **interesting** query and answer it using SSIS. In your report, explain why this query is interesting.

Using MDX, SQL Management Studio, and PowerBI

Assignment 14

Build a datacube from the tables in your database, defining the appropriate hierarchies. Create the necessary measures based on the queries you need to answer.

Assignment 15

For each month, show the total events for each account and the overall total for all accounts.

Assignment 16

Compute the average yearly media storage space required. Since the data in our dataset does not directly contain information about file sizes, use the following estimates:

- if the media type is JPG, assume that each file has a size of 400 KB;
- if the media type is PNG, assume that each file has a size of 2 MB;
- if the media type is GIF, assume that each file has a size of 10 MB.

Additionally, compute the grand total across all years.

This query can be answered in two ways: either by counting the same file multiple times when it is used in multiple posts, or by counting each distinct file only once. Specify which of these two approaches your query implements. In your report, also explain how the MDX query could be modified to answer the question using the other approach.

Assignment 17

For each post category, show the percentage increase or decrease in events compared to the previous year.

Assignment 18

For each season, list the categories where the number of likes exceeds the average of the same season in the previous year. Also, report the increase as a percentage.

Assignment 19

Propose and solve a query that reveals interesting and **non-trivial** insights from the previous analyses. In your report, explain why this query is important from an analytical perspective.

Assignment 20

Create a plot or dashboard of your choice that helps highlighting meaningful behavior about the reaction to a specific, relevant event in the dataset.

Assignment 21

Create a plot or dashboard of your choice that provides meaningful insights into Donald Trump's activities over one or more relevant time periods.

Delivery Instructions

When your project is ready, compress all files, including the PDF report (no more than 10 pages), into a .zip file named `LDS_project_GroupID.zip`. Upload this file using the correct Google form with your *studenti.unipi.it* email. If you encounter issues with the form, email all teachers with the subject: **[LDS] Project Delivery Group_ID**. Do not include the original dataset in the zip file.

The project **must** be delivered in two parts:

1. Code from the first assignment through assignment 14 (inclusive) **must** be delivered by December 11th. Google form: forms.gle/rRMbgqWNxy46S8Vc8.
2. Remaining assignments (MDX queries and dashboards) and the report **must** be delivered by December 18th. **Warning:** starting from December 11th, you will no longer be able to modify your group database. Google form: forms.gle/QPWzypHZ541i7gDa6. You can also commit the code delivered in the first part, highlighting any changes in a dedicated file.