Enron Email Analysis

Notebooks as Business Reports

ISYS2001 Introduction to Business Programming

Due: 23:59, Friday 26th May 2022

Introduction

The Enron Corporation, once a leading American energy, commodities, and services company, has become synonymous with corporate fraud and corruption. Established in 1985 by the merger of Houston Natural Gas and InterNorth, Enron quickly grew into one of the largest companies in the United States. At its peak, Enron's market value exceeded $70 billion, with shares trading at an all-time high of around $90. However, the company's rapid growth and seemingly endless success concealed a complex web of fraudulent accounting practices and financial scandals. In 2001, Enron's deceitful activities were exposed, leading to one of the most notorious corporate bankruptcies in American history. The Enron scandal had far-reaching consequences, shaking investor confidence, leading to regulatory reforms, and forever changing the perception of corporate governance.

This assignment aims to analyze the Enron Mail dataset, a collection of email communications from Enron executives, employees, and other associates, to gain insights into the company's communication patterns and uncover potential topics of interest. By leveraging Python programming in Google Colab notebooks, this analysis will explore SQLite databases, data visualization, and the application of industry best practices in programming. Through the examination of email traffic over time, identification of top senders and recipients, distribution by recipient type, subject keyword analysis, and the comparison of internal and external communication, this project seeks to provide a comprehensive understanding of the Enron Mail dataset while highlighting key trends and themes within the data.

Python Notebooks: Enhancing Data-Driven Business Reporting

Python notebooks, such as Jupyter Notebook and Google Colab, have emerged as powerful tools for creating business reports, particularly when working with data-driven projects. One of the main advantages of using notebooks for business reports is their ability to blend code, visualizations, and narrative text within a single, interactive document. This literate programming approach fosters a seamless integration of data analysis, interpretation, and communication, which is vital for presenting insights and findings in a coherent and accessible manner. Notebooks enable users to write and execute code, visualize data, and describe their thought process using markdown cells, making it easier for stakeholders to follow the analysis and grasp the significance of the results.

Furthermore, Python notebooks facilitate collaboration and reproducibility in the context of business reporting. Team members can easily share their work, providing opportunities for others to review, contribute, or modify the content. This collaborative environment enables organizations to leverage the collective expertise of their teams, leading to more robust and insightful analyses. Additionally, by including the data processing, analysis, and visualization code within the same document, Python notebooks ensure that the results can be easily replicated and validated by others. This transparency and reproducibility not only increase the credibility of the business report but also allow for efficient updates and adaptations as new data or requirements emerge. In summary, Python notebooks serve as a powerful medium for creating data-driven business reports, combining the analytical capabilities of programming languages with the communication benefits of a well-structured narrative.

Assessment Objectives

Analyze the Enron Mail dataset using Python in Google Colab notebooks to understand SQLite databases, data exploration, data visualization, and applying industry best practices in programming. Deliver two notebooks: a business report (which includes code, analysis, and discussion) and a development notebook which includes pseudocode, testing, and any other industry best practice not observable from the business report.

Learning Objectives

- Understand and work with SQLite databases.
- Perform data exploration and analysis using Python and relevant libraries.
- Create effective data visualizations.
- Apply industry best practices in programming, such as adding comments, creating a modular design, reusing code, and using version control with GitHub.
- Select and apply appropriate data analysis techniques.
- Interpret and communicate findings effectively.
- Demonstrate critical thinking and problem-solving skills.

The *Business Report Notebook* must run on a Google Colab instance and require no additional steps other than running code cells within the notebook.

> Note: You can have code-cells in the notebook set up the Colab instance, for example, copy data, python scripts, or other notebooks. But other than running a code cell your notebook should require no further interaction from the user/reader of the notebook.

Tasks:

Set up the environment:

- Create a new Google Colab notebook.

    - Connect the notebook to your GitHub account.
    - Import the necessary libraries (SQLite3, Pandas, Matplotlib, and ipywidgets).

- Access the database:

    - Connect to the Enron Email SQLite database using the SQLite3 library.
    - Examine the schema of the database and understand the structure of the tables.

- Data extraction and manipulation:

  - Write SQL queries to extract relevant information from the tables (e.g., sender, recipient, date, subject, etc.).
  - Use Pandas to load the query results into dataframes and perform data manipulation tasks such as filtering, grouping, and aggregation.
  - Clean and preprocess the data, addressing any missing or inconsistent values.

- Interpretation and conclusion:

  - Summarize the main insights you have gained from the data analysis.
  - Discuss any limitations of your analysis and suggest possible improvements.
  - Reflect on the usability and effectiveness of python notebooks

Enron Dataset

The publicly accessible Enron email dataset is a renowned collection of email communications exchanged between employees of the now-defunct Enron Corporation. These emails became part of the public domain during the US government's investigation into Enron's accounting fraud, and are now available for download. Researchers have extensively analyzed these emails to study various aspects such as workplace communication patterns, social networks, and other related topics. It is crucial to handle this dataset with care and respect the privacy of the individuals involved. It is essential to remember that many individuals in the dataset had no involvement in the malpractices that led to the investigation.

Database Schema

The SQLite3 database was imported from Enron email into a SQLite3 database. A Colab Notebook is provided with code to copy the database to a running Colab instance. These links are also available on the unit's Blackboard site in the Assessments section.

Table: Employeelist

```
eid: Employee-ID
firstName: First name
lastName: Last name
Email_id: Email address (primary). This one can be found in the other
tables/dataframes and is useful for matching.
Email2: Additional email address that was replace by the primary one.
Email3: See above
Email4: See above
folder: The user’s folder in the original data dump.
status: Last position of the employee. “N/A” are unknown.
```

Table: Message

```
mid: Message-ID. Refers to the rows in recipientinfo and referenceinfo.
sender: Email address (updated)
date: Date.
message_id: Internal message-ID from the mailserver.
```

```
subject: Email subject
body: Email body. Can be truncated in the R-Version!
folder: Exact folder of the e-mail inclusing subfolders.
```

Table: Recipientinfo Note: If an email is sent to multiple recipients, there is a new row for every recipient!

```
rid: Reference-ID
mid: Message-ID from the message-table/-dataframe
rtype: Shows if the receiver got the email normally (“to”), as a carbon
copy (“cc”) or a blind carbon copy (“bcc”).
rvalue: The recipient’s email address.
```

Table: Referenceinfo

```
rfid: referenceinfo-ID
mid: Message-ID
reference: Contains the whole email with shortend headers.
```

Analyses and Visualisations to Gain Insight inthe Enron's Coomunication and Organisation Structure

The analyses and visualisations below can provide insights into the communication patterns, popular topics, and organizational structure of Enron. Note that you may need to preprocess and clean the data to ensure accurate results. For this assigment you are to include any three in you submission.

Email Traffic Over Time

- Analyze the volume of emails sent over time by counting the number of messages sent per day, week, or month.
- Visualization: Create a time series line chart with the x-axis representing time and the y-axis representing the number of emails.

Top Senders and Recipients

- Identify the most frequent email senders and recipients by aggregating the data in the 'Message' and 'Recipientinfo' tables.
- Visualization: Create two horizontal bar charts, one for the top senders and the other for top recipients, with the x-axis representing the number of emails and the y-axis representing the employees.

Email Distribution by Recipient Type

- Analyze the distribution of emails by recipient type ('to', 'cc', 'bcc') using the 'rtype' column in the 'Recipientinfo' table.

- Visualization: Create a pie chart or stacked bar chart showing the proportion of emails sent to each recipient type.

Subject Keyword Analysis

- Extract keywords from email subjects in the 'Message' table and analyze their frequency to understand common topics of discussion.
- Visualization: Create a word cloud or horizontal bar chart to display the most frequently occurring keywords in email subjects.

Internal vs. External Communication

- Determine the proportion of internal communication (emails between employees) and external communication (emails between employees and external contacts) by comparing the email addresses in the 'Message' and 'Recipientinfo' tables with those in the 'Employeelist' table.
- Visualization: Create a pie chart or stacked bar chart to show the proportion of internal and external communication.

GitHub

Version control is an industry best practice technique for monitoring changes to a file or group of files over time and reverting to a previous version. For this assignment you are required to create a **new PRIVATE GitHub repository** to store the notebook and any support files. The assignment GitHub repository will contain:

- README
- Non-Conformance Report
- Notebooks required for the assignment
- Python scripts required for the assignment
- Any other relevant documents

Evaluation

As an IS Professional, you are expected to meet the specification to the best of your ability. This specification is to be treated as the output of a meeting between yourself and a client. Your instructor will take on the role of the client. If you want to implement any functionality or behaviour not described in this specification, please seek approval from the client (*your instructor*) **before** you begin writing your program.

Your submission will be assessed to see if it correctly applies the behaviours mentioned in this document. This problem specification completely describes all behaviours to be tested. You may only use programming constructs taught in the unit or demonstrated in the textbook. If you plan to use any advanced Python features not introduced in this unit, please seek approval from your instructor **before** you begin writing the program.

The code must follow the programming style naming conventions used in the PEP8, which include:

- Meaningful names for project, variable, methods, and controls.
- Correct capatlisation of variables and methods

- Appropriate use of comments
- Reference any relevant forums, websites, videos that you used.
- Use of space and indentation to program is easy to read.

Submission Guidelines

Save your Google Colab notebook(s) as an .ipynb file and push it to your GitHub repository. Write a brief README.md file describing the assignment and the purpose of the repository. Your GitHub repo shuld be private and contain all documetents relevant to this assignment.

Submit the link and zip file to your GitHub repository containing the notebook and README.md file.

This assignment is to be completed individually. **The assignment is due 23:59 Friday 26th May 2021.** The entire assignment GitHub project folder must be submitted as a single compressed archive file to the unit's BlackBoard site submission link.

Non-Conformance Report (NCR)

A non-conformance report (NCR) is a document that addresses issues where there has been a deviation from the project specification or where work fails to meet agreed quality standards. If you cannot implement some functionality or have difficulty meeting any of the requirements, you will need to provide a NCR. An example might be unable to produce the plots, or deviation from the style guide. For each non-conformance issue, you need to document:

- The problem
- Severity and impact
- How it occurred
- How to prevent it from happening again
- Plan or time estimate to fix

Grading Criteria

Your assignment will be graded based on the following criteria:

- Clarity and organization of your code (comments, modular design, code reuse).
- Proper use of version control with GitHub.
- Quality and completeness of the business report (literate programming, clear explanations, and visualizations).
- Effectiveness of the code testing notebook in identifying and resolving issues.
- Overall data analysis quality, including insights and findings based on the Enron Mail dataset.
- Critical thinking and problem solving skills

Academic Integrity

Curtin's Academic Integrity policy must be followed in all submissions. For more details, go to the Academic Integrity tab in Blackboard or the Academic Integrity website. Both submissions must adhere to the Copyright Act of 1968 as well as the 'Digital Agenda' revisions to the Copyright Act.