Institute and Faculty of Actuaries

Automating Spreadsheet Documentation with LLMs

Aniketh Pittea – Grant Thornton UK LLP

02 May 2025

Context

Excel spreadsheets play a critical role for actuaries in data management, analysis, and reporting. However, reviewing and documenting spreadsheets can be time-consuming and prone to oversight.

Challenges

- Reviewing spreadsheets can be tedious and inefficient.
- Dealing with legacy spreadsheets which lack documentation can be difficult.
- Understanding spreadsheet dependencies can be complex.
- Reproducing Excel formulas outside of spreadsheets is challenging.

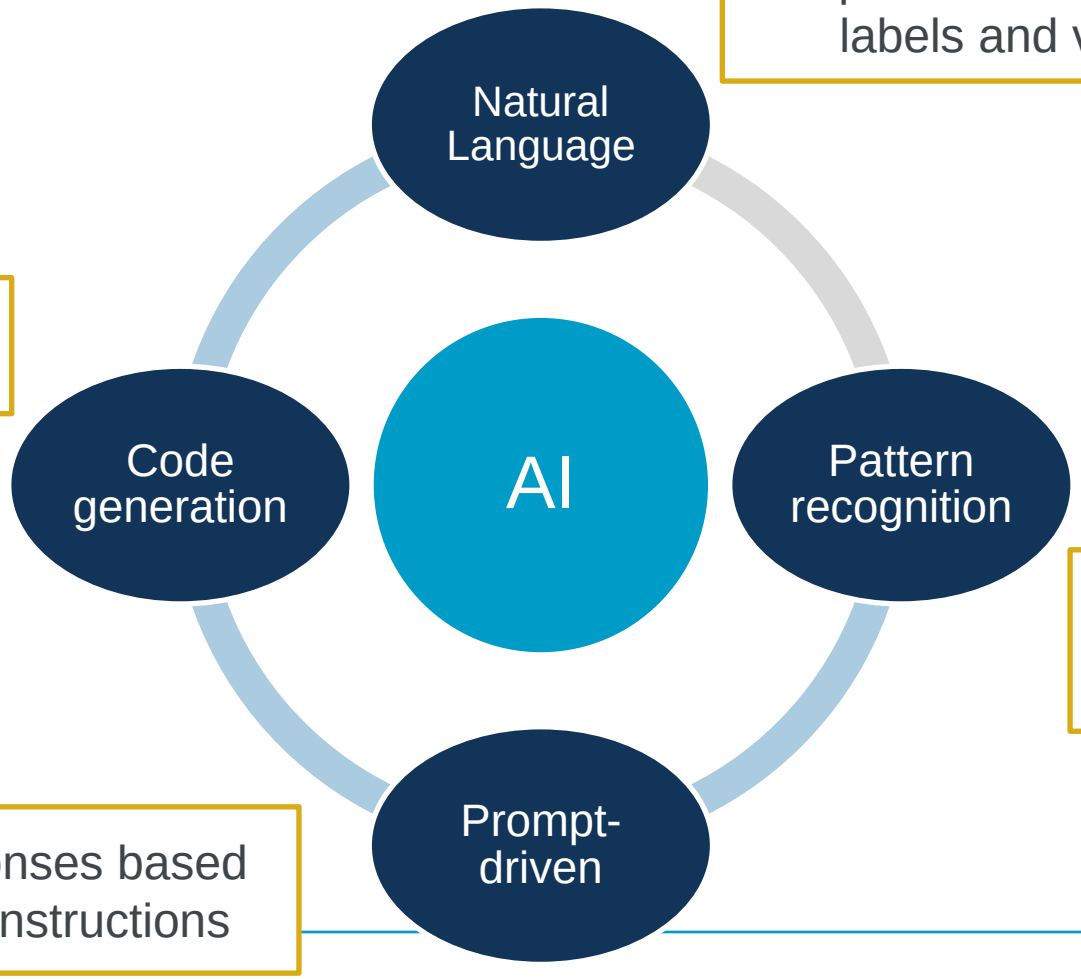
Solution

- Leverage AI to address the above challenges?

Partner Logo

Institute and Faculty of Actuaries

AI Capabilities



Interprets context from cell labels and values

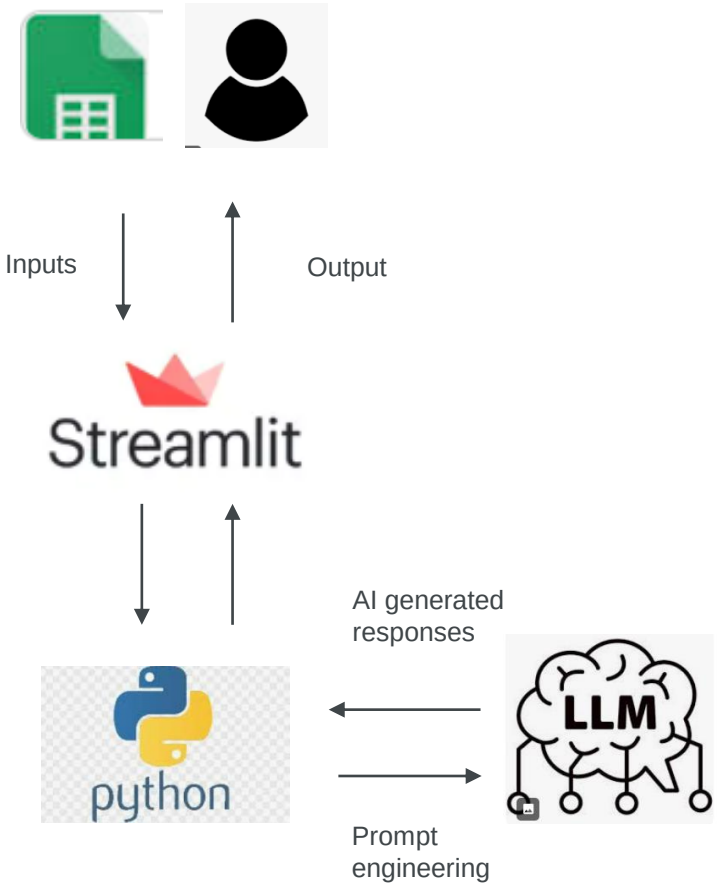
Powerful tool for coding

Identifies common structures, naming conventions and anomalies

Tailors responses based on specific instructions

Institute and Faculty of Actuaries

Proof of concept



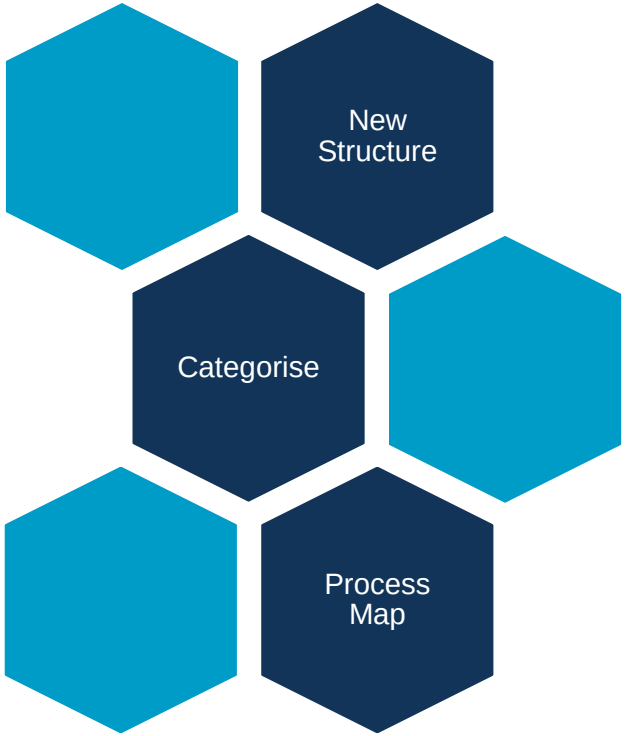
- We have built a proof of concept using Streamlit which is an open-source Python framework.
- How does the app work?
  - Simply upload your workbooks on the App.
- What does the app give you?
  - Generate a summary of each sheet using AI.
  - Identify and visualise relationships between sheets.
  - Generate Python scripts that replicate the Excel formulas.

Institute and Faculty of Actuaries

Live Demo

- Case study 1 – General model.
- Case study 2 - Tailored for model documentation.
- Case study 3 - Tailored for model formatting/EUCs.

Use Cases



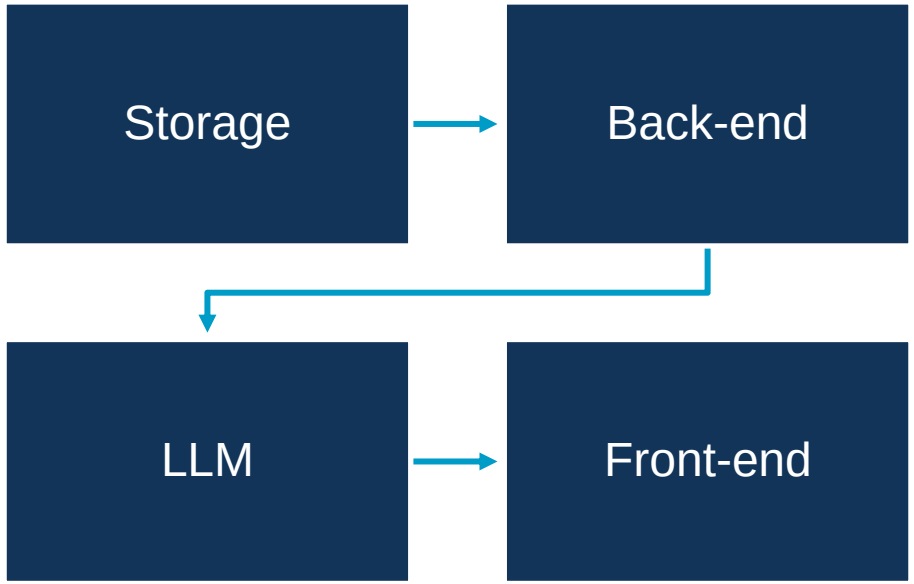
Institute and Faculty of Actuaries

Tools available

Cloud options
Microsoft Azure, AWS, OneDrive, S3, Google Drive, SharePoint

On premise options
Fully on-premise deployment on internal servers or virtual machines

Option to run a self-hosted LLM or integrate with a private API



Python
Pandas, Openpyxl, xlrd, pyxlsb

Node.js
Useful for building APIs or chatbot interfaces

.NET (C#)
Common in actuarial environments using Microsoft ecosystems

Can use Microsoft Graph for Excel file interaction

LLM integration
OpenAI API (GPT-4/GPT3.5)

Azure OpenAI (for enterprise deployments)

Self-hosted LLMs (e.g., LLaMA, Mistral, Claude via API)

Streamlit (Prototype stage)
Great for proof-of-concept apps
Rapid UI development in pure Python

Dash (Plotly)
More customizable, enterprise-ready
Better suited for production dashboards

React (JavaScript)
Highly customisable modern frontend
Can be used with chatbot UIs or internal portals

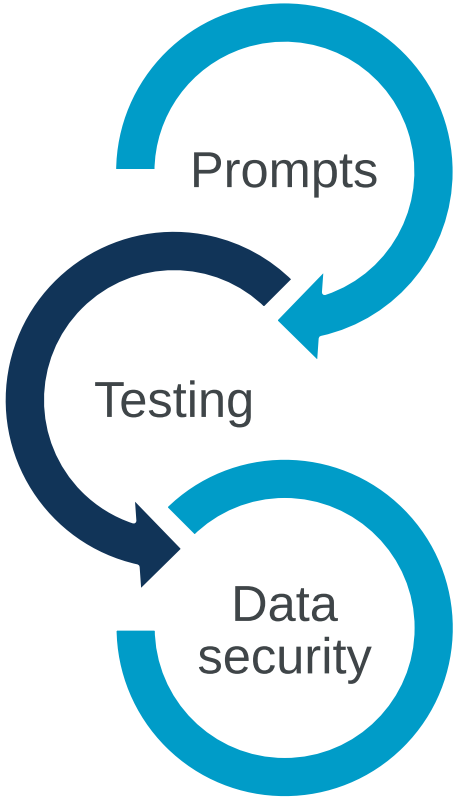
Shiny (Python or R)
Ideal for analysts familiar with R or data science dashboards

Things to bear in mind

Crafting clear, specific prompts is essential for accurate and relevant output

Prompts should reflect context: e.g., data structure, spreadsheet logic, domain language

Iterate and refine prompts based on test results and user feedback



Compare AI-generated outputs with manual documentation to ensure consistency

Test across diverse spreadsheet formats and complexity levels

Validate generated Python code before deployment or audit use

Avoid uploading sensitive spreadsheets to public APIs in production

Use private or enterprise LLM deployments (e.g. Azure OpenAI, private GPT models)

Ensure access controls and audit logging for any hosted solution

Institute and Faculty of Actuaries

Questions

Comments

Expressions of individual views by members of the Institute and Faculty of Actuaries and its staff are encouraged.

The views expressed in this presentation are those of the presenter.

Institute and Faculty of Actuaries