**Power BI model documenter using Open AI**

**Problem Statement:**

Power BI documentation is essential for maintaining the accuracy and integrity of data models, ensuring compliance with regulations, and improving collaboration and efficiency across teams. It is a time-consuming task that requires more human intervention and is highly error prone. Our model aims to automate this process using **OpenAI's** NLP capabilities.

**Purpose:**

The purpose of our Power BI model documentation automation is to provide a **Time-saving** and **Accurate** solution for fetching pbix details. Below are the purpose that our model is fulfilling:

* Automating the Power BI model documenter using one of the most efficient GPT-3 architecture i.e. **text-davinci-003**.
* Fetched the **Measures**, **Source Information,** and model **Relationships** attributes from the Power BI report.
* Wrote back the Measures and Modification Descriptions and displayed on hovering the respective properties in updated pbit files.
* Presented the output into three directories, namely **EXCEL, JSON and Updated PBIT.**

**Input/Output Deliverables:**

* **Input:** 
  + Single File
  + Multiple Files
  + A Folder
* **Output:**
  + Excel Directory
  + JSON Directory
  + Updated PBIT Directory

**Features:**

Implemented following features in our model:

1. **Features in JSON Deliverables:**

* **DataModelSchema Generation:** Generated the datamodelschema file for the respective pbit file and stored it in JSON format.

1. **Features in EXCEL Deliverables:**
   * **Measure Sheet:**

* Measure Name
* Measure Expression
* Measure Data Type
* Measure Description
  + **Source Information Sheet:**
* Table No
* Table Name
* Table Type
* Table Source
* Original Table Name
* Table Query
* Modification
* Modification Description
  + **Relationships Sheet:**
* From Table
* From Column
* To Table
* To Column
* State
* Direction
* Cardinality

1. **Features in UPDATED PBIT Deliverables:**
   * **Dynamic Hover Description:** Made the Measures and Modifications Description to hover on each respective properties.

**Prerequisites:**

In order to use this script, one need to ensure to met the following requirements:

* A Power BI Desktop installation (version - **2.115.663.0 64-bit**).
* A valid API secret key for OpenAI's NLP capabilities.
* Python 3.6 or later installed on your machine.
* Access to the Power BI models that you want to document.

**Getting Started:**

To get started with the solution, follow these steps:

* Clone the repository to your local machine.
* Install the required dependencies listed in the requirements.txt file.
* Run the solution using the Extractor.py file.

**Step-by-Step Guide:**

* User just need to download the file, and run the Project.py, on local system.
* Then user will be provided with following three selection option
* After selection is done by user, the respective pop up will open for selecting respective option from the local system.
* Then in the backend the python script will follow the below steps to extract details:
  + Read the pbit file in json format.
  + then from the json file, using the indexing, it extracted the details like measures, relationships, and the source information.
  + then for each measure and modification, we got the description of that measures using Open AI API.
* After this part is done, then the final extracted details are being stored in three folders namely EXCEL Output. JSON Output and UPDATED pbit.

**Package Description:**

**PyPbitAutomator:**

* Created a python package for the given python script and published it on [**https://test.pypi.org/**](https://test.pypi.org/)

**Package Usage:**

* **from PyPbitAutomator import Extractor**
  + This command will install all the uninstalled required libraries used in script.
* **Extractor.api()**
  + This command will prompt user for Open API Secret Key.
* **Extractor.main()**
  + This will prompt user for input of file selection and thereafter the respective file/folder.

**Package Installation:**

pip install -i https://test.pypi.org/simple/ PyPbitAutomator==0.0.3