Developer Documentation

# Introduction :

This project rests on these three premises –

1. Systems are social actors and humans perceive them as social in their interactions
2. Systems have personality and that can be assessed and defined
3. Two entities, human – human or human – system, will perceive better relationship when their personality matches are compatible

Leads to the question – Can LLMs personality be altered to better suit the personality of their users and hence improve user experience?

This project aims to understand the default personality traits of a Large Language Model (LLM) by having the model assess its own personality through prompt engineering. The results show that the LLM consistently exhibits specific personality traits across multiple runs over 30 days.

The next part of the project explores whether LLMs can be fine-tuned to change their default personalities. To do this, a dataset with over a million personality assessments is used. The dataset is broken down into subsets based on high or low scores in various personality traits. These subsets are then used to train base LLM models and create fine-tuned models. The outcome indicates that these fine-tuned models indeed show different personality traits based on their training data.

Moving forward, the project aims to make these models user-friendly. Users will be able to set their desired personality traits, and the system will use the fine-tuned models to exhibit those traits instead of the default ones. The project uses ChatGPT Turbo 3.5 as the LLM and relies on a well-established Big Five Factor questionnaire for personality assessment. The models are trained and fine-tuned using publicly available IPIP-FFM-data-8Nov2018.

In practical terms, real-time conversations have been tested using OpenAI Playground, where the desired personality traits in responses are clearly observed.

|  |  |
| --- | --- |
| **Files** | **Description** |
| AIPersonality\data | IPIP-FFM-data-8Nov2018.xls  This dataset contains 1,015,342 questionnaire answers collected online by [Open Psychometrics](https://openpsychometrics.org/tests/IPIP-BFFM/). |
| AIPersonality\Observation | Record observations of LLM personality changes in real-time scenarios |
| AIPersonality\Questionaire | IPIP prescribed BigFive Factor Questionnaire |
| AIPersonality\Results | Store results of personality assessment, every run across all models |
| AIPersonality\Training\_Records | Training records used to finetune different models |
| AIPersonality\Prereq\_Load\_Clean\_IPIPMaster Data.py | The source data set used to train models |
| AIPersonality\Prereq\_CreateTrainingDatasets.py | Python code to split source data set to subset for high/low score in personality trait |
| AIPersonality\Prereq\_CreateFinetune\_Models.py | Python code to use the subset datasets and create finetune models |
| Pickle | To store training datasets |
| AIPersonality\BigFive\_Personality\_Assessment.py | Python code to assess personality through prompt engineering, Few shot learning using the Big Five Questionaire and recording answers |

# How to Assess personality of a LLM

1. Add your model to Dic modelToAssess. modelToAssess can be found in AIPersonality\BigFive\_Personality\_Assessment.py.
2. You will need your API Key that can be obtained from Open AI, set it to system environment variable – OPENAI\_API\_KEY\_VV
3. Install packages langchain, openpyxl, pandas and openai
4. Make sure these files are in right folders –
5. questionFile (contains 50 pesonality assessment questions) with worksheet name NewIPIP.
6. scores\_file (to store the results of personality assessment) with two sheets score\_sheet and ans\_worksheet to record scores and actual LLM answers for each run
7. Running this file multiple times will help evaluate consistency of the personality scores of the system
8. If you want to run this for multiple LLMs simply add those LLMs to the dict modelToAssess

Screen capture of the Terminal while assessing personality scores of 8 fine tune models

A screen shot of a computer

Description automatically generated

# File : AIPersonality\BigFive\_Personality\_Assessment.py - IPIP Personality Trait Analysis

**Personality Assessment of Multiple Fine-Tune Language Models**

This python file contains code to assess the personality traits of a language model using a self-assessment questionnaire. The code imports several libraries, including openai, pandas, os, datetime, openpyxl, and others.

**Functions**

* assess\_LLM\_personality(llm): This function takes in a language model as an argument and returns a dataframe containing the questionnaire answers and corresponding traits.
* score\_personality(answerdataframe): This function takes in a dataframe containing the questionnaire answers and returns a dictionary containing the scores for each personality trait.
* write\_answers\_to\_excel(answerdataframe,llm\_model): This function takes in a dataframe containing the questionnaire answers and the name of the language model, and writes the answers to an Excel file.
* write\_scores\_to\_excel\_T(personality\_scores,llm\_model): This function takes in a dictionary containing the personality scores and the name of the language model, and writes the scores to an Excel file.
* access\_and\_record(llm,model\_name): This function takes in a language model and its name, assesses its personality traits, and records the answers and scores.
* defineLLM(model): This function takes in the name or ID of a language model and returns the defined language model.
* main(): This function assesses the personality traits of multiple language models.

**Usage**

To use this code, you will need to set several variables, including the OpenAI API key, the question file and worksheet, the scores file and worksheet, fine-tune model/s that you want to assess personality for, and the system time. You can then run the main function to assess the personality traits of multiple language models.

# How to create finetune models with different personality traits

1. Download publicly available dataset from Open Psychometrics - [Big Five Personality Test (kaggle.com)](https://www.kaggle.com/datasets/tunguz/big-five-personality-test)
2. This dataset contains 1,015,342 questionnaire answers collected online by [Open Psychometrics](https://openpsychometrics.org/tests/IPIP-BFFM/)
3. Store the file in this path - data\IPIP-FFM-data-8Nov2018\data-final.csv
4. Execute this script - AIPersonality\Prereq\_Load\_Clean\_IPIPMaster Data.py
5. This script performs the following functions

* Load the dataset from the specified file path.
* Remove records with missing values.
* Calculate the frequency of each value in each row and remove rows with overused values based on a threshold.
* Calculate personality scores for each trait.
* Identify outliers using z-scores and a specified threshold.
* Remove outliers from the dataset.
* Create one pickle file for each variant – For ex – A data frame for High Agreeableness or another for Low Agreeableness, one for high Openness and another for low Openness
* The script creates 10 data frames , two for each personality traits (High and Low)
* Store the cleaned dataset with scores in a pickle file.
* Store the cleaned dataset with scores in a shelved file.

1. Execute this script - AIPersonality\Prereq\_CreateTrainingDatasets.py
   1. This script creates 500 training records for each dataframe in the format prescribed by Open AI to train a model
   2. Will create 10 training datasets, two for each personality trait (High and Low)
   3. Execute this script OR follow the steps in Open AI to create finetune models using these 10 datasets - AIPersonality\Prereq\_CreateFinetune\_Models.py
   4. List of datasets and training data created

A screen shot of a computer program

Description automatically generated

* 1. List of fine tune models as seen in OpenAI dashboard

A screenshot of a computer

Description automatically generated

# Data Analysis and Observations

1. Execute this script, after adding all the finetune models to the dict modelToAccess - AIPersonality\BigFive\_Personality\_Assessment.py
2. Results will be recorded in this file - AIPersonality\Results\IPIP\_ScoresDB.xlsx
3. Results so far observes that finetune models clearly exhibit the personalities they are finetuned for

A screenshot of a spreadsheet

Description automatically generated

1. Also, real time conversation using these models produce different results based on the personality trait they are trained for. Sample can be found here - AIPersonality\Observation. This example shows how models with standard, high and low agreeableness behave in the same conversation and setting.

A screenshot of a computer

Description automatically generated

# Conclusion :

Further work needs to be done to understand user preference while using a LLM and be able to fuse these models to bring in a desired personality trait that could significantly improve user experience at a given context.