Final Configuration Manual

Project Title: *Evaluating Emotional Support from AI Chatbots: Trust, Effectiveness, and Ethical Implications*
Programme: MSc Artificial Intelligence for Business
Student: April Phyo Aung
Institution: National College of Ireland

---

1. Introduction

This configuration manual describes how to set up, configure, and run the AI chatbot evaluation system developed for this MSc project.
The project evaluates responses from three large language model (LLM)–based chatbots (ChatGPT, Gemini, and Claude) in a mental-health support context using textual similarity metrics (BLEU, METEOR, and ROUGE) and qualitative analysis.

The manual is intended to allow another user (examiner or researcher) to reproduce the evaluation process without ambiguity.

---

2. System Overview

The system consists of:

- A Jupyter Notebook–based evaluation pipeline
- CSV files containing evaluation prompts, chatbot responses, and reference answers
- Python libraries for NLP evaluation metrics
- Manually collected chatbot outputs from ChatGPT, Gemini, and Claude

The system **does not deploy a live chatbot**. Instead, it evaluates pre-generated chatbot responses.

---

3. Hardware Requirements

- Laptop or desktop computer
- Minimum 8 GB RAM (16 GB recommended)
- Stable internet connection (for environment setup only)
- Operating System:
    - macOS, Windows, or Linux

---

4. Software Requirements

4.1 Required Software

- **Python**: Version 3.10 or later
- **Jupyter Notebook** or **JupyterLab**
- **Anaconda** (recommended for environment management)

---

5. Environment Setup

5.1 Create a Virtual Environment (Recommended)

Using Anaconda:

```
conda create -n chatbot_eval python=3.10
conda activate chatbot_eval
```

Alternatively, using venv:

```
python -m venv chatbot_eval
source chatbot_eval/bin/activate  # macOS/Linux
chatbot_eval\Scripts\activate     # Windows
```

---

6. Dependency Installation

Install the required Python libraries:

```
pip install pandas numpy nltk rouge-score sacrebleu matplotlib scikit-learn
```

Additional setup for NLTK:

```
import nltk
nltk.download('punkt')
nltk.download('wordnet')
```

---

7. Project Folder Structure

Recommended structure:

```
AI_Chatbot_Thesis/
│
├── notebooks/
│   └── evaluation_metrics.ipynb
│
├── data/
│   ├── eval_promptsfrom3ai.csv
│
```

```
├── outputs/
│   ├── metric_results.csv
│   ├── charts/
│
└── README.md
```

---

8. Dataset Description

8.1 Evaluation Dataset

The primary dataset is a CSV file containing:

- User mental-health–related questions
- ChatGPT responses
- Gemini responses
- Claude responses
- Reference answers created for evaluation

File: `eval_promptsfrom3ai.csv`

8.2 Columns Example

Column Name	Description
user_question	Mental health–related prompt
Chatgpt_Answer	Response generated by ChatGPT
Gemani_Answer	Response generated by Gemini
Claude_Answer	Response generated by Claude
Reference_number	Reference answer used for evaluation

---

9. Running the Evaluation Notebook

9.1 Launch Jupyter Notebook

```
jupyter notebook
```

Open:

```
notebooks/evaluation_metrics.ipynb
```

9.2 Execution Order

Run the notebook **top to bottom** in this order:

1. Import required libraries
2. Load the CSV dataset

3. Preprocess text (handle missing values)
4. Compute BLEU scores
5. Compute METEOR scores
6. Compute ROUGE-L scores
7. Aggregate and visualise results

---

10. Evaluation Metrics Configuration

10.1 BLEU

- Used for n-gram overlap
- Implemented using `sacrebleu`
- Suitable for relative comparison only

10.2 METEOR

- Accounts for synonym matching
- Implemented using `nltk.translate.meteor_score`
- More tolerant of paraphrasing

10.3 ROUGE-L

- Measures longest common subsequence
- Implemented using `rouge-score`
- Useful for structural similarity

---

11. Output Files

The system produces:

- Tabular metric results (CSV)
- Bar charts comparing model performance
- Summary statistics for analysis

All outputs are saved in the `/outputs` directory.

---

12. Known Limitations

- Evaluation metrics measure **textual similarity**, not emotional appropriateness
- Reference answers influence metric scores
- No real-time chatbot API calls are performed

- Qualitative interpretation is required alongside numerical results

---

13. Ethical and Safety Considerations

- No real user data was used
- All prompts are hypothetical
- Chatbots are evaluated offline
- The system does not provide live mental-health advice

---

14. Troubleshooting

Common Issues

Issue	Solution
ModuleNotFoundError	Install missing library via pip
KeyError in CSV	Check column names
Low metric scores	Expected for open-ended responses
Kernel crashes	Restart kernel and rerun all cells

---

15. Conclusion

This configuration manual provides all necessary steps to reproduce the chatbot evaluation system used in this MSc project. Following the outlined setup, dataset structure, and execution steps ensures consistent results and transparency for academic assessment.