National College of Ireland

Configuration Manual

MSc Research Project
MSc Artificial Intelligence

MOHAMMED NIHAL THONIKKARA
Student ID: 23350750

School of Computing
National College of Ireland

Supervisor: Abdul Razzaq

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	MOHAMMED NIHAL THONIKKARA		
Student ID:	23350750		
Programme:	MSc Artificial Intelligence	**Year:**	2025
Module:	Practicum Part 2		
Lecturer:	Abdul Razzaq		
Submission Due Date:	11-12-2025		
Project Title:	Emotion-Aware AI Meal Recommendation with Calorie-Constrained Recipe Scoring and Dual-Stage Deep Learning Models		
Word Count:	1382 **Page Count:** 6		

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Nihal

Date: 11-12-2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

MOHAMMED NIHAL THONIKKARA
Student ID: 23350750

1 Introduction

The documentation contained in this guide will provide you with all of the information necessary to set up, configure and run all of the software aspects related to this AI Meal Recommender. In particular, it will provide a reference for anyone who is looking to re-create the experimental conditions, retrain any of the models and/or execute the recommendation engine. There will also be a section on system specifications, installation instructions and directories/structure of the project, as well as a discussion of key configuration settings for the implementation.

2 System Requirements

To ensure successful execution of the project's code, the following hardware and software specifications are recommended.

2.1 Hardware Requirements

- **CPU:** Multi-core processor (e.g., Intel Core i5/AMD Ryzen 5 or higher).
- **Memory (RAM):** 16 GB minimum, 32 GB recommended for handling larger datasets and model training.
- **GPU:** An NVIDIA GPU with CUDA support is highly recommended for training the deep learning models (e.g., NVIDIA GeForce RTX 2060 or higher with at least 6 GB of VRAM). The project can run on a CPU, but training times will be significantly longer.
- **Storage:** 10 GB of free disk space for datasets, libraries, and saved model artifacts.

2.2 Software Requirements

- **Operating System:** Windows 10/11, macOS 11.0+, or a modern Linux distribution (e.g., Ubuntu 20.04+).
- **Programming Language:** Python version 3.8 or higher.
- **Package Manager:** pip (Python Package Installer).
- **Python Libraries:** The primary dependencies are listed below. It is strongly recommended to use the provided requirements.txt file for a seamless installation.

```
# requirements.txt
pandas==2.1.4
numpy==1.26.2
tensorflow==2.15.0
scikit-learn==1.3.2
matplotlib==3.8.2
seaborn==0.13.0
nltk==3.8.1
```

3 Installation and Setup

Follow these steps to configure the project environment.

1. **Clone the Project Repository:** Obtain the project source code by cloning the repository from its source location using Git.

```
git clone <repository-url>
cd <project-directory>
```

2. **Create a Virtual Environment:** It is best practice to create a dedicated virtual environment to avoid conflicts with other Python projects.

```
python -m venv venv
```

Activate the environment:

   - On Windows: venv\Scripts\activate
   - On macOS/Linux: source venv/bin/activate

3. **Install Dependencies:** Install all required Python libraries using the requirements.txt file.

```
pip install -r requirements.txt
```

4. **Download NLTK Data:** The project requires specific data packages from the Natural Language Toolkit (NLTK). Run the following commands in a Python interpreter to download them:

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

5. **Place Datasets:** Download the required datasets from the Research paper and Food.com Recipes and Interactions (emotion_sentimen_dataset.csv, RAW_recipes.csv, RAW_interactions.csv) and place them in the data/ directory within the project's root folder.

4 Project Structure

The project is organized into the following directory structure to maintain clarity and separation of concerns.

```
.
├── data/
│   ├── emotion_sentimen_dataset.csv
│   ├── RAW_recipes.csv
│   └── RAW_interactions.csv
│
├── models/
│   ├── emotion_bilstm_model.h5
│   ├── tokenizer.pkl
│   ├── label_encoder.pkl
│   └── rating_predictor_model.h5
│
├── notebooks/
│   └── Main_Implementation.ipynb
│
├── src/
│   └── (Optional: Refactored Python scripts)
│
├── requirements.txt
└── README.md
```

- **data/**: Contains all the raw data files required for the project.
- **models/**: The designated location for storing trained model files (.h5) and other serialized artifacts like tokenizers (.pkl).

- **notebooks/**: Contains the Jupyter/Colab notebook with the end-to-end implementation.
- **src/**: An optional directory for any standalone Python modules if the code is refactored out of the notebook.
- **requirements.txt**: Lists all Python dependencies for easy installation.

5 Execution Guide

The primary implementation is contained within the Main_Implementation.ipynb notebook. The execution flow is divided into two main stages.

5.1 Stage 1: Training the Emotion Classification Model

1. Open the notebooks/Main_Implementation.ipynb file in a Jupyter Notebook or Google Colab environment.
2. Ensure the kernel is using the created virtual environment.
3. Execute the cells sequentially from the top, starting with library imports and data loading.
4. Proceed through the data analysis, preprocessing, and balancing steps for the emotion dataset.
5. Run the cells that define, compile, and train the Bi-LSTM model for emotion classification.
6. Upon completion of training, execute the final cells in this section to save the trained model (emotion_bilstm_model.h5), tokenizer (tokenizer.pkl), and label encoder (label_encoder.pkl) to the models/ directory.

5.2 Stage 2: Training the Rating Prediction Model and Running Recommendations

1. Continue executing cells within the same notebook. This stage depends on the artifacts saved in Stage 1.
2. Run the cells that load the recipe and interaction datasets, perform merging, and create the balanced 4,000-sample dataset.
3. Execute the feature engineering cells. These will load the saved emotion model from Stage 1 and apply it to the recipe reviews to create the 'emotion' feature.
4. Run the cells that prepare the multi-modal input tensor by combining TF-IDF vectors, normalized nutrition data, and one-hot encoded emotion labels.
5. Execute the cells to define, compile, and train the attention-based Bi-LSTM model for rating prediction. It is recommended to save the trained model to models/rating_predictor_model.h5.
6. The final cells in the notebook contain the recommend_recipes function. Execute this cell to define the function.
7. The recommender can now be used by calling this function with desired parameters, for example:

```
# Get the top 2 recommendations for recipes associated with the 'love' emotion
recommend_recipes('love', top_n=2)

# Get the top 2 recommendations for 'neutral' emotion with a low calorie filter
recommend_recipes('neutral', nutrition_filter={'calories': (0, 0.4)}, top_n=2)
```

6 Key Configuration Parameters

Several key parameters within the code can be adjusted to modify the behavior of the models and the data processing pipeline. These are primarily located in the Main_Implementation.ipynb notebook.

Parameter	Location/Cell	Default Value	Description
Emotion Sample Size	Data Analysis	80000	The number of samples to randomly draw from the emotion dataset.
Max Neutral Samples	Data Analysis	3500	The target size for the 'neutral' class during undersampling to balance the dataset.
Max Vocabulary Size	Tokenizer	60000	The maximum number of unique words to consider in the emotion model's tokenizer.
Embedding Dimension	Model Building	100	The size of the word embedding vectors in the emotion classification model.
LSTM Units	Model Building	128, 64	The number of units in the first and second Bi-LSTM layers of the emotion model.
Dropout Rate	Model Building	0.5	The dropout rate used for regularization in the emotion classification model.
Training Epochs	Model fit call	30 (Emotion), 50 (Rating)	The maximum number of training epochs for each model.
Batch Size	Model fit call	64	The number of samples per gradient update during training.
Early Stopping Patience	EarlyStopping callback	3	The number of epochs with no improvement after which training will be stopped.
Recipe Sample Size	Data Preparation	4000	The total number of recipes to include in the final balanced dataset for the rating predictor.
TF-IDF Max Features	TfidfVectorizer	300	The maximum number of features (words) to create from the recipe text data.
Text Sequence Length	pad_sequences	100	The fixed length to which the TF-IDF vectors are repeated for the rating predictor's input.