National College of Ireland

Configuration Manual

MSc Research Project
Data Analytics

Vishal Gajanan Patwardhan
Student ID: X18190839

School of Computing
National College of Ireland

Supervisor: Jorge Basilio

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	VISHAL GAJANAN PATWARDHAN
Student ID:	X18190839
Programme:	MSC DATA ANALYTICS **Year:** 2021-2022
Module:	RESEARCH PROJECT
Lecturer:	JORGE BASILIO
Submission Due Date:	15TH AUGUST 2022
Project Title:	SPEECH TO VISUALIZATION USING TRANSFORMER BASED DEEP LEARNING MODEL
Word Count:	**1083 Page Count: 5**

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: VISHAL GAJANAN PATWARDHAN

Date: 15TH AUGUST 2022

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

Vishal Gajanan Patwardhan
Student ID: x18190839

1 Introduction

This configuration manual document explains all the necessary steps which needs to follow to make this research project run on the machine. This document has covered all the configuration details and its purpose in the research project, Speech to Visualization using transformer based deep learning model.

2 System Configuration

2.1 Hardware Requirements

Implementation of the model and prediction of the models are done on the Visual Studio Code but due to less computation power in the local computer, model was trained on the Google Colab as it provides free GPU power for the high computational task for 12 hours continuously. Once the model is trained on the Google Colab, it is then saved and downloaded on the local machine for prediction and evaluation of the model performance. Followings are the hardware requirements needed for an ideal machine to supports this research project:

- Operating System: macOS Monterey / Windows 10 x64 (compatible)
- Processor: Apple M1 chip / Intel Core i5-9300H @ 2.4 GHz (compatible)
- RAM: 8 GB
- Hard drive: min 128 GB SSD

2.2 Software Requirements

This research thesis implementation was done using the python language (ver. 3.8). Visual Studio Code (VS Code) is used as an Integrated Development Environment (IDE) to write the python code. The code was downloaded from the GitHub link from the previous researchers. These python files were modified and added to support this research project. This research implementation code was developed using the Pytorch and SQLite3 database is used to get the data using the generated SQL query. Followings are the software's/libraries needed to support this research project:

- py_stringsimjoin
- python-dateutil
- torchtext==0.10.0
- torch==1.9.0
- vega
- SpeechRecognition
- PyAudio

- pyttsx3
- matplotlib

3 Dataset Description

The training, testing and validation dataset were created using the publicly available nvBench dataset Luo et al. (2021). nvBench is the large-scale dataset which contains the natural language question and SQL Query pairs in a json file. Along with the json file there are the database data files and database schema files on which json file was created. Training and testing files are in the artefact folder (x18190839_Artefact/Speech_to_visualization-master /dataset/dataset_final).

4 Environment Setup

Environment Setup in the research is divided into two parts i.e., training setup and prediction setup.

4.1 Training Setup

As already mentioned, the training of the model need high computation power, so the training was done on the Google Colab environment. Follow the below steps to setup the training environment.

i. Upload the project folder "x18190839_Artefact/Speech_to_visualization-master" on the google drive.

ii. Mount the drive to the Google Colab environment by adding lines as mentioned in the Fig. 1. in the new notebook file.

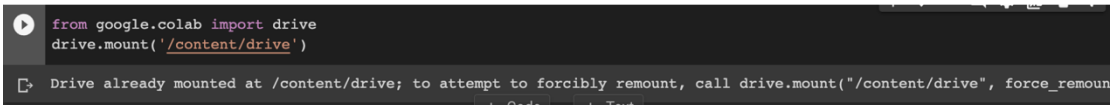

Fig. 1. Google drive mounting in Google Colab

iii. After than run the "x18190839_Artefact/Speech_to_Visualization-master/requirements.txt" file on to download all the packages and libraries needed for the project.

Fig. 2. Loading packages and libraries on Google Colab

4.2 Prediction Setup

In the Prediction setup environment, the trained model in the Google Colab is used for the predictions in the VS code IDE. Follow this step to perform the prediction on the VS code:

i. Load the “x18190839_Artefact/Speech_to_Visualization-master” folder in the VS code project directory.
ii. Run the “x18190839_Artefact/Speech_to_Visualization-master/requirements.txt” by running the code mentioned in Fig. 3. in the terminal to install all the libraries and packages which need for the project for its full functionality.

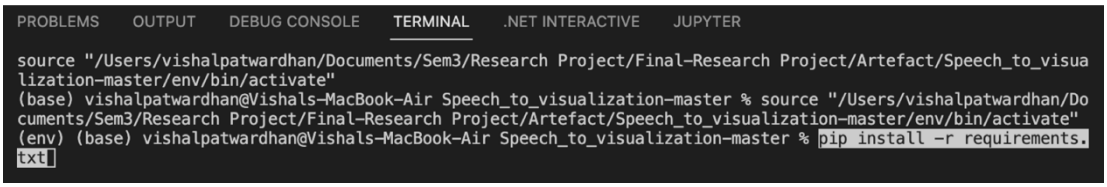

Fig. 3. Running the requirements.txt in VS code terminal.

5 Training and Evaluating the Model

As already mentioned, the training of the model need high computation power, so the training was done on the Google Colab environment. Follow the below steps to setup the training environment.

i. Make sure you have the training files and correct location of the training files are mentioned in “x18190839_Artefact/Speech_to_Visualization-master/train.py” as per Fig. 4.

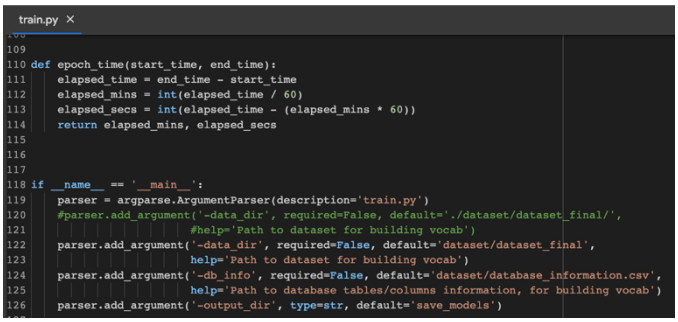

Fig. 4. Definition of the training dataset in the train.py file.

ii. Now run the “x18190839_Artefact/train.py” as by adding the mentioned line in the Fig. 5. To start the training of the model.

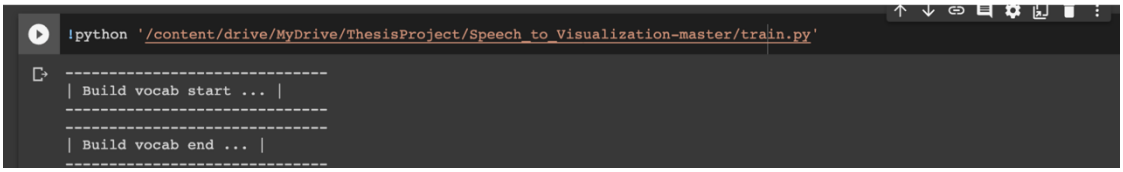

Fig. 5. Running the train.py file.

iii. After the training is done, the best model file is saved in “x18190839_Artefact/Speech_to_Visualization-master/save_models” path.

iv. Now to perform the testing, run the file "x18190839_Artefact/Speech_to_Visualization-master/test.py", to check the accuracy of the training.

Fig. 6. Running the test.py file.

v. If the accuracy is good enough you can download the best model file from this folder "x18190839_Artefact/Speech_to_Visualization-master/save_models" into your local machine for predictions.

6 Drawing Visualizations

In the Prediction setup environment, the trained model in the Google Colab is used for the predictions in the VS code IDE. Follow this step to perform the prediction on the VS code:

i. Copy and paste the downloaded best trained model from the Google Colab into the "x18190839_Artefact/Speech_to_visualization-master/save_models" folder.

ii. Open the "x18190839_Artefact/Speech_to_visualization-master/start.py" file and specify the trained model file path as mentioned in Fig. 7.

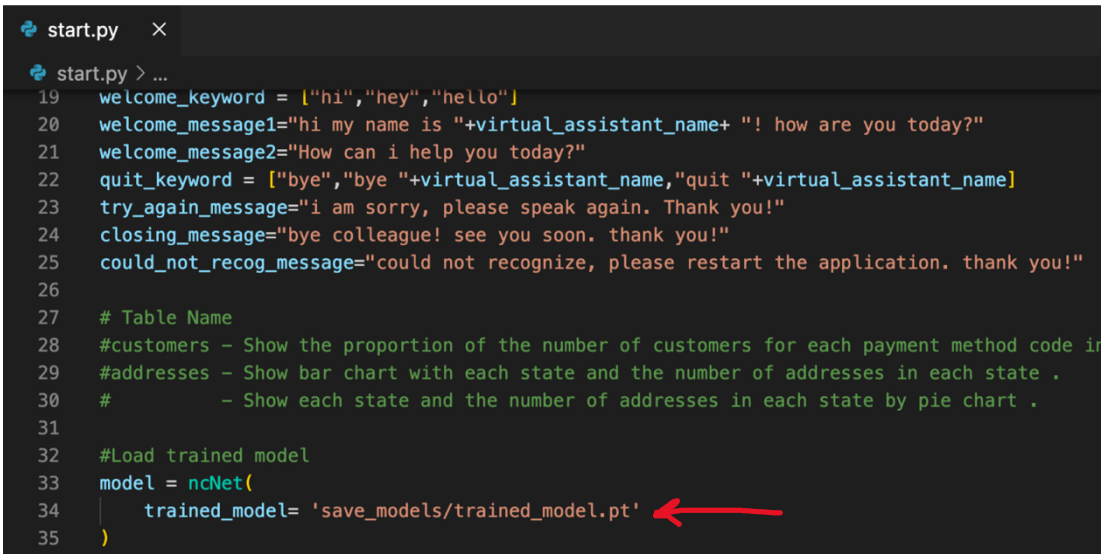

Fig. 7. Specifying the trained model path in start.py file.

iii. Also specify the database file and table name on which the visualization needs to be performed. Refer Fig. 8.

Fig. 8. Specifying the database name and table name.

iv. To start the execution run the file on the terminal by running the below command.

```
python start.py
```

v. Once the file is successfully executed, an automatic virtual assistant called "Sarah" will ask "Hi my name is Sarah, how are you today?" and the same can be read on the terminal. Once Sarah is done with speaking, the terminal will ask user to speak by displaying "user speaking~".

vi. User needs to greet the system by responding to the virtual assistant as "Hi Sarah, I am good". Then next Sarah will ask "How can I help you today?".

vii. After this question once you see the display message as "user speaking~" you can start asking the natural language question with regards to the table name mentioned in the start.py file with the desired chart.

viii. Then Sarah will confirm with you by spelling the question to you and will simultaneously ask you to confirm if she got your question correct or not. For. example "show the proportion of the number of customers for each payment method code in a pie chart, did you mean this? Please say yes or no".

ix. User needs to say to proceed further with the natural language. If user says yes, at this instance natural language question is passed to the model to predict the SQL query and then draws the visualization.

References

Luo, Y., Tang, N., Li, G., Tang, J., Chai, C. and Qin, X., 2021. Natural Language to visualization by neural machine translation. *IEEE Transactions on Visualization and Computer Graphics*, *28*(1), pp.217-226.