**Deep-Ad-Recommendation System**

The Documentation consist of details about the prototype developed for Clovitek Audio Streaming Device. Please note that the whole project is written in python with the appropriate libraries. The main purpose of this protype is to test whether the Deep Learning can help to optimize the Ad-Experience for Clovitek Customers.

**Getting Started (Detailed Information about Folders and Files)**

The prototype is on two folders: the first one is the protype and the second one is the protype-data.



The protype folder consist of the Main Library which makes the Protype work, this folder is reusable and is the back-end Library that makes the protype works. While the Prototype Data consist of Different Test Cases as how a User(in case you) can use the back-end Library. Let me further explain about the folders in **Protype Data Folder**:

1-Test-This Consist of a Python Script Named User-Test-Cases.py. The main purpose of this script is to test the functions of the Library made in Protype Folder. There are many functions in this script and the purpose of all these functions is just a way to show you how to use my back-end library. Detailed Comments are present in this script. Please also note that user-test-cases was tested on my machine therefore you might need to make some change (for example changing the directory). However, remember user-test-cases is just for testing, it does not have anything to do with final working of the protype (just way to check my library).

2-Audio-Mp4Scrapped Data-Test Data Scrapped from YouTube using AdAudioScrapper (back-end library). Please note that you yourself must mention and decide upon the categories of ads that you would like to install. I could have hard encoded it but to make the library more robust, I have left it so that you can implement the AdAudioScrapper according to your needs.

3- Audio-FlaccData-Test Data in Flacc Format. After downloading the YouTube audio, I used FlaccConverter Method of Back-End Library (SpeechToText.Py) to convert all the Mp4 Audio to Flacc (In user test cases.py Flacc\_Converter\_Test shows one way to doing it).

4-Data-This Folder consist of two npy file(Final\_Array and labels) and one Model file(word2vec.model). The npy files are the final input to my neural network and the model file is the word embedding I created from Audio Data (Don’t worry I will explain in detail about this).

5-There are number of folders named like Graphs-Performance-Different Learning-Rates-Daterun\_2019\_09\_15-00\_18\_55, and these folders consist information about the performance of my neural network as I test it at different times and optimized it.

6-my\_logs-This is just serving the same purpose as above (used for Tensor board).

7-Proposed-Model-The Report Submitted to you Earlier

8-Validation-This folder is recording of my own Audio and testing how well my neural network can classify the previously unseen new Audio. The function predict\_new\_audio in User\_test\_cases.py is taking this validation data and making use of my neural Network to see how well it performs (please see it so that you can see one way to test my neural network).

9-Customer-Pattern.py-The Mean Shift Algorithm to detect the Hidden pattern in your customer data (provided by in form of excel file named Clovitek.xlsx). Run this script and a visualisation would tell about the pattern of your customers.

10-my\_keras\_model.h5-Hadoop file consist of Neural Network.

Now please note that I have only mentioned about the purposes of every folder and file in **protype-data** and now I am going to explain the structure of **protype folder** (The main reusable library):

1-Src-The Folder consist of All the Python Script Used in Making the final model work. The first is **AdAudioScrapper.py**-the main purpose of this Script is to scrap audio from YouTube to create data on which my neural network can train. In order to use it you would have to decide your categories and then use the script. The Script can install only one company Ad at a time. So, for example if you want to install KFC ads, you would give its name and Scrapper would install. And in order to install multiple categories, you would have to optimize your implementation as I wanted to make it more reusable and robust. The second script in this folder is **AdRecommendation.py-**This is the Main Neural Network or the main part of the protype which makes the recommendation. Please note since this is a protype, the architecture could be changed as your company move forward or if we find better hyperparameters. **SpeechToText.py-**The main objective of this script is twofold. Firstly it is use to convert MP4 Audio to Flacc format and secondly it uses IBM Watson to convert Speech to Text and then reformat it in an Appropriate Array (FinalArray.npy). This array is then converted to numerical values either by Count Vectorizer or TD-IDF or Word Embedding (Please note that all these are completely different approaches and all of them performed quite similar therefore I included them all) . The way I used these approaches to covert Text to numerical Values which is then Fed to Neural Network is once again shown in User-Test-Cases.py (Please see word\_embedding & preprocessing\_pipeline functions). There are detailed comments for each of this function.

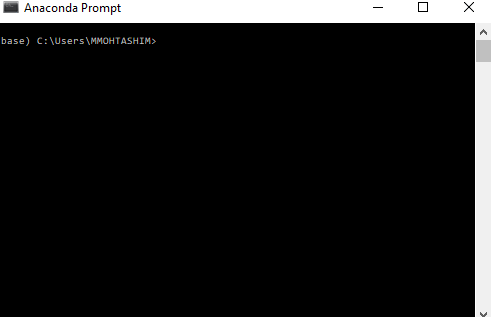
The rest of the folders in protype folder are not important and are just structured according to a pip package.

**Prerequisites**

What things you need to install the software and how to install them

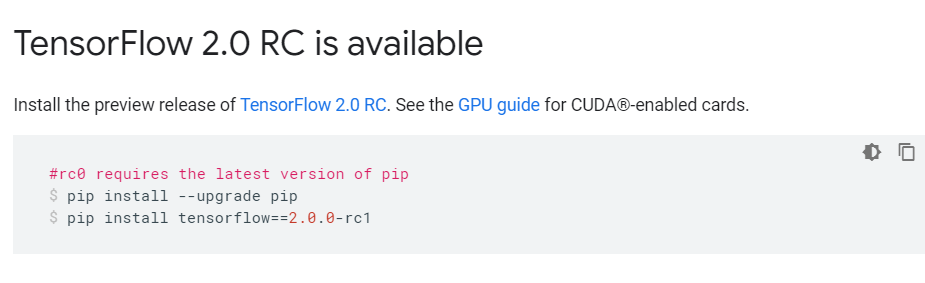
**1-Anaconda:**

Before Installing the Libraries, you need to install the Anaconda Python Working Environment. Go to this link: <https://www.anaconda.com/distribution/> and then Download the python 3.7 version. After this in window search for: Anaconda Prompt and open the Anaconda Prompt, this will lead you to the following screen:



Now on this screen type in below commands to install all of the pip packages.

**2-You need to install the following Libraries (Latest Versions):**

TensorFlow-

Sklearn-pip install scikit-learn

Pandas-pip install pandas

Matplotlib-pip install matplotlib

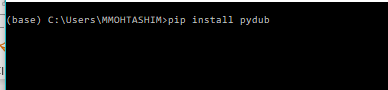
Genism-pip install genism

Numpy-pip install numpy

SpeechRecognition-pip install SpeechRecognition

Pydub-pip install pydub

IBM-Watson-pip install ibm-watson

-An example

**3-FFmpeg:**

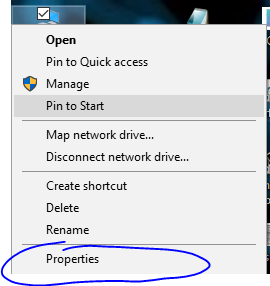
FFmpeg is the leading multimedia framework to decode, encode, transcode, mux, demux, stream, filter and play. All builds require at least Windows 7 or Mac OS X 10.10. Nightly git builds are licensed as GPL 3.0, and release build are licensed as GPL 3.0 and LGPL 3.0. LGPL 3.0 release builds can be found using the "All Builds" links. In order to install this go to the following link: <https://ffmpeg.zeranoe.com/builds/> and install the Build according to your need. Then follow these steps:

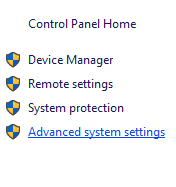
1-You will get a zip file 

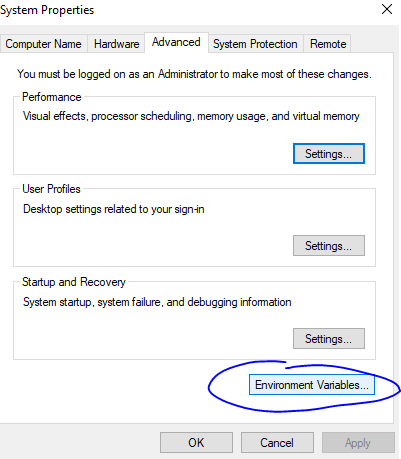
Unzip it and then you will have following folder:

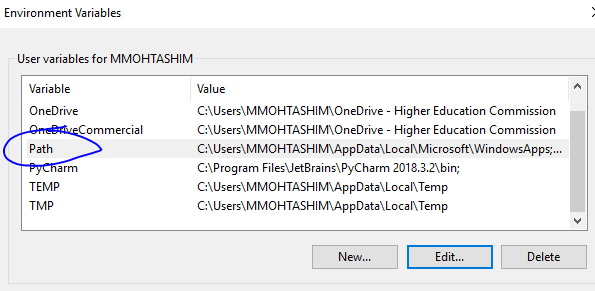


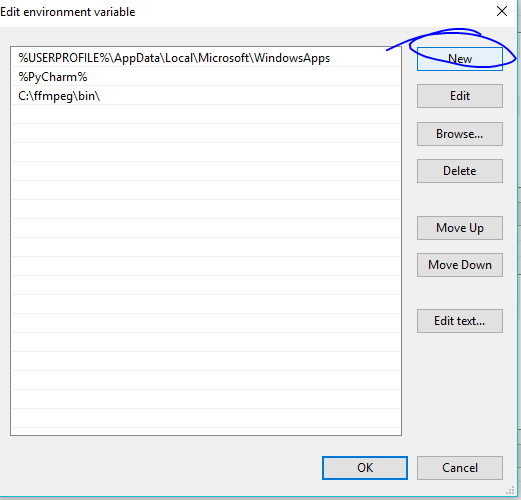
Rename the file to ffmpeg and copy the folder to the root directory of C drive.

2--Click on properties

3--In Properties click Advanced System Settings

4--Click Environment Variables

5--Click Path

6--click New

7-Type in C:\ffmpeg\bin and press OK

8-Reboot your Machine

**Running the tests**

Explain how to run the automated tests for this system

Now you have all the Dependencies installed for protype, now go to user-test-case.py. There are detailed comments for each of the function and see how to interact with my Library. As mentioned before User-Test-Case.py is just for your help and guidance therefore shows one way to interact with my Library.

**Authors**

* **Mohammad Mohtashim Khan**

**License**

This project is licensed under the MIT License - see the [LICENSE.md](https://gist.github.com/PurpleBooth/LICENSE.md) file for details and is property of Clovitek.