Telegram Bot to Detect Deepface Audio Fakes

**Frameworks**

* pyTelegramBotAPI
* TensorFlow 2.8.0
* librosa
* pydub
* ffmpeg
* torch
* numpy 1.21.5
* matplotlib

**Commands**

|  |  |
| --- | --- |
| **"/start" - Welcome message and brief instructions for using the bot** | **C:\Users\NitghtWay\AppData\Local\Microsoft\Windows\INetCache\Content.Word\68747470733a2f2f73756e392d35332e757365726170692e636f6d2f696d70662f41587a4f7048473863464c44623849412d485648394343374744674b745873705634724f6c512f4758715f385133386569342e6a70673f73697** |
| **"/help" is a command if the user does not understand how to work with the bot** | **C:\Users\NitghtWay\AppData\Local\Microsoft\Windows\INetCache\Content.Word\68747470733a2f2f73756e392d36342e757365726170692e636f6d2f696d70662f5750686d793333456f56497173313937563237455033343051436e483171427943614f6878512f7a6a616e73495f495335632e6a70673f73697** |
| **"/info" - For more information** | **C:\Users\NitghtWay\AppData\Local\Microsoft\Windows\INetCache\Content.Word\68747470733a2f2f73756e392d36342e757365726170692e636f6d2f696d70662f706e3632646e6b2d5f636337585764556b4668334154595f4d62573972735a614f64503465512f5f4a3742527636587267382e6a70673f73697** |
| **If the entered text is not one of the above commands, the bot will issue a warning message** | **C:\Users\NitghtWay\AppData\Local\Microsoft\Windows\INetCache\Content.Word\68747470733a2f2f73756e392d32392e757365726170692e636f6d2f696d70662f67442d314746526e344d565f70353366474145785568594a2d38486f69705154554b515542412f4650343348426c6b695a6b2e6a70673f73697** |

**Evaluation of an audio file**

|  |  |
| --- | --- |
| **To analyze your audio file, it is enough to send a .mp3 or .wav, then the bot will save your audio recording and respond:** | |
|  | **C:\Users\NitghtWay\AppData\Local\Microsoft\Windows\INetCache\Content.Word\68747470733a2f2f73756e392d38342e757365726170692e636f6d2f696d70662f6b66596d4a67464a4151355a546c57476c755f58485459724f35304a55517748794b484571412f76672d496b74306b6848512e6a70673f73697** |
| **Real** | **Fake** |

**The principle of operation**

**To analyze an audio file , the bot uses a neural network of the format .h5 Neural Network for Detecting Deepface Audio Recordings - Neural Network Training Repository. Since the neural network was trained on audio recordings lasting 2 seconds, it can analyze files only of this duration, but if the bot receives an audio file lasting longer than two seconds, it will divide it into early segments of 2 seconds.**

|  |  |
| --- | --- |
| **A class for cutting audio files, as well as calculating their duration in seconds** |  |
| **After splitting the file (if necessary) The bot converts each fragment into a mel spectrogram**  **Function for converting an audio file into a mel spectrogram:** |  |
| **To directly evaluate the file and predict its authenticity or not, the following function is used, which takes the weights of the model as input, as well as the absolute path to the audio recording and the folder for storing the spectrogram, the function returns 0 or 1 if the audio recording is not forged and if forged, respectively** |  |
| **Since the fibrosa library, which is used to convert audio recordings into a mel spectrogram, works only with wav files, the function of converting them to wav format is described for converting mp3 files** |  |
| **Since the bot saves all the received audio recordings in a folder for temporary files, in order not to overflow the memory, a function has been created to clear the folder of mel spectrogram image files** |  |

**Project structure**

* **File main.py this is the main algorithm of the bot**
* **File functions.py auxiliary file for a separate description of all functions used**
* **The config file.py is a configuration file that contains a bot token, as well as a dictionary with responses to messages and paths for getting model weights and folders with temporary files**
* **The 'temp' folder is a folder intended for temporary files**
* **'model' folder is a folder for storing neural network weights**