Milestone 2 Draft of Final Report – 10 pts Due to D2L 11/28/2022 @ 11:59 pm

Updated during teams meeting Sunday Nov 6 and in class Tues Nov 8

Delete previous versions emailed out

Note: If you re-upload your files to D2L, you must re-upload ALL files as the system keeps the most recent uploaded submission only. No zip files!

GENERAL SUBMISSION REQUIREMENTS

Upload all files individually as specified, not as zip files, to Assignments in D2L. Do not email files.

FOR M2 SUBMIT - DRAFT Final Report -

1. **Cover Page – standard stuff, website, collaboration**
2. **Abstract / Overview / Executive Summary**
3. Table of Contents
4. Background information -
5. Requirements / Analysis / Results / Development – SDLC - depends on the type of project
    This is the body of your project report – all the effort, decisions, streams of consciousness:
    Requirements derived from working with Team and Project Owner and Faculty Sponsors
        Include mockups of screens along with description of tool used to build mockup
    Analysis of tech platforms you reviewed before selecting the one to use for your project
    Tech platform – describe in detail the tools you are using – use TECHNICAL TERMS
        Include high level architecture drawings
    Development of software project
        Discuss your development effort – what went well, roadblocks, work arounds
        Include detailed architecture drawings, data models, etc (partly done)

Results – where are you with this software
    Research
    Mobile app –
    Web application, desktop application

Project Planning and Management
    Discuss collaboration tools and project meeting schedule
    Revisit your project plan and include updated Gantt Charts – show you can manage projects (don't need this)

Version Control

Test Plan and Test Report – recall your list of requirements becomes a test plan

Summary / Conclusion

APPENDIX – customize for your report

Save docs as file(s) named - ProjectNo-M2-DraftFinalReport.docx

Verbal Aggression Detection

CS 4850—Senior Project

Spring 2023

Professor Perry

04/04/2023

Team: SP2—Blue Data Mining and Artificial Intelligence

Website: https://edwardhachem.github.io/Senior-Project/

1.1 Overview

Abusive/toxic relationships are the root of many problems our society faces today. Whether it’s physical, emotional, mental, or verbal, we have a responsibility to our communities and society at large to protect ourselves and others from unhealthy relationships. Integrating research-backed indicators of abuse and technology is a popular and growing subset of artificial intelligence and machine learning. The team will create a web scraper to mine videos of conversations from youtube. A tool will be created in python to listen to the conversations and mark if the audio has indicators of verbal abuse or aggression. The results will be stored in a database on AWS for pattern analysis.

The goal of this project is to develop a tool that can help us detect when we or someone else is being aggressive in order to recognize and improve on toxic relationships.

1.2 Background Information

3.0 Functional Requirements

- System must collect videos from YouTube
- System must extract audio from YouTube videos
- System must preprocess audio data
  - Volume normalization
  - Noise reduction
- System must be available via Google Chrome web browser
- System must provide digital representation
- System must be trained using training data
- Training data collected from YouTube
  - Characterize aggressive Speech
- Train model
  - Model = Convolutional Neural Network

4.0 Non-Functional Requirements

4.1 Security

Access to files and data will be provided by an AWS secure S3 bucket that is limited to specific users through an IAM role. The database is also hosted on a secure EC2 instance that is limited to the same users.

4.2 Capacity

- In the process of being defined, however our current EC2 instance is capable of 200G of storage.

4.3 Usability

The program needs an active internet connection to run.

Development of software project

Our development process was one of trial and error, with a plethora of data engineering and neural network tuning. During the development process the first issue we ran into was converting the MP4 data to a form that was usable to a Convolutional neural network. Our solution was to first convert the videos into WAV format, and from there use “Librosa” to convert the WAV files to Mel-Spectrogram.

Joyre discusses web scraper

The use of the Mel-Spectrogram is that it allows the neural network to view video data as an image, which it can then detect patterns based on the label assigned to the image (supervised learning). We have 3 classes that take care of this in the project's source code. The “get_ffmpeg” class is responsible for converting MP4-> WAV, “get_mel” is responsible for converting WAV -> Mel-Spectrogram, and “automate_conversion” class is responsible for outputting a csv and orchestrating the other two classes.

Quinn Discusses Neural network.

Besides mining data, the live audio recorder can also be used to detect audio from the user's microphone. That audio is split into segments of four seconds each. These audio segments are converted to wav files and are converted to spectrograms which are then sent into the CNN for results processing.

Results

Currently, we have a web scraper to mine videos off of youtube, a live audio recorder, The data pipeline for converting audio to mel spectrogram at scale, and the CNN which processes the formatted data.

Project Planning and Management

Our team has been using zoom to conduct meetings on a semiweekly basis. Our team meets every Monday, Wednesday, and sometimes on Saturdays. We have been using github for version control with each person working on their own branch and creating pull requests to update the main branch.