Sentiment Analysis on Twitter Videos using Facial Emotion Recognition

By: Travis Baylor, Nathan Schaper, Hiep Duong

Mentor: Athina Bikaki

Contents

1. Introduction………………………………………………………………………………..2
2. Dataset Description………………………………………………………………………..2
3. Methodology………………………………………………………………………………3
4. Results……………………………………………………………………………………..4
5. Conclusion………….……………………………………………………………………..5
6. Contributions……………………………………………………………………………...6
7. Works Cited….……………………………………………………………………………7

Introduction

Throughout the twenty-first century, the Internet has become the prominent way to share data around the world; specifically, it has shown to be an easy way of sharing video data on platforms such as Twitter. In fact, in 2021, Internet video traffic accounted for 80% of all global Internet traffic (Cisco 3). With all this video data being shared, there is a significant need to find ways to manage and categorize this data in a useful manner. These are especially needed because, compared to other forms of data such as text, it is generally more difficult to automatically extract summarizing data from video using computer vision methods.

One method of categorizing this data is by the emotions portrayed in the video. Using the FER (Facial Expression Recognition) Python library developed by Justin Shenk, we have created a way to organize Twitter videos through emotions portrayed by the faces of people in the videos. These emotions range from simply happy, sad, or neutral, to some more specific emotions such as anger, disgust, or surprise.

Dataset Description

The dataset for our project consists of videos collected off of Twitter using the Python library snscrape. These videos are grouped according to the search term used to find them, which can be a hashtag, a single word, or a phrase. For this project, the search terms used to demonstrate the program are “#COVIDBooster” and “COVID booster” since we wanted to focus on the general public reaction to the newly introduced COVID-19 booster shots. At present, the videos are downloaded in full according to how recently they were uploaded, and they are removed immediately after analysis to save space.

Methodology

Our methodology consists of 3 steps: gathering videos from Twitter, processing each gathered video using FER, and exporting overall emotion information as it pertains to a particular search term. To start out, our program searches through Tweets on Twitter using the snscrape Python library. Given a search term and the number of videos to scrape, it returns a list of URLs of video content. With the list of URLs, the program then takes each video and detects the average emotion of the videos by going frame by frame of each video and detecting the emotions of people in it. Facial detection in FER is done by using OpenCV's Haar Cascade classifier, and emotion recognition is done by a Keras model included in the library.

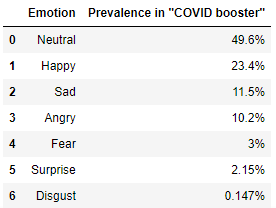


Example output frame with face detection and emotion scores

The Haar cascade classifier is a machine learning based approach where a cascade function is trained with, in this case, over 6,000 positive (pictures with faces) and negative (pictures without faces) test cases. The Keras model implements a pre-trained convolutional neural network validated on the FER-2013 dataset, which contains 35,887 images of faces that correspond to one of seven emotion classes: angry, disgust, fear, happy, sad, surprise, and neutral. According to the publication by Arriaga, Plöger, and Valdenegro (2017), the FER class has an emotion classification test accuracy of 66%. Our program was also tested for accuracy with an online dataset compiled by Rohit Verma which is similar to the FER-2013 dataset. With this testing, there was an accuracy rating of 67.83%, very close to the 66% shown in the paper. Once each video is processed frame by frame, the emotions of each frame are averaged out over all the videos for a key term to give the final results of that term.

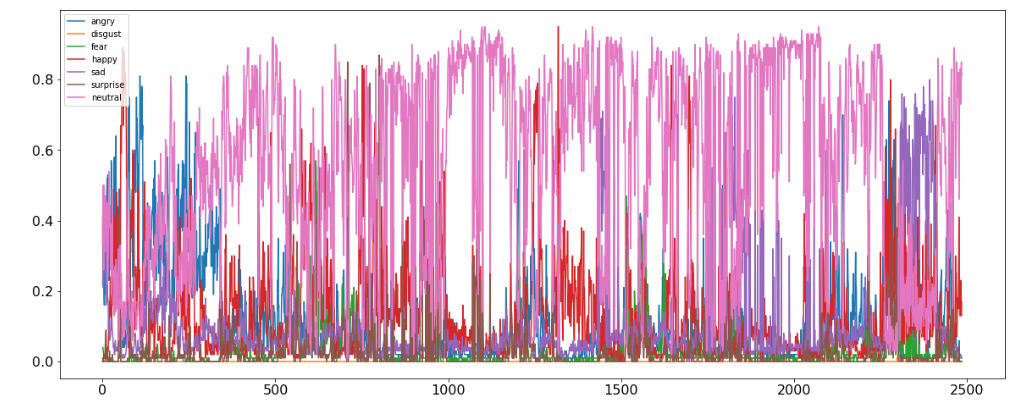
Results

The results of our program are expressed in a table for each search term, describing what percentage of the video’s frames were recognized by FER as having each emotion.



Dataframe of our output with phrase “COVID booster” and 5 Twitter videos

The goal here is to see the overall emotion regarding the search term. In the example output above, the most prevalent emotion is neutral, making up 49.6% of all frames. Thus, we can conclude (according to the program) that the general population has a neutral view on the search term “COVID booster.” A notable downfall in the data above is that the sample size was only 5 videos. To get a more ideal viewpoint of the overall population, the sample size should have been closer to 10,000 or more.

In addition to this table, there are other optional outputs that show the processing of the videos. One is shown as an example frame in the methodology section where the facial bounding box is displayed on the frame alongside the likelihood of the current face matching each of the seven measured emotions. This can be outputted for every frame to create a live video of the original video with the data overlaid. Another output is a graph depicting how the sentiments in each video change throughout the video.

Emotions plotted against each video frame

Conclusion

Sentiment recognition has become a much needed technology as the amount of video data being shared on the Internet continues to increase. In our program we used technologies such as OpenCV's Haar cascade classifier for face recognition and emotion recognition through a Keras model. With these technologies, we were able to create a Python program to detect the sentiments of gathered Twitter videos with a success rate that is consistent with previous tests using the FER library. One thing that we would like to do to improve efficiency of the program would be to use a GPU based computational method such as with tensorflow-GPU to improve the number of frames processed per second. Also, a change from the Haar cascade classifier to a more accurate network like MTCNN would reduce the amount of face misidentifications.

Contributions

Travis: Wrote code including data collection and implementation of FER, presented on walkthrough of code, proofread and edited presentations/report

Nathan: Created presentations, wrote report, presented on project introduction and applications

Hiep: Presented on key technologies used and future improvements for the project

Works Cited

Arriaga, Octavio, et al. “Real-Time Convolutional Neural Networks for Emotion and Gender Classification.” ArXiv.org, 20 Oct. 2017, https://arxiv.org/abs/1710.07557.

Global 2021 Forecast Highlights - Cisco. https://www.cisco.com/c/dam/m/en\_us/solutions/service-provider/vni-forecast-highlights/pdf/Global\_2021\_Forecast\_Highlights.pdf.

Verma, Rohit. Fer2013. https://www.kaggle.com/deadskull7/fer2013