Audio Search with Convolutional Networks

Atishay Jain, Author

I. INTRODUCTION

In 2017, a user on posted a short clip of the tv show 'The Big Bang Theory' on YouTube. The thing that set this clip apart was that the user muted parts of the video which contained canned laughter and no dialogues. So, when people saw this clip they saw the show as it was recorded, and they found out that the content of the tv show was not funny. This sparked a conversation on the internet about different tv shows that use canned laughter. Apart from The Big Bang Theory, the show that came to light during this conversation was F.R.I.E.N.D.S. When people saw some clips of the show without the canned laughter, they observed that the show was creepier at a lot of places than it was funny. This brought about the idea of watching all the episodes rather than short clips without canned laughter. Finding and muting the laugh track manually without a lot of manned hours, in this paper I discuss a deep learning solution to do this process.

II. PROBLEM AND DATA ACQUISITION

A. Problem

I started with an application of deep learning in mind and had to backtrack to the process to achieve it. I identified that there are three steps involved to get the desired results:

1. Detect whether the video/audio file has a laugh track
2. If it has a laugh track, system should output where in the original is it present
3. Perform action on laugh track. Either mute the laugh track or cut it and join back it to the original video.

After identifying these steps, I started to develop a deep learning solution for the first two steps. The problem boiled down to searching for a specific audio in a huge audio file. This problem statement has many more applications than detecting laugh tracks and it could be extended to apply to a lot of different applications.

B. Data Acquisition

Initial days of research included deciding between supervised learning and unsupervised learning. Supervised learning requires labeled data and I didn't have access to any kind of labeled data for laugh tracks. This problem could be solved with unsupervised learning but that wouldn't accomplish the task of searching through the file. I solved these issues by manually creating the data and utilizing semi-supervised learning.

I downloaded a twenty-minute video clip of The Big Bang Theory from YouTube and processed that in Adobe Premiere Pro. With the help of Premiere Pro, I manually cut out 3-5 seconds clips of clear laugh tracks and dialogues. I created 20 clips of laugh tracks and 23 clips of dialogues using this approach. I further downloaded 3 clips of canned laugh tracks from YouTube which provided a more general laugh tracks rather than specific to The Big Bang Theory.

III. APPROACH AND ALGORITHM

There have been a lot of advancements in Convolutional Deep Learning for images which motivated me to utilize the Convolutional networks. I converted all my audio data intro

spectrograms and saved those as images so that they could be processed by a convolutional network. Spectrograms are a way to represent the audio files in images by plotting time on x-axis, frequency on y-axis and amplitude of a frequency at a time is represented by the intensity or color of each point in the image. After looking at different spectrograms of laugh tracks and dialogues, I observed a vast difference between there spectrograms. Fig 1 shows a spectrogram of a laugh track and Fig 2 show a spectrogram of a dialogue. There is a vast difference in both the intensity and frequency between the two. Laugh tracks have much more intensity and frequency than the dialogues and we can distinguish them clearly.

I used Inception model by Google which pre-trained on the ImageNet dataset which has over 14 Million images. Since, I have very less available data I utilized transfer learning. Since, this network was pre-trained on images, it can detect low level features from images very well. I then trained the model using the 46 audio files of dialogue and laugh tracks. On this trained model I passed the whole episode of The Big Bang Theory. I processed the whole episode in chunks of 5 seconds, to get 50 outputs of each laugh tracks and dialogues. I then manually listened to these 100 tracks to verify how the system did. I then used these 100 tracks to retrain the model. To test the system, I downloaded an episode of F.R.I.E.N.D.S. and then processed it again in chunks of 5 seconds. The system outputs 10 audio files each of laugh tracks and dialogue. It also outputs the time interval in multiples of 5, since the chunk size is 5, of where the laugh tracks are present with respect to the whole episode.

IV. Result

After listening to the output of system after passing the F.R.I.E.N.D.S. episode, I got some interesting insights.

1. I observed the system can classify and extract laugh tracks as well as dialogues with high confidence.
2. In some cases, it was also able to detect laugh tracks which were overlapped with dialogues as laugh tracks.
3. The system gives out false positives when the other sounds apart from dialogues and laugh tracks. In one case it was fooled into thinking bell sounds as laugh track. Further analysis showed, in that case the spectrogram of bell sounds was very similar to that of the laugh tracks.

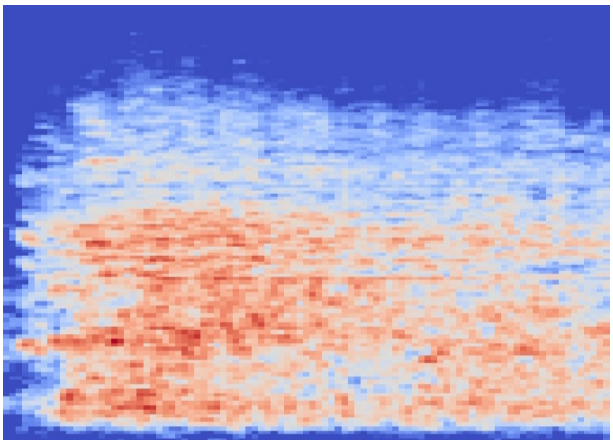

Figure 1: Spectrogram of Laugh Track

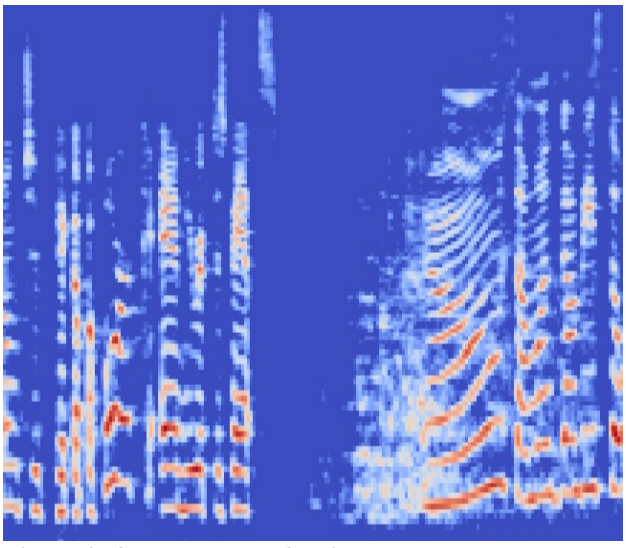

Figure 2: Spectrogram of Dailogues

V. Conclusion

I explored one application of audio search to stream an audio file and find laugh tracks with in it. This system can be applied to many more applications for searching a specific audio in a huge file. To handle the false positives,

we can look to explore more features than just spectrogram and combine them to get better predictions.