Findings

Task 1 – Web scrapping

Observation for Deep Learning NPTEL course
The download_audio.py works but when we play the audio the duration doubles but this gets fixed when we perform the second task
The download_transcript.py works – but it is quite slow

I did try to use yt-dlp-tools however since the links are from nptel sever not youtube servers, the codes I tried did not work hence I went with ffmpeg

Testing with another course link - https://nptel.ac.in/courses/106106248
When I executed with another NPTEL course link, it is able to extract the links from the website, but we can observe during audio extraction from the links using ffmpeg. This is due to the fact that the video links given in the website do not work.

Download_transcript.py works well

Task 2 – preprocessing audio

Observation:
Works fast when we increase the number of cpus and it saves the audio in their original durations.

I did think of another preprocessing step where we can crop out the background music in the last 10 seconds, and that script works well. However, I think we should crop more because the background music is more than 10 seconds at the end for many videos and is also present in the first 10 seconds of the video.

Task 3 – preprocessing text

Observation:
Extracts the normal text in the pdf but it does not extract the text in the images in the pdf file. I tried integrating pytesseract however, I am having some dependency issue persistently.

Task 4 – Creating manifesto

Observation:
Works fast but it doesn't save the list in the order of the audios, it is quite random.

Task 5 – Dashboard

I used Tableau for creating the dashboard.