***wordprocessing.py***- makes a list of list of all words in each page of a slide/pdf document

**command line**

*python wordprocessing.py -t [address to the text document(output of apache-tika)] -n(optional.define in case you want numbers)*

**NOTE**:the text document should be the output of tika in structured-format for detection pf slide changes.

***ground\_truth.py***-for building the groundtruth.It uses the wordprocessing script to build a list of words for each slide and opencv for alignment of slides with video.Outputs a json file containing list of lists,each list containing the starting and ending frame numbers,the start and end position in video in msec,for each slide in video along with all the words in the slide.

**command line**

*python ground\_truth.py -v [specify the path to video] -o [specify the path to output and name of the output file] -t [address to the text document(output of apache-tika)] -n(optional.define in case you want numbers)*

upon running this command a window opens playing the video.Use **space** to pause the video.Once paused, you can assign a frame as starting by pressing the key **'s'** and as ending frame by pressing the key **'e'**.Press **space** again to play the video and repeat the steps for next slides.The video will automatically stop once we have obtained starting and ending frames for each slide,or if the video playback time is exceeded,in which case unmarked slides will have 0 as starting/ending frame numbers.In case you want to exit midway,just press the key **‘q’** while video is running.

**NOTE**: \*requirement openCV

\*the text document should be the output of tika in structured-format for detection of slide changes.

***loovprocessing.py*** script to run loov on the video for each slides and save the loov output for each slide in a directory.Then process the loov output to obtain a list of words and evaluate(find average and individual precision and recall) by comparing with the groundtruth.

**command line**

*python loovprocessing.py -v [specify the path to video] -g [path to the groundtruth,ie the json file we obtained from ground\_truth.py] -l [specify path to build directory of LOOV ending with '/',e.g /home/here/is/LOOV/build/] -o [specify the output directory ending with '/',e.g /home/my/output/directory/]*

output of this program- \*loov output of each slide is saved sequentially as .OCR files in the output directory

\*list of all words in loov output for all slides is saved as list.json in the output directory

\*list of individual precision for each slide(printed on command prompt)

\*list of individual recall for each slide(printed on command prompt)

\*average precision(printed on command prompt)

\*average recall(printed on command prompt)

\*total number of words in ocr output for the video(printed on command prompt)

\*total number of words in the groundtruth for the video(printed on command prompt)

**NOTE**-in certain cases,there might be a single slide covering multiple videos,in that case use cat to combine the videos using the command

*ls 01.mp4 02.mp4 | perl -ne 'print "file $\_"' | ffmpeg -f concat -i - -c copy* Movie\_Joined.mp4

01.mp4 and 02.mp4 are the videos to be joined and Movie\_Joined.mp4 is the name of the new combined video.

similarly there might be multiple slides for one video,we can use cat to combine all the slides for a particular video.

**Edit**

**convert.py** to convert a video from color to binary.A makeshift solution to address the LOOV problem where color text was not being recognised

**commandline**

python convert.py -v [*specify the address of the input video*] -o [*specify output video name with path*] -b [*window size while finding threshold in adaptive thresholding*]

Once you run this program the video plays in binary and once the video playback is complete,you will get an output video file which is the binary version.In case you are not happy with the binarization you can stop the program midway by pressing **space** and then make suitable changes to the parameters and rerun the script.