PROGRESS REPORT(PROJECT DIARY) :

Name : Hiren Bavaskar

Roll : 190050049

Project : Unscripted SOC 2021

DATE	PROGRESS
April Week 1, 2021	Revised basics of Python and Git. Revised Numpy, Pandas and Matplotlib libraries. Solved the following exercises from Google developers course: ● Numpy colab ● Pandas colab
April Week 2, 2021	Read 2 chapters of the prescribed ML textbook. Read about univariate regression, supervised learning, unsupervised learning, batch learning, online learning and exercises on scikit. Solved some exercises in python for ML in Jupyter notebook.
April Week 3 - May Week 1	4th Semester Endsems
May Week 2, 2021	I learned about NLP from Stanford lectures and other resources[Resource]. Got to know about word vectors, stop words, regex, word and sentence tokenization. Read about the nltk package in python and looked over some code snippets.
June Week 1, 2021	Started exploring Speech Recognition Learnt about ASR(Automatic Speech Recognition) There are 2 kinds of models: ● Acoustic model - Deals with sound to phonetic representation ● Language model - Deals with the language structure Learnt use of hidden markov model (HMM) and Deep Neural networks (DNN) in ASR Started exploring DeepSpeech library for STT Learnt about WER (word error rate) and the use of

	jiwer library in python for comparing transcribed and original text The STT and WER calculation files can be found in the github repo.
June Week 2- 3	Couldn't do much because of unavailability of my laptop
June Week 4, 2021	Started huggingface crash course [link] Learnt about the use of transformers library and the pipeline module in various applications of NLP like: ● Classifying whole sentences ● Classifying each word in a sentence ● Generating text content ● Extracting an answer from a text ● Generating a new sentence from an input text Pipeline object of transformers library is useful for executing NLP tasks on one or several lines of text. Pipeline by itself chooses a model appropriate for the task and does the required pre and post processing with the pre-trained model to make reasonable predictions. Eg: Sentiment analysis, classification based on labels(zero shot classification), text generation, filling masks(fill in the blanks). An ideal way to use transformers is to take a pre-trained model and then fine tune it according to our use. Learnt about encoder models, decoder models and sequence to sequence model(combination of the 2) Encoder: Takes the input and gives a vector representation of its features. Useful for understanding inputs like sentence classification. Decoder: Takes representation of features along with the input for generating output. Useful for generative tasks like text generation. Tokenizer -> Model -> Post processing Tokenizer analyzes the raw text and converts them into numbers for neural networks to understand Models can only process numbers so they take the output of tokenizers and do the further predictions.

	Implemented wav2vec model and transformers for speech to text in english using audio interval feature of librosa (to avoid crashing of file in colab) Colab file link [Number of hours spent this week: 11-12] (Studying course: 8-9 hrs, coding and references: 3-4 hrs)
July Week 1, 2021	Agenda for this week was to find good datasets to fine tune our model for speech to text and for documents summarization. https://www.openslr.org/12/ The above site contains about 300MB of data. The content involves .flac files of 2-10 seconds with their corresponding correct transcripts. https://www.kaggle.com/sunnysai12345/news-summary?select=news_summary.csv This site provides the text files (news articles) and provides their summarized text. The file is in the type of CSV files. It is about 50 MB of data which will be sufficient for fine tuning. https://colab.research.google.com/drive/1bewbhx9Ej4_iyeM9eBO_M_bPhHu48_Ap?usp=sharing This is the testing done on our voices. It's still gibberish so we are thinking of not using the wav2vec2 model ahead. https://colab.research.google.com/drive/13yuNiuNmYP9Bl-itd2t5cn5_QsV6ZSoe#scrollTo=ZHLHvkPy85Zy Further part of text summarization was taken upon by Harshit. [Number of hours spent this week: 9-10]
July Week 2, 2021	Final week before project submission Used google speech recognition library for speech to text for large audio files. However, it is running correctly only on US-English language.

	Tested on 52MB audio file(5 mins of audio) and the results were good enough Added summarizer python file which summarizes the transcription into any number of lines of summary the user wants. Added documentation of the project in github [Number of hours spent this week: 14-15 hrs]