# Project Proposal
# Lecture Summarizer using Extractive Summarization Approach

**Drashti Bhuta, Jatin Taneja**
CS6120 NLP Spring 2018
bhuta.d@husky.neu.edu
taneja.j@husky.neu.edu

**Introduction:**

A lecture in an educational institute plays an important role in understanding a concept. These lectures are usually long with too much information conveyed in each session making it difficult for student to Grasp. These volume of information is invaluable source of knowledge which needs to be summarized to be made useful. Moreover, students with disability require additional tools or resources in order to follow with these lectures. Unavailability of such resources may put disabled students behind in their academic performance. As a responsible citizen it is our duty to ensure that those students with disabilities have equal access to the academic experiences at their University. Considering the above mentioned difficulties for any student we propose a tool that will make it easy for students to get a gist of their professor's lecture. This would allow students to participate more in the lecture instead of taking notes.This tool will aid students with disability to understand the concept taught in class with significant learning gain.

The tool mainly addresses the inability of those students to take notes in the class along with participation. The application will record the ongoing lecture in the class. This recorded lecture would be converted to text and fed as an input dataset to the text summarization model.The text summarization model will process the given data to generate a best possible summary of the data. This generated summary will be converted into an audio recording and Braille text as well for disabled students.Our approach for text summarization will use different algorithms to generate a gist of the given data then evaluate those generated results and the best scored result will be used as the summary of the given lecture to be made available to the target audience.

**Related Work**

Text summarisation is the process of creating compressed version of given data. Summarisation techniques are majorly divided into extractive techniques and abstractive techniques. Extractive approach selects subset of sentences from original text to create summary. On the other hand, abstractive approach learns internal language representation and generates human like summaries by paraphrasing the intent of original text. [4] Extractive summaries yield better results than abstractive summaries.This is because semantic representation and generating inferences from natural language is much harder and has not yet reached a matured stage.

Paper [1] introduces various techniques to generate extractive summaries. They are as follows:

1) Topic Words:

In this approach, we first identify the topic of the text and then weight each sentence on the bases of count of topic words it contains i.e. higher the score, more relevant to topic.

2) Frequency driven approaches:

In this approach, weights assigned to the word uses two techniques: - word probability and tf-idf to decide which words are co-related to the topic.

3) Graph methods for summarization

In this approach, sentences represents the node of the graph and edges between two nodes is use to denote the similarity between those two sentences. Normally, we use cosine similarity to identify the edges weight. There are two main outcomes from this graph representation

a) It identifies discrete topics covered in the graph.
b) It identifies important sentences in the document. I.e sentences that are connected to many other sentences are likely to be included in the summary.

4) Machine learning based techniques - In this approach, we build a classifier to segregate summary and non summary sentences. Classifier is trained to locate particular features in the sentences and calculate their probabilities.

Paper [2] explained summarization approach based on maximizing volume in a semantic vector space. The choice of the sentence to be included in the summary is decided if convex hull maximises the volume in that semantic space.To achieve this paper introduces a greedy algorithm based on Gram-Schimdt process to efficiently perform volume maximization.

The paper [3] generates extractive summary by using an unsupervised document summarisation method that creates summary by clustering and extracting representative sentences from each cluster.The function used for sentence clustering are proposed as follows

a) Similarity based on term co-occurrence
b) Objective function which clusters the document in a way that objects within each group would be highly similar to each other than to objects in other group.

Modified discrete differential evolution algorithm(MDDE) was proposed to maximize the objective function.

Paper [4] is exploring graph based methods for computing importance of a sentence in a given document. It identifies the most important sentences in a document or a set of document. This paper hypotheses the sentences that are similar to many of the other sentences in a cluster are central to the topic. For this a degree centrality is calculated. It considers a new approach called Lex-rank for computing the sentence importance based on the concept of eigenvector centrality in a graph representation. Eigenvector centrality considers every node having a centrality value and distributing this centrality towards neighbors.This algorithm uses a connectivity matrix of intra-sentence cosine similarity as the adjacency matrix. Another advantage of this algorithm is that it eliminates the effects of high idf-score from boosting the score of the sentence that is unrelated to the topic.

**Overview**

In Order to obtain a thorough understanding of the right technique to choose for achieving our objective, we would like to explore more resources on text summarization and key phrase extraction. Once we have enough understanding to proceed we will begin with the implementation as per following steps:

1) Convert speech to text: We will convert the recorded audio to text format. We are planning to use Amazon's API LEX, known and verified for this purpose.
2) Text Summarization: This phase is the core of our proposal. We are planning to generate the summary of the given text using following two techniques.

a) Text Rank algorithm using BM25 [5] (Unsupervised)

Following are the steps that we will follow to implement this algorithm

- Preprocessing the text to remove stop words and stemming.
- To create a graph where each sentence is a node and the edge between two sentence represents how similar the sentences are.
- The measure of similarity is the weight assigned to each node and the weight is calculated by BM25 algorithm.
- Run the pagerank algorithm on the graph with the weight measure as described above.
- Select the vertices with page rank score above threshold value.

b) Lex rank or lexical page rank algorithm [4] (Unsupervised)

- This algorithm is based on the concept of centrality. According to this concept sentences which are similar to many other sentences in the cluster are more central to the topic. This can be clarified by following two points
    - How to define the centrality between two sentences.
    - How to compute the centrality of a sentence given its similarity with every other sentence.
- Similar to Text Rank, first we need to do preprocessing of data i.e stop words removal and stemming.
- We need to find stationary distribution of a Markov chain model.
- We will calculate a new measure of similarity i.e similarity lexical page rank or lex rank by using cosine similarity.
- We will calculate Lex score of graph by setting damping factor to some random value X. We will experiment with different

values of damping factor to select the best output.

3) Process summarized text to other formats (audio & braille script or translate it into any other language) : We aim to cater disabled students along with other students and to achieve the same we will convert the summarized text into audio format and braille script. In order to increase the usability of this tool we will try to translate the obtained summary to other languages.

**Dataset and Evaluation**

We will be using a small dataset created by us which will comprise of recorded audio available on internet. The primary purpose of this dataset is to evaluate the performance of the system in whole. In addition to this, we are planning to use DUC 2004 dataset that comprises of news documents in order to evaluate the performance of our text summarisation module. We will consider the ROUGE metrics for evaluation of the generated summaries.

**References**

[1] M. Allahyari, S. Pouriyeh, M. Assefi, S. Safaei, E. D., J. B., and K. Kochut, "Text Summarization Techniques: A Brief Survey," International Journal of Advanced Computer Science and Applications, vol. 8, no. 10, 2017.

[2] D. Yogatama, F. Liu, and N. A. Smith, "Extractive Summarization by Maximizing Semantic Volume," Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 2015.

[3] R. Alguliev and R. Aliguliyev, "Evolutionary Algorithm for Extractive Text Summarization," Intelligent Information Management, vol. 01, no. 02, pp. 128–138, 2009.

[4] LexRank: Graph-Based Lexical Centrality as Salience in Text Summarization, www.cs.cmu.edu/afs/cs/project/jair/pub/volume22/erkan04a-html/erkan04a.html.

[5] "Text Summarization in Python: Extractive vs. Abstractive Techniques Revisited." *Machine Learning Consulting*, rare-technologies.com/text-summarization-in-python-extractive-vs-abstractive-techniques-revisited/.