Analysing the text of Mr. Modi's speeches*

*Text analysis of speeches given by Mr. Modi between Aug'14 and Aug'20

Neha.G.Vyas
Electrical Engineering
IIT Bombay
Mumbai, India
190070037@iitb.ac.in

T Pavan Kalyan
Electrical Engineering
IIT Bombay
Mumbai, India
190020124@iitb.ac.in

Shah Zeel Sanjay
Humanities and Social Science
IIT Bombay
Mumbai, India
19b080026@iitb.ac.in

***Abstract*—Narendra Damodardas Modi is an Indian politician serving as the 14th and current Prime Minister of India since 2014. The very popular Mr. Modi is largely known for his attractive speeches. Here we try to analyse the text of the speeches given by Mr. Narendra Modi on various occasions dated from Aug'14 to Aug'20. The primary data is taken from Kaggle[1]. Another interesting source of data is Google search trends[2]. A compilation of these two data sets gives an important insight that there is a relation between the popularity of Mr. Modi and his speeches. We try to find the main topics his government deals with through his speeches. We try to predict the period in which a particular act or scheme was implemented based on the text of the speech. A sentiment analysis was performed on the speeches using lexicon based method to find the sentiment of the speeches given. Finally the speeches were summarized using various methods of extractive summarization.**

***Index Terms*—Topic Modelling, Sentiment Analysis, Summarizing**

I. Introduction

The text analysis is widely divided into four different ideas which include topic modelling, sentiment analysis, and summarizing and some predictions of the time period of implementation of various schemes by the government. The text presented in the speeches was in both English and Hindi which poses a problem of converting the text into one language. The earlier analysis done on the same data set worked with the English speeches completely ignoring the Hindi speeches, But the number of Hindi speeches are comparable to the number of English speeches. The text is unstructured and the problem in hand is an unsupervised machine learning problem. One of the interesting analysis done is predicting the year in which a particular scheme was introduced. A list of schemes was gathered from [3] and the data set can be converted into a structured data using TF-IDF method.

The next part of analysis is topic modelling. Earlier work on topic modelling was based just on LDA. Here we show the results using both LSA and LDA through visualization using UMAP and pyLDAvis. We also draw some important conclusions from the word cloud of each topic out of 10 most significant topics. The next step is sentiment analysis. There is not much addition to the related works which can be found here[3] but the results are beautifully presented here using pywaffle charts. The final part of the analysis deals with summarizing the speeches using 3 different techniques and analysing the results of these summarizing techniques. A sample of the summaries is also presented here.

II. Data set

A. Speeches given by Mr.Narendra Modi-The Prime Minister of India from Aug'14 to Aug'20

The Data set is available at kaggle[1]. It consists of the transcript of the speeches given by Mr. Narendra Modi who is serving as the 14^{th} Prime Minister of India since 2014.The transcript of these speeches is publicly available at [4]. The data set contains text both in Hindi and English languages. The data set contains following features: 1. date: Date on which speech was given 2. title: Title of the speech as mentioned on the website 3. url: Link to the official transcript 4. lang: Language in which the speech was given 5. words: Total number of words in the speech 6. text: Transcript of the speech The column 'lang' can be used to find the proportion of Hindi speeches. Since, the number of speeches in Hindi are comparable to that in English, the feature 'text' requires cleaning and translation into a single language.The Hindi speeches are translated into English using googletrans that implemented Google Translate API. It is used to detect the language and translate into a target language.

B. Google search trends data of the word 'MODI' for India

The Data set is available at [2]. The data is extracted from the google search trends homepage by searching the term 'MODI'. The data is extracted for the period between Aug'14 and Aug'20. The data set contains only two features: 1. index : Year and month for which the number of searches were measured. 2. Category: All categories : This has integer values from 0 to 100. A value of 100 is the peak popularity for the term. A value of 50 means that the term is half as popular. A score of 0 means there was not enough data for this term.

III. Analysis Pipeline

The challenge of this data set is text mining. Text is unstructured data which makes the analysis difficult. The entire analysis is divided into four broad goals:

- Analyze the year in which a particular event was introduced by his government.

- Topic modelling - Extracting the most significant topics from the text
- Sentiment analysis- classification of the given speech into positive, negative or neutral using unsupervised machine learning techniques.
- Summarizing- extracting the summary from the speeches using various techniques.

A. Analyze the year in which a particular event was introduced by the government

The Assumption is that when a new initiative is taken or a new event occurs it is talked about considerably more as compared the previous year. So, the question that needs to approached is to find the year where the initiative is talked about the more as compared to the previous year.There are a few ways to convert the unstructured data into structured data set. One of the ways is TF-IDF method(Term frequency - inverse document frequency method for text analysis).The method intends to reflect how important a word is to a document in a collection or corpus.

Term frequency(tf): The no. of times a word shows up in a document/ Total no of words in the doc.

Document frequency(df): The no. of docs in which the word appears

Inverse document frequency(idf): log(N/df) where N is the no. of documents in the corpus.

$$tf_idf = tf * idf \tag{1}$$

Understanding the method:

The tf-idf value always belongs to [0,1] (both included). This value can be zero under 2 conditions: i. tf is equal to zero which happens when the word is not present in the particular document of the corpus. ii. Idf is zero. which is when the word is present in all the documents.

So, if all the tf-idf values associated with a word is zero then the word is present in all the documents. But , if it is zero for a particular document the it is absent only in that document.Additionally, if all values associated with the word are not zero, then the tf-idf value variation depends on tf, which means higher the value more important the word is for the document.

B. Topic Modelling

Topic modelling is an unsupervised machine learning technique where the most important topics from the text are identified based on parameters like word frequency, distance between words. Here Topic modelling is implemented using two different algorithms :

- Latent Semantic Analysis
  This method is primarily based on Distributional hypothesis which is in simple words "linguistic items with similar distribution have similar meanings". Hence we compute how frequently the words occur in the text. assuming the entire text just to be a bag of words without any syntactic and semantic information. We basically divide the corpus into two :
  1. a list of topics covered by documents in corpus
  2. set of documents grouped by the topic they cover
  First step is to clean the data i.e. removing all characters except alphabets. removing short words like conjunctions and finally converting entire text into lowercase to nullify case sensitivity.Then tokenize the text, i.e. split the text into individual words. Remove the words like 'it', 'when', 'while', 'about', 'because' etc. These words are called stopwords. Once the stop words are removed the tokens are stitched back. Here we will create a document term matrix A. Given m texts and n unique words in our vocabulary, we can construct a mxn matrix A in which each row represents a document and each column represents a word. This matrix is filled by tf-idf scores(term frequency-inverse document frequency). This assigns a weight for term j in the document i as follows:

  $$w_{i,j} = tf_{i,j} * log(\frac{N}{df_i}) \tag{2}$$

  Hence we use TfidfVectorizer to create a document term matrix. The next step is Matrix decomposition and dimensionality reduction:
  1. We will use TruncatedSVD to decompose A into multiple matrices and also to reduce the number of topics to 20. This is a hyperparameter and can be finetuned to get the best results possible.
  2. TruncatedSVD or singular value decomposition factorizes A into 3 separate matrices where dimensionality is reduced by keeping 20 most significant topics. The matrix is decomposed as $A = USV^T$
  where U is the document term matrix and V is the term topic matrix.
- Latent Dirichlet Allocation
  The purpose of both the algorithms are same but the main difference between LDA and LSA is that LDA assumes that the distribution of topics in a document and words in a topic are dirichlet distributions unlike LSA where it does not assume any predefined distributions. More about the algorithm can be found out from [5] which is the original paper.The first step here is the same as LSA. we will also normalize the corpus here using WordNetLemmatizer.Then convert the corpus into document term matrix using a robust and efficient library called gensim. Train LDA model on the document term matrix using gensim library.
- One of the applications related to topic modelling is to predict if a given speech was given pre or post covid19 period. This is done by supervised learning SVM. To apply SVM convert the unstructured data into structured numeric data which can be done by in-built module for tf-idf.

C. Sentiment Analysis

Sentiment analysis determines the attitude or the emotion of the text, i.e., whether it is positive or negative or neutral. The sentiment function of textblob returns two properties, polarity,

and subjectivity. Polarity is float which lies in the range of [-1,1] where 1 means positive statement and -1 means a negative statement.Here, This is a lexicon based approach where a predefined set of words denote a text as positive, neutral or negative.

D. Summarizing

The Main objective of this part is text summarizing i.e. to create a summary speech that is coherent and captures the salient points in the speech. Average words in Modi's speeches in the dataset are 12,374. Summarizing will help in reducing time and accelerates the process of understanding key points covered in the speech.

Approach for text summarizing was extractive i.e. where important sentences are selected from the input text to form a summary.The basic approach is as given below:

- Split speech into sentences and pre process each sentence including tokenization, stop word removal and lemmatization
- Assign algorithmic specific score to each sentence and based on scores select top n sentences as summary.

For score calculation, Three different methods were used:

- Term Frequency: frequencies of terms appearing in a sentence are added up to calculate score.
- Text Rank: sentence term matrix is used to cosine similarity between sentences. The similarity matrix is used to construct a graph, where sentences are nodes.
- Latent Semantic Indexing (LSI): matrix factorization is done with Singular Value Decomposition.

IV. Results

A. Analyze the year in which a particular event was introduced by the government

As tf-idf is the numeric value representing the relative importance of a word in a particular document in a Corpus. We use the words in the title of the scene to find the tf-idf value for a particular scheme.

We will plot the increase in importance of the scheme over the years along with the actual year of introduction(in red).

- If any initiative is taken in 2014 and talked about quite frequently (important consequences) then the idf is 0 through all the years so the difference in the consecutive years. So, it is equally important all over the years.
- Startup India was introduced in the year 2015. The year sees a peak in the tf-idf value. The subsequent peaks represent the consequence of the event.
- Ujjwala Yojna was introduced in 2016. So, there is a peak during 2016,2017.
- Beti Bachao,Beti Padhao was introduced in 2015 and it shows a peak at 2015.
- First case of corona was found in 2020 in India and it shows a peak at 2020.
- GST was introduced in 2017 and it shows a peak at 2017

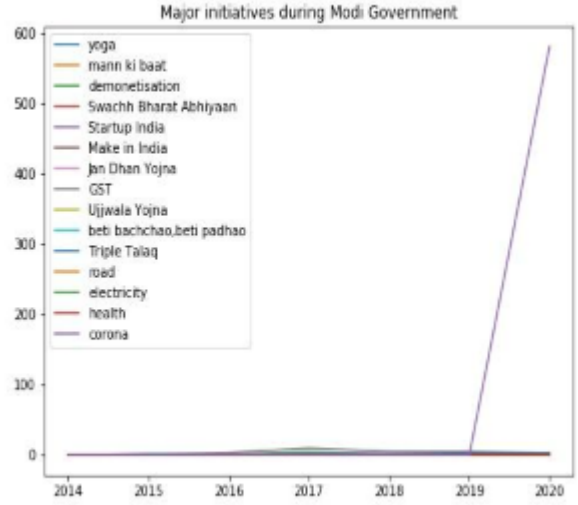

Fig. 1. Graph of tf-idf for various schemes and events over the years

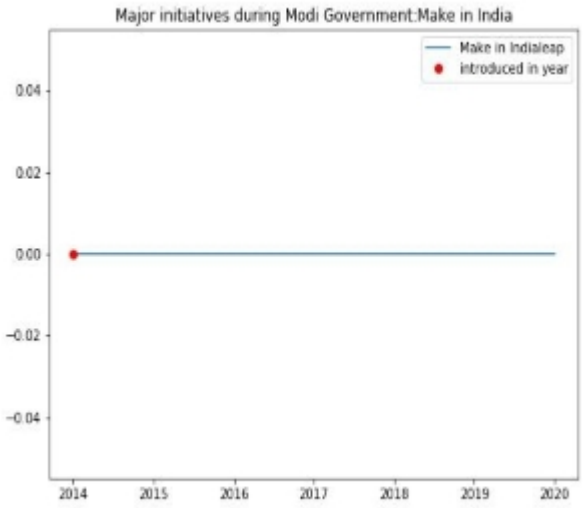

Fig. 2. Major initiatives during the modi government - Makein India

- Triple Talaq was introduced in 2019 and it shows a peak at 2019. It has a peak in the year 2017 which may represent the problem.
- While the graph for demonetisation does not follow the assumption which leads to inaccuracy of the assumption . This might be because of the consequence of the event is more talked about.

B. Topic Modelling

- Latent Semantic analysis
  Visulaization of LSA models is quite difficult. We reduce the dimensionality here and visualization is done using

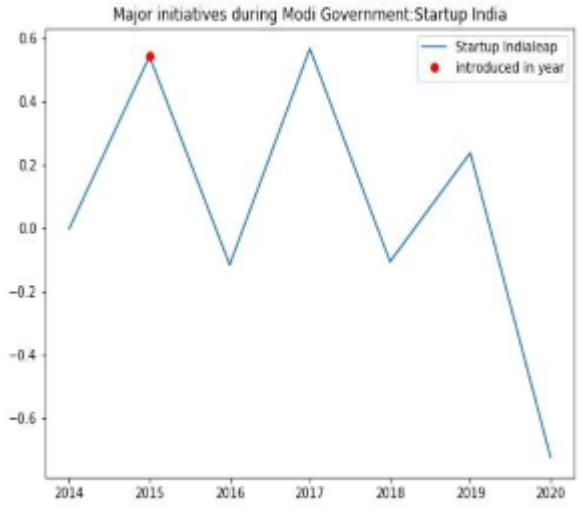

Fig. 3. Major initiatives during the modi government - Startup India

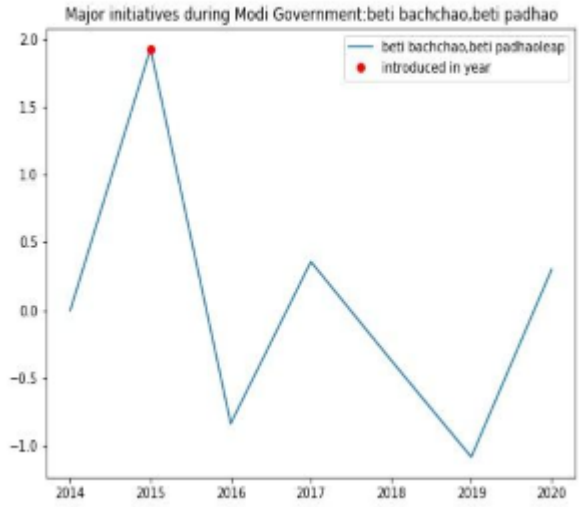

Fig. 5. Major initiatives during the modi government - Beti Bachao Beti Padhao

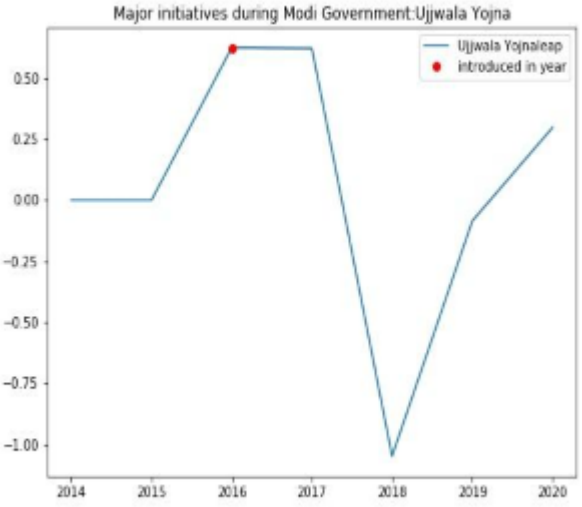

Fig. 4. Major initiatives during the modi government - Ujjwala Yojna

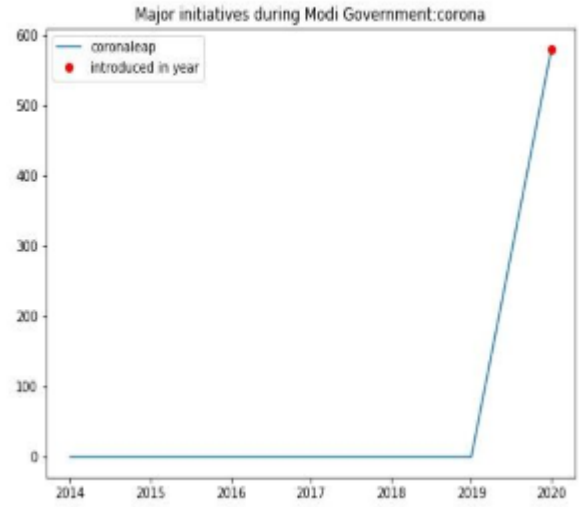

Fig. 6. Major initiatives during the modi government - Corona

UMAP(Uniform manifold approximation and projection) which is a scalable and efficient dimension reduction algorithm. Here we have used the following parameters: n-neighbors: which controls local versus global structure. min-dist: controls how tightly UMAP is allowed to pack the points together

Here in this plot, each blue dot represents a text and the colors represent the most significant 20 topics. As it can be seen that the topics have covered all the texts properly, it shows that our model performed well.

- Latent Dirichlet Allocation

  Here we see all the significant words present in four topics from which some important observations can be drawn out of the ten most significant topics using Word cloud.

  pyLDAvis is an interactive LDA visualization python package.The results of LDA model are visualized using this. The area of circle represents the importance of each topic over the entire corpus, the distance between the center of circles indicate the similarity between topics. Here is an inter-topic distance map.
- One of the interesting analysis was to develop a model to predict if a speech was given after the pandemic or

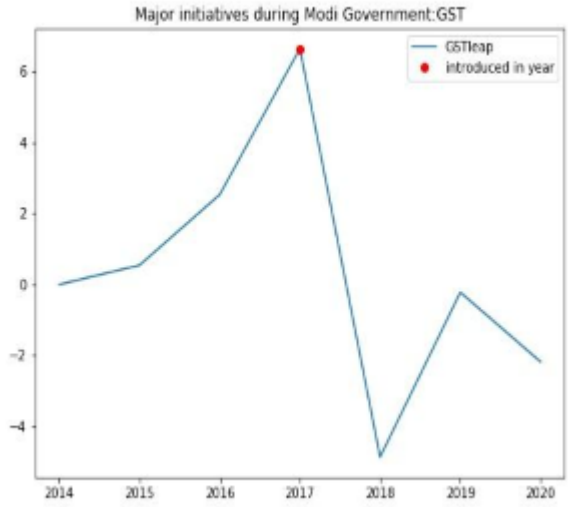

Fig. 7. Major initiatives during the modi government - GST

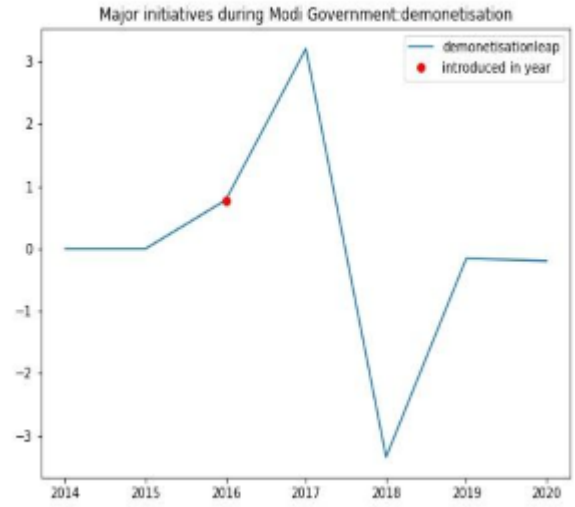

Fig. 9. Major initiatives during the modi government - Demonetisation

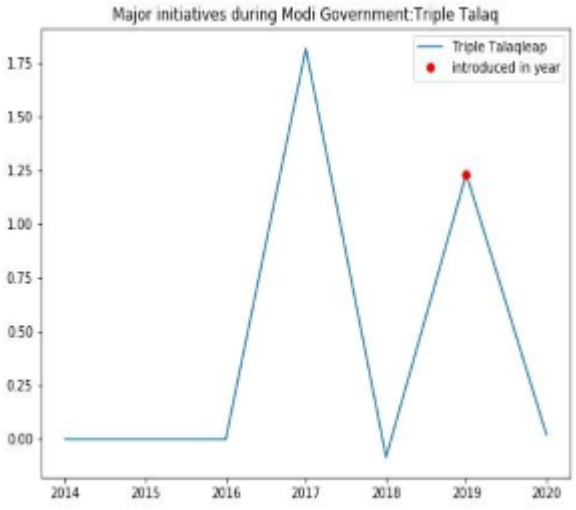

Fig. 8. Major initiatives during the modi government - Triple Talaq

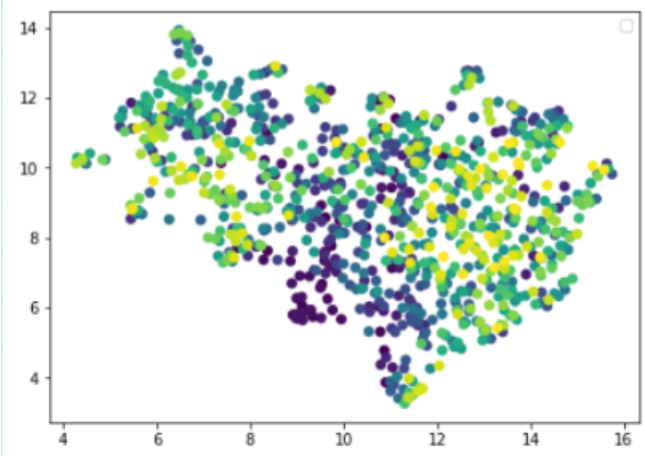

Fig. 10. UMAP - Latent Semantic Analysis

Fig. 11. Word cloud of last four topics

before it's arrival. The model was trained Using SVM.The outcome of SVM is 69.44%.

C. Sentiment Analysis

The results of the sentiment analysis can be visualized using pywaffle.PyWaffle is an open source, for plotting waffle charts. It provides a Figure constructor class Waffle, which could be passed to matplotlib.pyplot.figure and generates a matplotlib Figure object.

D. Summarizing

A sample from the data set i.e. speech given on Jul 04 2020 was summarized using three techniques. some extracts from

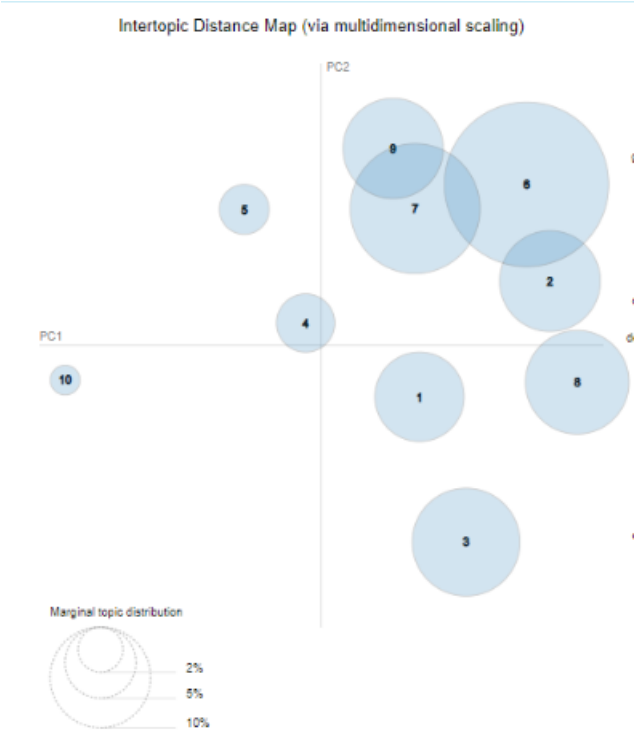

Fig. 12. Visualization of the results of LDA model using pyLDAvis

Fig. 13. pictograph using pywaffle depicting the proportion of speeches that are positive, negative and neutral

the summaries are presented here:

- Term frequency
  "I would urge my young friends to also stay connected with the thoughts of Lord Buddha. If you want to see a great example of how hope innovation and compassion can remove suffering it is our Start-up sector. Friends in his very first sermon in Sarnath and his teachings after that Lord Buddha spoke on two things- hope and purpose. You know how people also know my parliamentary constituency of Varanasi? We have to rise to the occasion and do whatever we can to increase hope among people. This would bring so many people pilgrims and tourists. It would also generate economic opportunities for many. Friends the eight-fold path of Lord Buddha shows the way towards the well-being of many societies and nations. This hope comes from my young friends- our youth. Friends it is the need of the hour to connect more and more people with Buddhist heritage sites."
- Text Rank
  "You know how people also know my parliamentary constituency of Varanasi? We want to focus on connectivity to Buddhist sites. We in India have many such sites. We have to rise to the occasion and do whatever we can to increase hope among people. To these challenges, lasting solutions can come from the ideals of Lord Buddha. This would bring so many people, pilgrims and tourists. This is a day to remember our Gurus, who gave us knowledge. This hope comes from my young friends- our youth. They will motivate and show the way ahead. They were relevant in the past."
- Latent Semantic Indexing (LSI)
  "Respected President Shri Ram Nath Kovind Ji, other distinguished guests. Let me begin by conveying my greetings on Ashadha Poornima. This is a day to remember our Gurus, who gave us knowledge. The teachings of Lord Buddha celebrate simplicity both in thought and action. If you want to see a great example of how hope, innovation and compassion can remove suffering, it is our Start-up sector. I would urge my young friends to also stay connected with the thoughts of Lord Buddha. Infact, Lord Buddha's teaching of – appah deepo bhavah:, be your own guiding light is a wonderful management lesson. To these challenges, lasting solutions can come from the ideals of Lord Buddha. A few days back the Indian Cabinet announced that Kushinagar airport will be an international one. May the thoughts of Lord Buddha further brightness, togetherness and brotherhood."

OBSERVATIONS

According to word cloud of the various words in the top 10 significant topics using LDA model following inferences can be drawn: Observations:

Topic 9: this has words like lord,ayurveda, buddha, compassion which points mainly at the "cultural heritage of India"

Topic 8: Words like society, life, time and people draws attention at the discussion of "general life of people in india"

Topic 7: it talks about country and the government in general.

Topic 6: it talks about india and the world in general.

Topic 5: it talks about country and the government in general.

Topic 4: This topic points at festive mood and unity

Topic 3: deals with the international affairs

Topic 2: points at economy of the country.

Topic 1: it talks about country and the government in general.

Topic 0: it talks about country and the government in general.

These observations can be further improved using more sophisticated techniques of finding optimum number of topics using coherence values[]. This can be a further area of interest.

From the intertopic distance map following observations can be drawn: Each bubble here shows a topic. The larger the bubble, the most prevalent is that topic.hence according to our model it is topic 6. If you move the cursor over one of the bubbles, the words and bars on the right-hand side will update. These words are the salient keywords that form the selected topic. Since most of the bubbles are clustered in the 1st and 4th quadrant, it is not very good topic model.

It is very clear from the sentiment analysis that most of the speeches of Mr. Modi are positive. This also brings up a question to ask if Mr.Prime Minister's speeches really do always defend or attack but are never neutral?

Following observations can be drawn from summarization: for term frequency and text rank few sentences are carrying the same meaning. Due to lack of diversification in the method multiple sentences are conveying essentially the same information. Ordering of sentences in term frequency and text rank method is not in accordance with the speech. For example sentences of buddha ideology are spread across summary. Whereas, LSI method has given summary which has maintained the flow of speech. Top sentences are chosen from initial part of the speech and so on. When question raised by modi is includes in the summary by text rank method, it's answer is missing. In our opinion, the LSI method provides the best summary as it closely resembles a human summary.

REFERENCES

[1] https://www.kaggle.com/abhisheksjmr/speeches-modi
[2] https://trends.google.com/trends/explore?q=modigeo=US
[3] https://www.deccanherald.com/specials/10-best-initiatives-modis-712563.html
[4] https://www.narendramodi.in/category/text-speeches
[5] https://ai.stanford.edu/ ang/papers/nips01-lda.pdf https://www.analyticsvidhya.com/blog/2018/11/introduction-text-summarization-textrank-python/
[6] https://pkghosh.wordpress.com/2019/06/27/six-unsupervised-extractive-text-summarization-techniques-side-by-side/
[7] https://iq.opengenus.org/latent-semantic-analysis-for-text-summarization/
[8] https://lowerwisdom.com/create-your-own-lsa-text-summarizer-python/
[9] https://www.machinelearningplus.com/nlp/topic-modeling-gensim-python/14computemodelperplexityandcoherencescore
[10] https://dzone.com/articles/simple-text-summarizer-using-extractive-method
[11] https://medium.com/jatana/unsupervised-text-summarization-using-sentence-embeddings-adb15ce83db1
[12] https://towardsdatascience.com/tf-idf-for-document-ranking-from-scratch-in-python-on-real-world-dataset-796d339a4089
[13] https://en.m.wikipedia.org/wiki/Tf
[14] https://www.analyticsvidhya.com/blog/2018/07/hands-on-sentiment-analysis-dataset-python/
[15] https://pypi.org/project/pywaffle/

The notebook that is related to the report is attached with the report or can be accessed from this link[].