**IDENTIFYING MENTAL CONDITIONS Of COVID-19 STRESSORS BY USING TOPIC MODELLING**

### A PROJECT COMPONENT REPORT

***Submitted by***

#### MANOJ KANNAN S (Reg. No. 201904086)

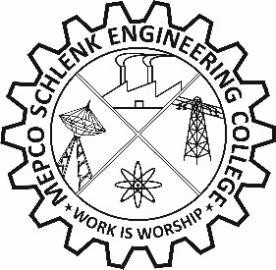
**NABIL AHAMED H (Reg. No.201904097)**

***for the Theory Cum Project Component of***

**19CS695 – SPEECH AND LANGUAGE PROCESSING**

***during***

***VI Semester – 2021 – 2022***



### DEPARTMENTOFCOMPUTER SCIENCE ANDENGINEERING MEPCO SCHLENK ENGINEERING COLLEGE, SIVAKASI

**(An Autonomous Institution affiliated to Anna University Chennai)**

### April 2022

**MEPCO SCHLENK ENGINEERING COLLEGE, SIVAKASI**

**(An Autonomous Institution affiliated to Anna University Chennai)**

### Department of Computer Science and Engineering

**BONAFIDE CERTIFICATE**

Certified that this project component report titled **IDENTIFYING MENTAL CONDITIONS Of COVID-19 STRESSORS BY USING TOPIC MODELLING** is the

bonafide work of **S.MANOJ KANNAN (Reg. No. 201904086),** and **H.NABIL AHAMED (Reg. No. 201904097)** who carried out this work under my guidance for the Theory cum Project Component course **“19CS695 – SPEECH AND LANGUAGE PROCESSING”** during the sixth semester.

#### Mrs.S.SANTHI,M.E.,(Ph.D) Dr. J. RAJA SEKAR, M.E.,Ph.D.

Assistant Professor (Sr. Grade) Professor

Course Instructor Head of the Department

Department of Computer Science & Engg. Department of Computer Science & Engg. Mepco Schlenk Engineering College Mepco Schlenk Engineering College Sivakasi. Sivakasi.

Submitted for viva-Voce Examination held at **MEPCO SCHLENK ENGINEERING COLLEGE (Autonomous), SIVAKASI** on **………/……/…. 20..........**

#### Internal Examiner - I Internal Examiner - II

i

### ACKNOWLEDGEMENT

First and foremost, we thank the **LORD ALMIGHTY** for his abundant blessings that is showered upon our past, present and future successful endeavors.

We extend our sincere gratitude to our college management and Principal **Dr.**

**S. Arivazhagan M.E., Ph.D.,** for providing sufficient working environment such as systems and library facilities. We also thank him very much for providing us with adequate lab facilities, which enable us to complete our project.

We would like to extend our heartfelt gratitude to **Dr. J. Raja Sekar M.E., Ph.D.,** Professor and Head, Department of Computer Science and Engineering, Mepco Schlenk Engineering College for giving me the golden opportunity to undertake a project of this nature and for his most valuable guidance given at every phase of our work.

We would also like to extend our gratitude and sincere thanks to **Mr.B.Lakshmanan M.Tech., (Ph.D).,** Assistant Professor (Sr. Grade), Department of Computer Science and Engineering, Mepco Schlenk Engineering College for being our Project Mentor. He has put his valuable experience and expertise in directing, suggesting and supporting us throughout the Project to bring out the best.

Our sincere thanks to our revered **faculty members and lab technicians** for their help over this project work.

Last but not least, we extend our indebtedness towards out beloved family and our friends for their support which made the project a successful one.

ii

### ABSTRACT

The Machine learning techniques is a fast growing field. Machine learning is the field of computer science that gives computer ability to learn without being explicitly programmed. Machine learning (ML) is a type of artificial intelligence (AI) that allows software applications to become more accurate at predicting outcomes without being explicitly programmed to do so. Machine learning algorithms use historical data as input to predict new output values.

The COVID-19 pandemic has affected lives of people from different countries for almost two years. The changes on lifestyles due to the pandemic may cause psychosocial stressors for individuals, and have a potential to lead to mental health problems. To provide high quality mental health supports, healthcare organization need to identify the COVID-19 specific stressors, and notice the trends of prevalence of those stressors. This study aims to apply natural language processing (NLP) on social media data to identify the psychosocial stressors during COVID-19 pandemic, and to analyse the trend on prevalence of stressors at different stages of the pandemic.

We obtained dataset of 9266 Reddit posts from subreddit \rCOVID19\_support, from 14th Feb ,2020 to 19th July 2021. This work uses Latent Dirichlet Allocation (LDA) topic model and lexicon methods to identify the topics that were mentioned on the subreddit. The result presented a dashboard to visualize the trend of prevalence of topics about covid-19 related stressors being discussed on social media platform.

The result could provide insights about the prevalence of pandemic related stressors during different stages of COVID-19. The NLP techniques leveraged in this study could also be applied to analyse event specific stressors in the future.

iii

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **CHAPTER NO.** | **TITLE** | **PAGE NO.** |
|  | **ABSTRACT** | **iii** |
|  | **LIST OF TABLES** | **vii** |
|  | **LIST OF FIGURES** | **viii** |
|  | **LIST OF ABBREVIATIONS** | **ix** |
| **1** | **INTRODUCTION** | **1** |
|  | 1.1 PURPOSE OF THE PROJECT | 2 |
|  | 1.2 OBJECTIVES | 3 |
|  | 1.3 OUTCOMES | 3 |
|  | 1.4 ORGANIZATION OF THE PROJECT | 3 |
| **2** | **LITERATURE SURVEY** | 4 |
|  | 2.1 OVERVIEW | 5 |
|  | 2.2 IMPACT OF THE COVID 19 PANDEMIC ON MENTAL HEALTH AMONG 157,213. | 5 |
|  | 2.3 DEVELOPMENT AND INITIALIZATION OF COVID STRESS SCALES | 5 |
|  | 2.4 DEEP SENTIMENT CLASSIFICATION AND TOPIC DISCOVERY ON NOVEL COVID-19 | 7 |
|  | 2.5 IMPACT OF COVID 19 AND LOCKDOWN ON  MENTAL HEALTH OF CHILDREN AND ADOLESCENTS | 7 |

|  |  |  |
| --- | --- | --- |
|  | 2.6 MENTAL HEALTH DISCOURSE ON  REDDIT:SELF DISCOURSE, SOCIAL DISCOURSE | 8 |
|  | 2.7 NATURAL LANGUAGE REVEALS VULNERABLE MENTAL HEALTH SUPPORT GROUPS AND HEIGHTENED HEALTH  ANXIETY. | 9 |
| **3** | **SYSTEM STUDY** | **11** |
|  | 3.1 OVERVIEW | 12 |
|  | 3.2 EXSISTING MODEL | 12 |
|  | 3.3 PROPOSED MODEL | 12 |
|  | 3.4 TECHNOLOGIES USED | 13 |
| **4** | **SYSTEM DESIGN** | **14** |
|  | 4.1 DATASET EXTRACTION | 15 |
|  | 4.2 DATA CLEANING | 15 |
|  | 4.3 TEXT PREPROCESSING | 16 |
|  | 4.4 FEATURE EXTRACTION | 16 |
|  | 4.5 TOPIC MODEL ALGORITHMS | **16** |
|  | 4.5.1 LDA | 16 |
|  | 4.5.2 NMF | 16 |
|  | * 1. VISUALIZATION   2. GROUPING LDA TOPICS | 17 |
| **5** | **SYSTEM IMPLEMENTATION** | **19** |
|  | 5.1 OVERVIEW | 20 |
|  | 5.2 ALGORITHM FOR LDA | 20 |

|  |  |  |
| --- | --- | --- |
|  | 5.3 ALGORITHM FOR NMF | 21 |
|  | 5.4 ALGORITHM FOR GRIDSEARCHCV | 21 |
| **6** | **RESULTS AND DISCUSSIONS** | **24** |
|  | 6.1 OVERVIEW OF THE RESULTS | 25 |
|  | 6.2 COUNT VECTORIZATION | 25 |
|  | * 1. TF-IDF VECTORIZATION   2. WORDCLOUD   3. LDA TOPIC REPRESENTATION   4. NMF TOPIC REPRESENTATION   5. GRIDSEARCHCV 6.8TOPIC ASSIGNING | 27 |
| **7** | **CONCLUSION AND FUTURE FRAMEWORKS** | **29** |
|  | 7.1 CONCLUSION | **31** |
|  | APPENDIX A | **32** |
|  | APPENDIX B |  |
|  | REFERENCES | **38** |

|  |  |  |
| --- | --- | --- |
| **Table**  **No.** | **Table Caption** | **Page**  **No.** |
| 5.3.1 | Labelling of data | 21 |
| 5.4.1 | Shows vectorized form for the data | 22 |
| 6.3.1 | Performance values for algorithm | 27 |
| 6.4.2 | Topics assigned for each of the documents | 29 |

|  |  |  |
| --- | --- | --- |
| **Figure**  **No.** | **Figure Caption** | **Page**  **No.** |
| 4.1 | Graphical view of the System | 22 |
| 6.1 | Number of symptoms for each patient | 26 |
| 6.2 | Particular symptoms for each patient | 26 |
| 6.3 | Performance values of each algorithm | 27 |

|  |  |
| --- | --- |
| **ABBREVIATION** | **DESCRIPTION** |
| LDA | Latent Dirichlet Allocation |
| NMF | Non-Matrix Factorization |
| TF-IDF | Term-Frequency Inverse Document Frequency |
| NLTK | Natural Language Tool Kit |

# INTRODUCTION

### CHAPTER 1 INTRODUCTION

#### PURPOSE OF THE PROJECT

The COVID-19 pandemic has affected lives of people from different countries for almost two years. To prevent the risk of infection, countries have implemented different safety measures such as social distancing, lockdown, and school closure. Individuals’ mental health has been impacted due to experiencing stress, anxiety, loneliness, and feeling uncertain about the pandemic. Recognizing the common psychosocial stressors and their prevalence at different stages of the pandemic will allow the healthcare providers and social workers to provide high quality mental health interventions and support. Several studies have been done to understand the impacts of COVID-19 on mental health. The majority of the studies used questionnaires or interviews to obtain data for analyzing the mental health impacts of the pandemic. However, the process of obtaining data could be time-consuming and costly. As the development of the pandemic is evolving rapidly, individual’s mental health status could also be changing quickly. The survey methods may not be able to capture the latest needs of mental health support. In contrast, social media may allow close to real-time monitoring of the mental health impacts during the pandemic, as individuals actively share their feelings and difficulties on social media platforms. In this study, Natural Language Processing (NLP) techniques are applied to identify COVID-19 related psychosocial stressors that were discussed on social media, and to visualize the prevalence of stressors at different stages of the pandemic.

Social media platforms, such as Twitter and Reddit, are commonly used as the data source for obtaining insights regarding the mental health status. As people will share their feelings or experiences on the platforms, the contents on social media may reflect users’ emotions. The changes on emotions of the population could be reflected on their behavior on social media. Many researchers have utilized NLP techniques and social media data to analyze mental health status for the population.

#### OBJECTIVES

* + - To collect the mental conditions of covid-19 stressors related reviews and comments from twitter dataset.
    - To pre-process the collected data and extract its features based on mental conditions.
    - To predict the topics about the pandemic related stressors during different stages of Covid-19 using Latent Dirichlet Allocation technique and lexicon methods.
    - To assign the topics to documents.

#### OUTCOMES

* + - Extracted features.
    - Topics identified by topic model.
    - Word cloud for each of the topics identified.
    - Classify the dominant topic among the documents.
  1. **ORGANIZATION OF THE PROJECT**
     + Chapter 2 describes about the various existing methodologies used for Identifying mental conditions
     + Chapter 3 discusses about the system study.
     + Chapter 4 outlines the design and explains the methodology of the proposed system.
     + Chapter 5 describes about the system implementation.
     + Chapter 6 discusses about the results.
     + Chapter 7 discusses about the conclusion and future enhancements.

# LITERATURE SURVEY

### CHAPTER – 2 LITERATURE SURVEY

#### Impact of the COVID-19 pandemic on mental health among 157,213 Americans

This paper represents an unpredicted crisis with potential negative mental health impacts. It uses Latent growth curve model and linear regression model to predict changes. The outcome variable is psychological distress as measured by the Kessler psychological distress scale. This uses Longitudinal Survey methods are conducted.

#### Merits:

* + - This paper uses Latent growth curve model and Latent growth linear model which has the ability to use variables simultaneously as independent and dependent variables in the same model, allowing for complex representations of growth and correlates of change.
* Other factors such as health related characteristics and personality traits, were associated with the level of distress before the pandemic.

#### Limitations:

* The system does nothing to guard against network spoofing.
* It doesn’t to distinguish ‘live’ face & ‘not live’ fact.

#### Development and initial validation of the COVID Stress Scales

Data were collected from Canada and the United States using an Internet-based self- report survey delivered in English. The self-report survey comprised measures demographics and depression experiences with covid-19. To determine whether the covid stress syndrome is dimensional or multicategorical construct, latent class analyses were conducted. This paper uses either Latent Semantic Analysis or the LDA method.

#### Merits:

* The 5-factor model, obtained in the exploratory factor analysis from the Canadian sample, could be tested in RML confirmatory factor analysis in the United States sample.
* The scales were scored by adding the unit-weighted items together. Higher scores indicate greater levels of COVID-19-related distress.
* Due to the large sample sizes, the correlations between the five scales of the CSS and social desirability were statistically significant for each country.

#### Limitation:

* There is a need to improve this accuracy by the use of other classifiers.
* In terms of limitations, the present study did not include structured diagnostic assessments (i.e., DSM-5 or ICD-11 diagnoses), which would have been useful in evaluating criterion-related (known-groups) validity of the CSS.

#### Deep Sentiment Classification and Topic Discovery on Novel Coronavirus or COVID-19 Online Discussions: NLP Using LSTM Recurrent Neural Network Approach,

This paper proposed a system for convenient channels for users concerned about healthcare issues and share information with others. They investigated how to use LSTM recurrent neural network for sentiment classification of COVID-19 comments. They used automated extraction of COVID-19-related discussions from social media and a natural language process (NLP) method based on topic modelling. This work uses automated extraction of COVID-19-related discussions from social media and a natural language process (NLP) method based on topic modelling. Experiments demonstrated that the research model achieved an accuracy of 81.15% - a higher accuracy than that of several other well- known machine-learning algorithms for COVID-19-Sentiment Classification.

#### Merits

* A higher accuracy than that of several other well-known machine-learning algorithms for COVID-19-Sentiment Classification.
* This findings shed light on the importance of using public opinions and suitable computational techniques to understand issues surrounding COVID-19 and to guide related decision-making.

#### Limitations:

* This research was limited to English-language text, which was considered a selection criterion. Therefore, the results do not reﬂect comments made in other languages.

#### Impact of COVID-19 and lockdown on mental health of children and adolescents: A narrative review with recommendations

This paper is aimed at narratively reviewing various articles related to mental-health aspects of children and adolescents impacted by COVID-19 pandemic. The Author searched the electronic data bases of MEDLINE through PubMed, Cochrane Library, Science-direct and Google Scholar databases, from January,2020 till June,2020. It uses various topic modelling algorithms Latent Semantic Allocation and Non matrix Factorization.

#### Merits:

* There is a need to ameliorate children and adolescents’ access to mental health support services geared towards providing measures for developing healthy coping mechanisms during the current crisis.
* Identify the people with the bad mental conditions and give treatment with priority

#### Limitations:

* The review articles for this review have been selected during the time of global lockdown, where the issues and challenges were new and the global crisis was at peak times.
* Due to strict selection criteria and the short period of data collection and the only use of electronic databases for this research, there is a possibility of missing studies relevant to the care of children and adolescents.

#### Mental Health Discourse on reddit: Self-disclosure, Social Support, and Anonymity

Social media is continually emerging as a platform of information exchange around health challenges. Building on findings about health information seeking and sharing practices in online forums, and social media like Twitter, we address three research challenges. Here they uses negative binomial regression as a prediction method because both the dependent variables, karma and comments are counts, and negative binomial regression is typically well-suited to handle over dispersed count outcome variables. Here they uses a measure called deviance to evaluate goodness of fit, since this model has no direct analog of the proportion of variance explained by the predictors (R 2 ) in linear regression.

#### Merits:

* This research reveals how social media like reddit are fulfilling unique information and social needs of a cohort challenged with a stigmatic health concern looking through the lenses of disclosure, social support, and disinhibition.
* Potentially, this work may provide a wealth of resources to clinicians, health practitioners, caregivers, and policy makers to identify communities at risk.

#### Limitations:

* This work is a preliminary exploration, focusing on a set of high precision reddit communities, however expanding to other subreddits is a ripe area of future research.
* Understanding the extent to which the greater reddit population engaged in mental illness discourse embodies the observed behaviour, is also valuable from a generalization perspective.

#### Natural Language Processing Reveals Vulnerable Mental Health Support Groups and Heightened Health Anxiety on Reddit During COVID-19:

The aim of this study is to leverage natural language processing (NLP) with the goal of characterizing changes in 15 of the world’s largest mental health support groups (e.g., r/schizophrenIa, r/SuicideWatch, r/Depression) found on the website Reddit. Using regression, they analyzed trends from 90 text-derived features such as sentiment analysis, personal pronouns, and semantic categories. Using supervised machine learning, this work classified posts into their respective support groups and interpreted important features to understand how different problems manifest in language. Unsupervised methods are applied such as topic modelling and unsupervised clustering throughout the reddit before and during the pandemic.

#### Merits:

* By using a broad set of NLP techniques and analyzing a baseline of prepandemic posts, the author uncovered patterns of how specific mental health problems manifest in language.

#### Limitations:

* Support group reveals the mental health conditions of all the people in the pandemic.

But fails to predict the exact number of people who are really in the height of anxiety.

# SYSTEM STUDY

### CHAPTER-3 SYSTEM STUDY

#### OVERVIEW:

The idea is to make the topics to be assigned to the respective documents posted by the people in the reddit post and identify the conditions of those people

#### EXISTING SYSTEM:

In LDA, topics are defined as a mixture of terms with different probability distribution; this means a word could belong to more than one topic and it could cause inaccuracy on the prediction. In contrast, lexical approach has higher interpretability on topic classification; but it requires careful selection of keywords to avoid the terms that could belong to more than one topic. In this study, we assumed we do not have prior knowledge on how Reddit users expressed their feelings on the subreddit. To obtain insights about what topics existed in the subreddit and what could be the keywords for each of the topics, so applied LDA model before applying the lexical approach. In this study, lexical approach was created for two purposes.

The above technique focuses on the LDA topics for the documents posted in the reddit post

#### 3.5 PROPOSED SYSTEM

In this proposed system , we applied the feature extraction methods to identify the bag of words and the frequency of words in the multiple documents posted in the reddit. Then we applied LDA and NMF modeling to apply the topic modeling. These models predict the tokens for the specified n number of components(topics) in the algorithm. We measure that LDA model is slight better than that of NMF model and then use the GridSearchCV to apply the multiple possibilities of components and learning decay to identify the best estimator and topics for the documents.

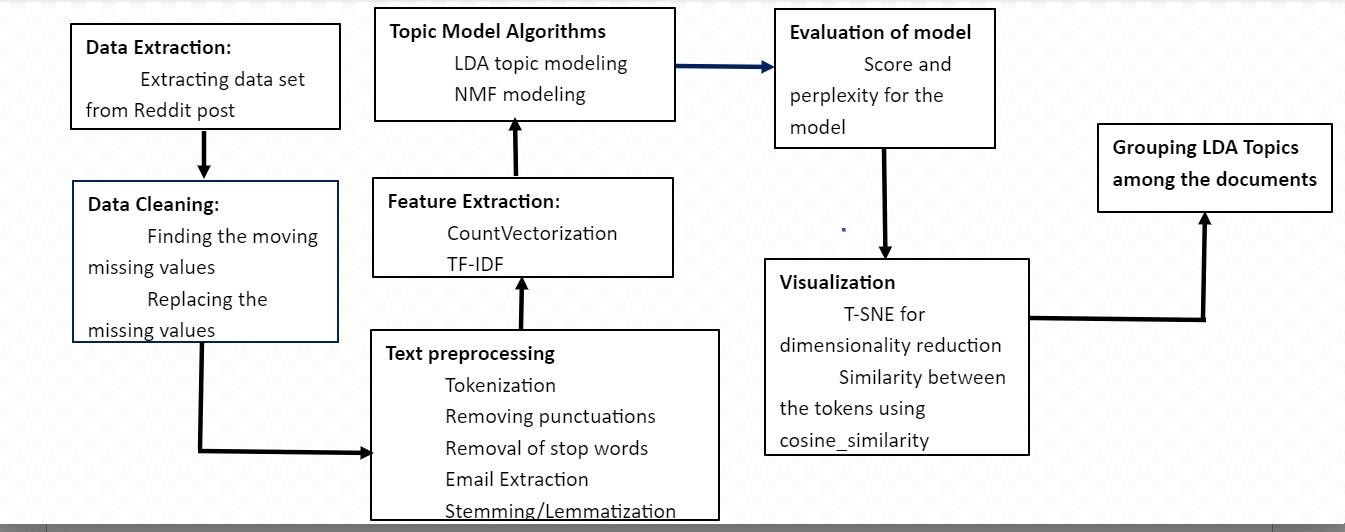
#### 3.4 TECHNOLOGIES USED

**COLAB**

Colab is a jupyter notebook run in the cloud. It is created by Google Company for people to work in complex machine learning program even in low configuration system. Colab provide python service to run.

# SYSTEM DESIGN

### CHAPTER – 4 SYSTEM DESIGN



#### Fig : 4.1.1 Overview of Research framework

The above figure shows the Iteration process of improving the topic model. By this iteration of feature selection and evaluation of topic model output, the performance of topic model was controlled.

#### DATA EXTRACTION:

Data is collected from the kaggle website. This Dataset contains the Reddit post id, tokens, text . It has about 11000 rows and 12 columns with text specified which we need to use to identify the topics for the documents.

#### DATA CLEANING:

Data cleaning involves removing the null values and NA values in the dataset and removing unwanted rows and columns in the dataset.

#### TEXT PREPROCESSING:

After cleaning process is carried out which includes removal of punctuations, tokenization, removal of stopwords, Email and website url extraction, Stemming/ Lemmatization.

#### FEATURE EXTRACTION:

Feature extraction involves CountVectorization and Tf-idf Vectorization. Count Vectorization is the bag of words indicates the count of words in the whole documents. Tf-idf vectorization is the multiplication of term frequency within the single document and inverse document frequency in overall documents.

#### TOPIC MODEL ALGORITHMS:

We applied LDA (Latent Dirichlet Allocation) and NMF (Non-matrix factorization) model to predict the topics and respective tokens and evaluate the LDA model by identifying the log likelihood and perplexity values.

#### LDA:

We used the sklearn LDA algorithm . Latent Dirichlet Allocation (LDA) is a popular topic modeling technique to extract topics from a given corpus. The term latent conveys something that exists but is not yet developed. In other words, latent means hidden or concealed. Now, the topics that we want to extract from the data are also “hidden topics”

* 1. **NMF:**

Non-negative matrix factorization (NMF or NNMF), also non-negative matrix approximation is a group of algorithms in multivariate analysis and linear algebra where a matrix V is factorized into (usually) two matrices W and H, with the property that all three matrices have no negative elements.

#### VISUALIZATION:

Similarities between the words in the tokens are visualized with th help of cosine similarity. We used the pyLDAvis tool to display the number of topics and respective tokens in the form of circle and bar chart(horizontally). We assigned the topics to the respective documents based on the domination of values.

#### GROUPING LDA TOPICS:

LDA topics are then assigned to each document based on the dominated value in the frequency (tf-idf vectorization).

**SYSTEM IMPLEMENTATION**

### CHAPTER5

**SYSTEM IMPLEMENTATION**

#### OVERVIEW:

Dataset contains the text to be processed to assign the topics and find the mental conditions of people in the pandemic period.

|  |  |
| --- | --- |
| **S. No** | **Text** |
| 1) | I think Coronavirus is good. Don't need to develop the cure, Corona is good for population control if the cure comes out then  we human will certainly way too much and cause  major problem on earth like pollution. If many of you died and only the finest ones remain... |
| 2) | When Trump almost started a World War a month ago, I couldn't help but feel excited. I know it's wrong, but I found  myself anticipating, even hoping that a |

war would start, knowing there's a good chance I could die. I know it's horrible, and the number of people that would die alongside me would be staggering, but I feel so numb to the concept of mortality when my own life feels like a joke with me as the punchline....

**Table. 5.1.1 Overview of the documents.**

This table shows the text documents posted by the people in the reddit post

* 1. **Algorithm for LDA**

Function: LDA(preprocessed\_data) INPUT://preprocessed\_data

OUTPUT://lda\_topics and tokens based on LDA model BEGIN

LDA(preprocessed\_data)

//LDA Model available in sklearn.decomposition package

lda = LatentDirichletAllocation(n\_components=12, max\_iter=5, learning\_method = 'online', learning\_offset = 50., random\_state = 0)

lda\_model = lda.fit\_transform(x)

END

## Algorithm for NMF (Non-matrix factorization)

Function: NMF(preprocessed\_data) INPUT://preprocessed\_data

OUTPUT://nmf\_topics and tokens based on NMF model BEGIN

NMF(preprocessed\_data)

//NMF Model available in sklearn.decomposition package

nmf= NMF(n\_components=12, random\_state=1,beta\_loss="kullback-leibler", solver="mu", max\_iter=1000, alpha=0.1, l1\_ratio=0.5)

nmf\_model = nmf.fit\_transform(x)

END

## Algorithm for GridSearchCV

Function: GridSearchCV(preprocessed\_data) INPUT://preprocessed\_data

OUTPUT:Best LDA model with high low perplexity and high log likelyhood BEGIN

GRID(preprocessed\_data)

search\_params ={'n\_components': [10, 15, 20, 25, 30], 'learning\_decay': [.5, .7, .9]} lda = LatentDirichletAllocation()

model = GridSearchCV(lda, param\_grid=search\_params) model.fit(x)

END

**RESULTS AND DISCUSSION**

### CHAPTER – 6 RESULT AND DISCUSSION

#### OVERVIEW:

This chapter deals with Results to be discussed in our project that has been produced.

The setup is designed and result is discussed as below.

#### COUNT VECTORIZATION:

It includes the count of each of word in the documents.

|  |  |
| --- | --- |
| **WORDS** | **COUNT** |
| Feel | 3 |
| friend | 3 |
| Parent | 3 |
| much | 2 |
| potentially | 2 |

#### Table.6.2.1 Count Table for the words

This table shows the count for each of the words in the document. It represents the bag of words.

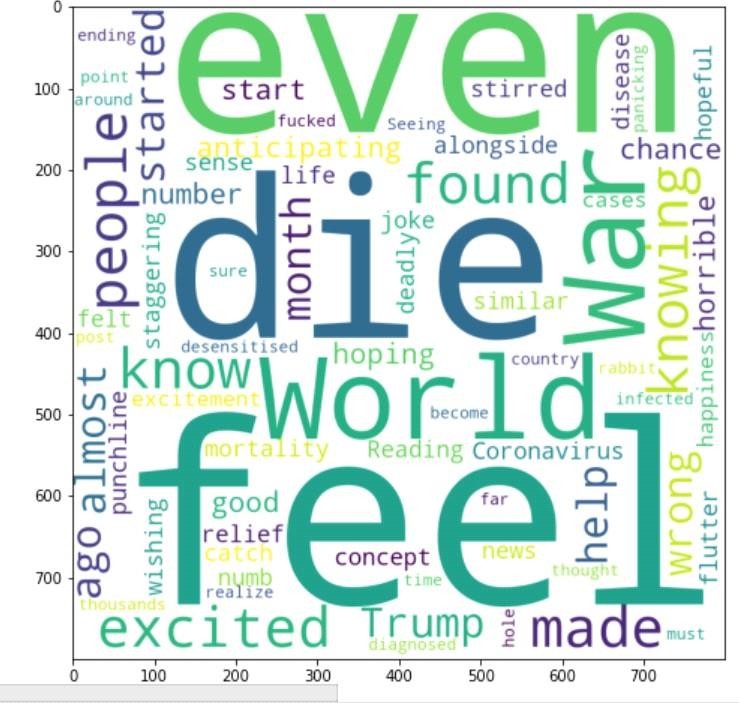
#### TF-IDF VECTORIZATION:

|  |  |
| --- | --- |
| **WORDS** | **FREQUENCY** |
| Cure | 0.467929 |
| goddent | 0.233964 |
| develop | 0.233964 |
| remain | 0.233964 |
| population | 0.233964 |

**Table.6.3.1 TF-IDF Vectorization for the words**

This table shows the frequency of each words by using the TF-IDF Vectorization.

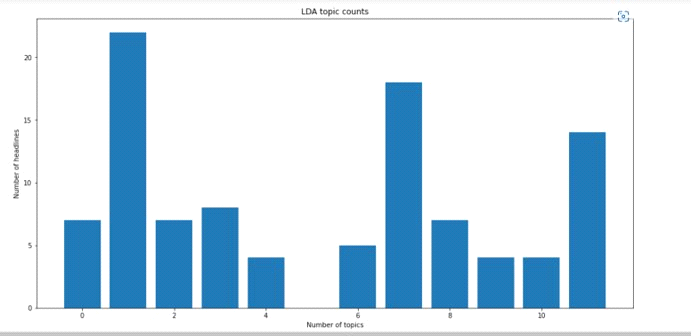
#### 6.4. WORD CLOUD



**Fig .6.4.1 Word Cloud for each word**

This figure shows word cloud for the all words occurs in dataset frequently.

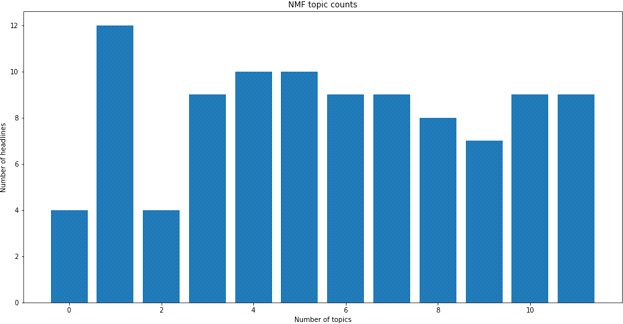
#### 6.5 LDA TOPIC REPRESENTATION:



**Fig. 6.5.1 Headlines for LDA Model**

This figure shows LDA topics and number of headlines are analyzed.

#### NMF TOPIC REPRESENTATION:



**Fig. 6.6.1 Headlines for LDA Model**

This figure shows the NMF topics and number of headlines.

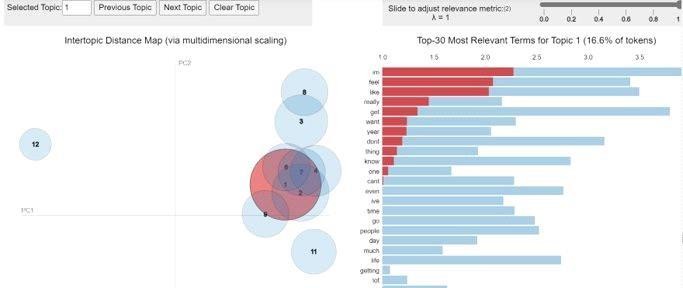
#### GRIDSEARCHCV

By applying GridSearchCV we concluded that,

Best Model param : { learning\_decay:0.9, n\_components: 10} Best log liklihood score : -3039.07127369487

Model perplexity : 118889.9760750

PyLDAvis visualization for the best LDA model



#### Fig. 6.7.1 GridSearchCV

This figure shows the GridSearchCV for visualizing the most relevant terms for

topics.

#### 6.8 TOPIC ASSIGNING:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Topi**  **c0** | **Topi**  **c1** | **Topi**  **c2** | **Topi**  **c3** | **Topi**  **c4** | **Topi**  **c5** | **Topi**  **c6** | **Topi**  **c7** | **Topi**  **c8** | **Topi**  **c9** | **dominan**  **t\_topic** |
| Do  C0 | 0.02  0000 | 0.44  0000 | 0.02  0000 | 0.02  0000 | 0.02  0000 | 0.43  0000 | 0.02  0000 | 0.02  0000 | 0.02  0000 | 0.02  0000 | 1 |
| Do  C1 | 0.01  0000 | 0.01  0000 | 0.43  0000 | 0.01  0000 | 0.01  0000 | 0.49  0000 | 0.01  0000 | 0.01  0000 | 0.01  0000 | 0.01  0000 | 5 |
| Do  C2 | 0.01  0000 | 0.39  0000 | 0.01  0000 | 0.01  0000 | 0.01  0000 | 0.52  0000 | 0.01  0000 | 0.01  0000 | 0.01  0000 | 0.01  0000 | 5 |
| Do  C3 | 0.01  0000 | 0.01  0000 | 0.01  0000 | 0.01  0000 | 0.01  0000 | 0.45  0000 | 0.47  0000 | 0.01  0000 | 0.01  0000 | 0.01  0000 | 6 |
| Do  C4 | 0.01  0000 | 0.58  0000 | 0.01  0000 | 0.01  0000 | 0.01  0000 | 0.35  0000 | 0.01  0000 | 0.01  0000 | 0.01  0000 | 0.01  0000 | 1 |

**Table. 6.8.1 Assigning Topic by Dominance**

This table shows the assigning of topics to the documents based on the dominance of the tokens.

**CONCLUSION AND FUTURE FRAMEWORKS**

### CHAPTER 7

**CONCLUSION AND FUTURE FRAMEWORK**

This chapter concludes the project report and provides scope for various enhancements that can be made to this project in the future.

#### CONCLUSION

The proposed system predicts score and perplexity for the Latent Dirichlet Allocation Technique. When comparing with the NMF model LDA model predicts the better topics and tokens for the documents.

Hence, we used GridSearchCV to find the best LDA model with high score and low perplexity. And we assign the topics to the documents based on the dominance of tokens.

* 1. **FUTURE WORK**

In future we can make sentiment analysis on each of the topics identified and make comparison with the positive, negative, neutral comments over the each of the documents specified

#### APPENDICES APPENDIX A

**HARDWARE REQUIREMENTS**

|  |  |  |
| --- | --- | --- |
| **PROCESSOR** | **:** | AMD Ryzen |
| **HARD DISK** | **:** | 1 TB Hard Disk Space |
| **RAM** | **:** | 8 GB |

**7.4 SOFTWARE REQUIREMENTS**

|  |  |  |
| --- | --- | --- |
| **LANGUAGE** | **:** | Python Programming  Language. |
| **OPERATING SYSTEMS** | **:** | Windows 11 |
| **SOFTWARE USED** | **:** | Google Colab |

### APPENDIX B

**CODING:**

from nltk.corpus import stopwords

from nltk.tokenize import word\_tokenize from nltk.tokenize import sent\_tokenize import pandas as pd

from nltk.stem import WordNetLemmatizer

from gensim.models.coherencemodel import CoherenceModel import re

#remove the punctuations and symbols

df = pd.read\_csv("posts\_data\_with\_topics\_and\_sentiments.csv"); punctuations = '''!@#$%^&\*()[].''"",'''

count1 = 0

#characters to be removed in the text count1 = df.count()

l = []

email = [] text\_with\_no\_punc = [] for i in range(0,100):

row = [] row.append(df.text[i]) l.append(row)

for j in range(0,100): no\_punc = ''

row = []

for i in l[j][0]:

if(i not in punctuations): no\_punc = no\_punc + i

row.append(no\_punc) text\_with\_no\_punc.append(row)

for j in range(0,100):

text\_with\_no\_punc[j] = word\_tokenize(text\_with\_no\_punc[j][0]) df.text[5]

#StopWords

stop\_words = stopwords.words('english') stop\_words1 = stopwords.words('english')

for i in range(0,len(stop\_words)): stop\_words[i] = stop\_words[i].capitalize()

for i in stop\_words1: stop\_words.append(i)

stop\_words.append('im')

#extracting emails

for i in range(0,len(text\_with\_no\_punc)):

for j in range(0,len(text\_with\_no\_punc[i])):

email = re.findall(r"[a-z0-9\.\-+\_]+@[a-z0-9\.\+\_]+\.[a-z]+",text\_with\_no\_punc[i][j]) text\_with\_no\_email = []

for i in range(0,len(text\_with\_no\_punc)): row = []

for j in range(0,len(text\_with\_no\_punc[i])): if text\_with\_no\_punc[i][j] not in email:

row.append(text\_with\_no\_punc[i][j]) text\_with\_no\_email.append(row)

#removing stop words for j in range(0,100):

row = []

for i in text\_with\_no\_email[j]: if(i not in stop\_words):

row.append(i) text\_with\_no\_email[j] = row

text\_with\_no\_email[0] #Stemming

#from nltk.stem import PorterStemmer #stemmer = PorterStemmer() #new\_list = []

#for i in range(0,100):

# words = []

# for word in l[i]:

# words.append(stemmer.stem(word)) # new\_list.append(words)

#new\_list[2]

#Lemmatization convert a word to a dictionary form lemmatizer = WordNetLemmatizer()

new\_list = []

for i in range(0,100): words = []

for word in text\_with\_no\_email[i]: words.append(lemmatizer.lemmatize(word))

new\_list.append(words)

#merge the words into a correct sequence for i in range(0,100):

new\_words = ''

for word in new\_list[i]:

new\_words = new\_words + word + ' ' new\_list[i] = new\_words

for i in range(0,len(new\_list)):

new\_list[i] = word\_tokenize(new\_list[i]) count\_words = 0

new\_list st1 = 'Im' st2 = 'im' st3 = 'ia'

for i in range(0,100): if('im' in new\_list[i]):

new\_list.remove(st)

if('Im' in new\_list[i]): new\_list[i].remove(st1)

if('ia' in new\_list[i]): new\_list

#Make the words lower case word = []

for i in new\_list: l = []

p = ''

for j in i:

p = p + j.lower() + ' ' l.append(p) word.append(p)

word[0]

#Count vectorization

from sklearn.feature\_extraction.text import CountVectorizer import matplotlib.pyplot as plt

sents = word

cv = CountVectorizer()

X = cv.fit\_transform(sents) df\_count = []

for i in range(0,100):

df1 = pd.DataFrame(X[i].T.todense(), index=cv.get\_feature\_names(), columns=["COUNT"])

df1 = df1.sort\_values('COUNT', ascending=False)

df\_count.append(df1) df1.iloc[2:40,:]

#Tf-idf vectorization sent2 = word

#print(cv.get\_feature\_names())

from sklearn.feature\_extraction.text import TfidfVectorizer tf = TfidfVectorizer()

x = tf.fit\_transform(sent2) x.toarray()

df2 = []

for i in range(0,100):

df3 = pd.DataFrame(x[i].T.todense(), index=tf.get\_feature\_names(), columns=["TF-IDF"])

df3 = df3.sort\_values('TF-IDF', ascending=False) df2.append(df3)

df2

from wordcloud import WordCloud #WordCloud visualization for each doucument import matplotlib.pyplot as plt

for i in range(0,10):

wordcloud = WordCloud(width = 800, height = 800, background\_color ='white',

min\_font\_size = 10).generate(df['text'][i]) plt.figure(figsize = (8, 8), facecolor = None) plt.imshow(wordcloud)

#Latent Dirichilet Allocation

from sklearn.decomposition import LatentDirichletAllocation lda = LatentDirichletAllocation(n\_components=12, max\_iter=5,

learning\_method = 'online', learning\_offset = 50.,

random\_state = 0)

lda\_model = lda.fit\_transform(x)

def print\_top\_words(model, feature\_names, n\_top\_words): fig, axes = plt.subplots(2, 6, figsize=(30, 15), sharex=True) axes = axes.flatten()

l = []

for topic\_idx, topic in enumerate(model.components\_): print("Topic #%d:" % topic\_idx)

print(" ".join([feature\_names[i]

for i in topic.argsort()[:-n\_top\_words - 1:-1]])) st = ([feature\_names[i]

for i in topic.argsort()[:-n\_top\_words - 1:-1]]) l.append(st)

top\_features\_ind = topic.argsort()[: -n\_top\_words - 1 : -1] top\_features = [feature\_names[i] for i in top\_features\_ind] weights = topic[top\_features\_ind]

ax = axes[topic\_idx]

ax.barh(top\_features, weights, height=0.8)

ax.set\_title(f"Topic {topic\_idx +1}", fontdict={"fontsize": 30})

ax.invert\_yaxis()

ax.tick\_params(axis="both", which="major", labelsize=20) for i in "top right left".split():

ax.spines[i].set\_visible(False) fig.suptitle("title", fontsize=40)

return l

#print(model.components\_) print()

n\_top\_words = 15 print("\nTopics in LDA model: ")

tf\_feature\_names = tf.get\_feature\_names() print()

l = print\_top\_words(lda, tf\_feature\_names, n\_top\_words) print()

print(l) print()

print(lda.components\_)

#T-SNE for dimensionality reduction and display topics in 2-d format from sklearn.manifold import TSNE

tsne\_lda\_model = TSNE(n\_components=3, perplexity=50, learning\_rate=100, n\_iter=2000, verbose=1, random\_state=0, angle=0.75)

tsne\_lda\_vectors = tsne\_lda\_model.fit\_transform(lda\_model) tsne\_lda\_vectors.shape

plt.scatter(x=tsne\_lda\_vectors[:,0], y=tsne\_lda\_vectors[:,1],color = 'blue')

#DISPLAYING THE LDA TOPIC COUNTS

count\_pairs = ''

from collections import Counter

keys = lda\_model.argmax(axis=1).tolist() count\_pairs = Counter(keys).items() categories = [pair[0] for pair in count\_pairs] counts = [pair[1] for pair in count\_pairs] fig, ax = plt.subplots(figsize=(16,8)) ax.bar(categories, counts); #ax.set\_xticks(categories); #ax.set\_xticklabels(labels); ax.set\_title('LDA topic counts'); ax.set\_ylabel('Number of headlines'); ax.set\_xlabel('Number of topics') print("LOG LIKELIHOOD :",lda.score(x)) print("PERPLEXITY :",lda.perplexity(x)) #NON-MATRIX FACTORIZATION

from sklearn.decomposition import NMF nmf = NMF(

n\_components=12, random\_state=1, beta\_loss="kullback-leibler", solver="mu", max\_iter=1000,

alpha=0.1,

l1\_ratio=0.5,

)

nmf\_model = nmf.fit\_transform(x) n\_top\_words = 15

print("\nTopics in NMF model: ") tf\_feature\_names = tf.get\_feature\_names() print()

print\_top\_words(nmf, tf\_feature\_names, n\_top\_words) print()

#Visualizing similarities between the tokens in the topics based on LDA model from sklearn.metrics.pairwise import cosine\_similarity

cosine\_sin = cosine\_similarity(lda\_model) cosine\_sin

cos = pd.DataFrame(cosine\_sin) import matplotlib.pyplot as plt import seaborn as sns

fig,ax = plt.subplots(figsize=(20,5)) for i in range(0,10):

sns.distplot(cosine\_sin[i],color='blue',hist=False) import pyLDAvis.gensim\_models

import pickle pyLDAvis.enable\_notebook import pyLDAvis.sklearn

g = pyLDAvis.sklearn.prepare(lda,x,tf) pyLDAvis.display(g)

pyLDAvis.enable\_notebook

g = pyLDAvis.sklearn.prepare(nmf,x,tf) pyLDAvis.display(g)

from sklearn.model\_selection import GridSearchCV

search\_params = {'n\_components': [10, 15, 20, 25, 30], 'learning\_decay': [.5, .7, .9]}

# Init the Model

lda = LatentDirichletAllocation()

# Init Grid Search Class

model = GridSearchCV(lda, param\_grid=search\_params) # Do the Grid Search

model.fit(x) # Best Model

best\_lda\_model = model.best\_estimator\_ # Model Parameters

print("Best Model's Params: ", model.best\_params\_) # Log Likelihood Score

print("Best Log Likelihood Score: ", model.best\_score\_) # Perplexity

print("Model Perplexity: ", best\_lda\_model.perplexity(x)) import numpy as np

lda\_output = best\_lda\_model.transform(x)

topicnames = ["Topic" + str(i) for i in range(best\_lda\_model.n\_components)] docnames = ["Doc" + str(i) for i in range(len(new\_list))]

df\_document\_topic = pd.DataFrame(np.round(lda\_output, 2), columns=topicnames, index=docnames)

dominant\_topic = np.argmax(df\_document\_topic.values, axis=1) df\_document\_topic['dominant\_topic'] = dominant\_topic #Grouping the topics on each document

def color\_green(val):

color = 'green' if val > .1 else 'black' return 'color: {col}'.format(col=color)

def make\_bold(val):

weight = 700 if val > .1 else 400

return 'font-weight: {weight}'.format(weight=weight) # Apply Style

df\_document\_topics = df\_document\_topic.head(110).style.applymap(color\_green).applymap(make\_bold) df\_document\_topics

### REFERENCES

1. J.S.Yarrington et al, and J. Affect. Disord, ”Impact of the COVID-19 pandemic on mental health among 157,213 Americans” ,Journal of Affective Disorders, vol. 286, pp. 64–70, 2021.
2. S. Taylor, C. A. Landry, M. M. Paluszek, T. A. Fergus, D. McKay, and G. J. G. Asmundson,” Development and initial validation of the COVID Stress Scales”, Journal of Anxiety Disorders, vol. 72, p. 102232, May 2020.
3. H. Jelodar, Y. Wang, R. Orji, and S. Huang, “Deep Sentiment Classification and Topic Discovery on Novel Coronavirus or COVID-19 Online Discussions: NLP Using LSTM Recurrent Neural Network Approach”, Procedia Computer Science, vol. 24, no. 10, pp. 2733–2742, Oct. 2020
4. S. Singh, D. Roy, K. Sinha, S. Parveen, G. Sharma, and G. Joshi, “Impact of COVID-19 and lockdown on mental health of children and adolescents: A narrative review with recommendations”, Sage Open Medicine, vol. 293, p. 113429, Nov. 2020.
5. M. D. Choudhury and S. De, “Mental Health Discourse on reddit: Self-disclosure, Social Support, and Anonymity”, Computer Science Bibliography, 2021.
6. D. M. Low, L. Rumker, T. Talkar, J. Torous, G. Cecchi, and S. S. Ghosh, “Natural Language Processing Reveals Vulnerable Mental Health Support Groups and Heightened Health Anxiety on Reddit During COVID-19”, Journal of Medical internet research, 2020.