Text Mining Analysis On COVID Dataset

Yamini Chitikela
dept. of Computer Science
Montclair State university
Montclair, NJ
chitikelay1@montclair.edu

Chaitanya Yadavally
dept. of Computer Science
Montclair State university
Montclair,NJ
yadavallyc1@montclair.edu

Rishika Devaragatla
dept. of Computer Science
Montclair State University
Montclair, NJ
devaragatlar1@montclair.edu

Sravanthi Shyamreddy
dept. of Computer Science
Montclair State university
Montclair, NJ
shyamreddys1@montclair.edu

Saidi Reddy Chennu
dept. of Computer Science
Montclair State university
Montclair, NJ
chennus1@montclair.edu

Abstract*—Text mining is gaining useful information from text by Artificial Intelligence (AI). It uses NLP, which is short for Natural Language Processing, to convert unstructured data into structured data. This is needed for analyzing and for machine learning (ML) algorithms.It is used by many different types of companies for all sorts of purposes, such as analyzing, research, decision-making based on data etc.This incredible rate of scientific productivity leads to information overload, making it difficult for researchers, clinicians and public health officials to keep up with the latest findings. Automated text mining techniques for searching, reading and summarizing papers are helpful for addressing information overload. In this review,we describe the many resources that have been introduced to support text mining applications over the COVID-19 literature;specifically, we discuss the corpora, modeling resources, systems and shared tasks that have been introduced for COVID-19.In this paper we provide a brief background and literature survey concerned with text mining. Along with an implementation and explanation of the text mining algorithm on covid data. We used a data set from covid website directory to demonstrate and examine how this algorithm performs.

***Keywords*—** *Information management, Data mining, Text mining, Web mining,COVID-19; text mining; natural language processing; information retrieval; information extraction; question answering; summarization; shared tasks*

Introduction

Since the year COVID-19 began, more than 50000 publications have been published regarding COVID-19, and each day, several hundred more are released. Due to information overload brought on by this astounding amount of scientific output, it is challenging for academics, doctors, and public health authorities to stay up to date with the most recent results. Text mining techniques for summarizing, searching, and reading are useful for information overload.

Extraction of useful information and intricate patterns from huge text collections is known as text mining. There are several techniques and tools for text mining, which may be used to generate predictions and make choices. The choice of an accurate and appropriate text mining approach affects speed and time complexity. This article analyzes and briefly discusses text mining and its applications in a variety of fields.

The major steps involved in text mining are gathering together unstructured data from several sources that are available in various document formats, such as plain text, web pages, PDF files, etc. Pre-processing and data purification techniques are used to find and eliminate discrepancies in the data. The data purification step makes sure that the original text is recorded in order to prevent word stemming. The data gathering, processing, and controlling processes are used to review and further clean the data. Pattern analysis is employed by the Management Information System.

A sound and practical decision-making process as well as trend analysis are conducted using the information derived from the information processed in the aforementioned stages.

Since the start of COVID-19, there have been between 55 and 100 000 releases. The information deluge brought on by this substantially accelerated pace of scientific activity makes it difficult for academics, medical professionals, and public health authorities to keep up with the most current findings. There has been an unusually large volume of study on COVID-19 and coronaviruses, placing a significant load on medical professionals, researchers, and others who must keep up with this new information.

Finding efficient therapies for COVID-19 is one of the most important problems. Numerous studies describing various treatment strategies have been published since the virus started to spread. The issue statement and its resolution via text mining are presented in this work. As previously said, the researchers have found it challenging to weed through all the accessible resources to get the necessary information due to the information overload. Finding the most important information, not all the stuff, is the problem statement. Building a system that enables users to search the database to find the most pertinent material is the solution to this concern. This work discusses a sentiment model that, given the names of well-known COVID-19 therapies, gives the sentiment of the various papers which are related to these treatments.

The dataset consists of various research papers compiled from many different source which includes, CORD-19 - compiles articles from many sources, WHO(World health organization), bioRxIV, medRxIV which are originally in the format of metadata and files. The data is harmonized/parsed in to an structured JSON format. The data should be comprehensive, up-to-date, clean metadata for accessing within the dataset, clear provenance that describes where the paper is extracted from which enables text mining and information retrieved on full text.

Using text mining techniques, we will address the problem's foundations, as well as the context of its application and the problem's foundation. A brief literature review of many works addressing text mining techniques and application is also included, Additionally, there are areas that can be improved. In addition, we used text mining techniques to address the problem statement. We exhibited the implementation and discussed its performance, benefits, and drawbacks during the experiment.

I. Related work

The problem of information overload is widely discussed and highly regarded among researchers in literature. Perhaps due to the wide range of applications and approaches that can be used to tackle information overload. Many publications survey such approaches and provide the readers with sufficient background to fathom the concept of text similarity and its importance. While other published work focus is on providing enhanced approaches to address a certain area or an application where filtering relevant information is of essential value. In this section we provide a brief survey of related work describing the use of data mining to determine the filtering of relevant content.

In this work conducted by Reina, Yusuke, Aln, the survey assess general japanese population thoughts on coronavirus disease 2019 related discrimination by tweets. The collected data from twitter was tokenized with a japanese-specific tokenizer package available in R called "RMelab". Independent nouns, and adverbs were extracted to be visualized in word clouds. Data collection and subsequent analysis was performed using R Version 4.0.2. Text mining is an emerging method to semi-quantitatively analyze data by utilizing text mining techniques on tweet,Tweets were retrieved from search query using keyword "health care providers and discrimination" and "carona and rural area".The top 20 common words with the frequencies are identified. The result of this study gives 51,906 tweets for "corona and healthcare", 59560 for "corona and rural" between July 29 2020 and September 30 2020. Most common words from tokenized data were translated to english. This is the first english language study to review Covid-19 related discrimination in Japan using text mining. This analysis reveals people fear being ostracized, where this study may serve as the first step towards improvement of social acceptance in japan.

From the work of Patrica Carracedo, Rosa Puertas, Luisa Marti, the goal in this paper was to discover current research lines established around COVID-19 and their influence on the corporate environment. They employed text mining methods and statistical software 'R' to do this. The capabilities of 'R' have enabled the creation and examination of the formed corpus. To that purpose, they created a term matrix of dimension 'nxt,' where n represents the number of research publications (rows) and t represents the number of unique words (columns). Each cell reflects the term's absolute frequency. When the most common words were identified in one example, a hierarchical cluster analysis was used. The first cluster is made up of a single research paper, accounting for 6.25% of all publications examined. It is distinct from the rest of the documents in the sample.

In this work from Miftahul Qorib, Timothy Oladumni, Max Denis, Esther Ososanya, Paul Cotae, the tweets such as individual opinions on the COVID-19 vaccine hesitancy were extracted from twitter via Application Programming Interface. In the first stages of the research the data was collected from twitter daily and was combined as a data set. In the early stages of the research data preprocessing was used and cleaned the data such as removing retweets and URLs from the dataset from twitter. Once the preprocessing stage is done they computed sentiment scores of the dataset using three sentiment computation methods (Azure machine learning, VADER, TextBlob). They have built 42 different models to determine the best model for classifying the dataset from twitter into positive, negative or neutral. When they trained the dataset from twitter using Logistic Regression, Support Vector Machine , decision tree, TF-IDF vectorization. TF - IDF vectorization had the highest accuracy as 93.15. The decision tree and support vector machine were the next regression models that

also received good accuracy but the decision tree was faster as compared to the support vector machine. The result of our study shows that COVID-19 vaccine hesitancy is gradually decreasing overtime, suggesting that societies positive opinions on getting vaccinated have gradually increased.This suggests that there is a positive feeling about COVID-19 vaccination.

The paper by Kyle Lo and Lucy Lu Wang describes the many resources that have been introduced to support text mining applications over the COVID-19 literature. Specifically, it gives an overview of the corpora, modeling resources, systems, and shared tasks that have been introduced for COVID-19. This work was conducted on the COVID-19 dataset dealing with text mining approaches. In this study, we concentrate on strategies for dealing with information overload and use the term "text mining" as a catch-all for techniques from the aforementioned fields. In this instance, we concentrate on corpora containing scientific articles. Corpora are collections of documents that have been text mined and have been preprocessed to extract machine-readable content. Text mining professionals may integrate resources for modeling into production systems, which include items like text embeddings, data annotations, pretrained language models,knowledge graphs and more. Systems are applications that employ user interfaces and text mining methods to deliver features like the capacity to search for, find, or see article content. Shared tasks are group contests that encourage focused effort on certain scientific issues. Over the past few decades, text mining techniques have dramatically improved. We have the chance to test these techniques with COVID-19 in the kind of time- and resource-constrained environment where automation or computational aid may be most beneficial. Initial findings are encouraging. Two shared projects have been accomplished with more in the works since early March, and biological specialists have been hired to examine and evaluate many of the systems and tools that have been developed and distributed since that time. This system was mainly developed to help researchers manage information overload and further works are planned to provide meaningful and actionable results in the fight against COVID-19.

II. Background

In this section we summarize definitions of essential concepts regarding our research

A. Data Cleaning - is the process of detecting and correcting (or removing) corrupt or inaccurate records from a record set, table, or database and refers to identifying incomplete, incorrect, inaccurate or irrelevant parts of the data and then replacing, modifying, or deleting the dirty or coarse data.[1] Data cleansing may be performed interactively with data wrangling tools, or as batch processing through scripting or a data quality firewall.

B. Data Extraction - Data extraction is the process of collecting or retrieving disparate types of data from a variety of sources, many of which may be poorly organized or completely unstructured. Data extraction makes it possible to consolidate, process, and refine data so that it can be stored in a centralized location in order to be transformed. These locations may be on-site, cloud-based, or a hybrid of the two.

Structured data Extraction - Structured data refers to data formatted according to standardized models, making it ready for analysis. It can be extracted via a relatively straightforward method known as logical data extraction. Structured data extraction is itself broken down into two subtypes, i.e., full and incremental extraction.

C. Programming

VADER Sentiment Analysis - VADER (Valence Aware Dictionary for sEntiment Reasoning) is a model used for text sentiment analysis that is sensitive to both polarity (positive/negative) and intensity (strength) of emotion. Introduced in 2014, VADER text sentiment analysis uses a human-centric approach, combining qualitative analysis and empirical validation by using human raters and the wisdom of the crowd. Consider the following sentences: “The party is wonderful” and “I hate that man.”

Do you get a sense of the feelings that these sentences imply? The first one clearly conveys positive emotion, whereas the second conveys negative emotion. Humans associate words, phrases, and sentences with emotion. The field of Text Sentiment Analysis attempts to use computational algorithms in order to decode and quantify the emotion contained in media such as text, audio, and video.Text Sentiment Analysis is a really big field with a lot of academic literature behind it. However, its tools really just boil down to two approaches: the lexical approach and the machine learning approach.

```
# import SentimentIntensityAnalyzer class
# from vaderSentiment.vaderSentiment module.
from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer

# function to print sentiments
# of the sentence.
def sentiment_scores(sentence):

    # Create a SentimentIntensityAnalyzer object.
    sid_obj = SentimentIntensityAnalyzer()

    # polarity_scores method of SentimentIntensityAnalyzer
    # object gives a sentiment dictionary.
    # which contains pos, neg, neu, and compound scores.
    sentiment_dict = sid_obj.polarity_scores(sentence)

    print("Overall sentiment dictionary is : ", sentiment_dict)
    print("sentence was rated as ", sentiment_dict['neg']*100, "% Negative")
    print("sentence was rated as ", sentiment_dict['neu']*100, "% Neutral")
    print("sentence was rated as ", sentiment_dict['pos']*100, "% Positive")

    print("Sentence Overall Rated As", end = " ")

    # decide sentiment as positive, negative and neutral
    if sentiment_dict['compound'] >= 0.05 :
        print("Positive")

    elif sentiment_dict['compound'] <= - 0.05 :
        print("Negative")

    else :
        print("Neutral")


# Driver code
if __name__ == "__main__" :

    print("\n1st statement :")
    sentence = "Geeks For Geeks is the best portal for \
# Driver code
if __name__ == "__main__" :

    print("\n1st statement :")
    sentence = "Geeks For Geeks is the best portal for \
                the computer science engineering students."

    # function calling
    sentiment_scores(sentence)

    print("\n2nd Statement :")
    sentence = "study is going on as usual"
    sentiment_scores(sentence)

    print("\n3rd Statement :")
    sentence = "I am very sad today."
    sentiment_scores(sentence)
```

III. APPROACH AND IMPLEMENTATION

The purpose of our study was to examine research papers on potential treatments for the COVID-19 pandemic and evaluate the viability of the therapies using sentiment analysis on the text's tone to determine if they were effective or not. The study article abstracts showed a wide range of outcomes, ranging from around -0.95 to 0.95. (where 0-1 is a good feeling and -1 to 0 is a negative attitude). Our research revealed that zanamivir, interferon, convalescent plasma, and remdesivir all had favorable effects. Our programme may be used to thoroughly scan upcoming documents in order to hunt for potential solutions to this problem.

The implementation is comprised of the following steps:

- filtering papers specific to COVID-19
- adding popular COVID-19 treatments
- filtering papers that mention the COVID-19 treatments
- initializing VADER sentiment model, adding sentiment scores based on title and abstract
- scanning the abstracts and displaying the treatments that were in the papers with the highest compound polarity score

A. Description of the algorithm

VADER (Valence Aware Dictionary for Sentiment Reasoning) is a model used for text sentiment analysis that is sensitive to both polarity (positive/negative) and intensity (strength) of emotion. Introduced in 2014, VADER text sentiment analysis uses a human-centric approach, combining qualitative analysis and empirical validation by using human raters and the wisdom of the crowd.

VADER sentiment analysis calculates the sentiment score of an input text. It combines a dictionary, which maps lexical features to emotion intensity, and five simple heuristics, which encode how contextual elements increment, decrement, or negate the sentiment of text.

B. Implementation

1. Data is first retrieved and added into the dataframe.

```
# df = pd.read_csv("data/metadata.csv")
df = pd.read_csv(f'{datadir}/{metadata}', na_filter= False)
corona, covid19_counts = count_and_tag(df, COVID19_SYNONYMS, 'disease_covid19')
df
```

```
Added tag_disease_covid19 to DataFrame
```

2. Filtering papers based on the COVID-19 treatments

```
corona_filtered = corona[corona.loc[:,"tag_disease_covid19"] == True]
corona_filtered
```

```
covid19_counts.sort_values(ascending=False)
```

3. Adding Popular COVID-19 treatments

```
']:
treatment_names = [
'remdesivir',
'kaletra',
'actemra',
'kevzara',
'convalescent plasma',
'avigan',
'favilavir',
'tjm2',
'medicago',
'at-100',
'tzls-501',
'oya1',
'bpi-002',
'ino-4800',
```

4. Filtering papers that mention COVID-19 treatment.

```
treatments, treatments_counts = count_and_tag(corona_filtered, treatment_names, 'treatments_covid19')
```

```
treatments_filtered = treatments[treatments.loc[:,"tag_treatments_covid19"] == True]
treatments_filtered2 = treatments_filtered[treatments_filtered['abstract'].notna()]
# titles = treatments_filtered.loc[:,"title"]
titles = treatments_filtered2.loc[:,"abstract"]
```

5. Initializing VADER sentiment model, adding sentiment scores based on title and abstract

```
sid = SentimentIntensityAnalyzer()
```

```
treatments_filtered2["scores"] = treatments_filtered2["abstract"].apply(lambda review: sid.polarity_scores(review))
treatments_filtered2["compound"] = treatments_filtered2["scores"].apply(lambda d:d["compound"])
treatments_filtered2["comp_score"] = treatments_filtered2["compound"].apply(lambda score: 'pos' if  score >=0 else 'neg')
```

```
treatments_filtered2
```

```
pd.set_option('display.max_rows', 500)
pd.set_option('display.max_columns', 500)
pd.set_option('display.width', 1000)
```

6. scanning the abstracts and displaying the treatments that were in the papers with the highest compound polarity score

```
abstracts = treatments_filtered2.sort_values("compound", ascending = False).loc[:,"abstract"]
for ab in abstracts:
    found_words = []
    for word in treatment_names:
        if word in ab:
            found_words.append(word)
    print(found_words)
```

```
['convalescent plasma']
```

IV. RESULTS & DISCUSSIONS

We had a wide range of results in the study article abstracts, from around -0.95 to 0.95. (-1 to 0 being a negative sentiment and 0-1 being a positive sentiment). Remdesivir, convalescent plasma, interferon, and zanamivir were the therapies we discovered to have a favorable attitude. Our tool may be used to scan future documents on a wide scale in order to look for potential remedies for this dilemma.

Plot depicting the results which is plotted between the number of papers and sentiment. The sentiment of positive is way more high than negative according to the research.

Number of positive/negative papers:

```
bars = treatments_filtered2.loc[:,"comp_score"]
pos_height = len(treatments_filtered2[treatments_filtered2.loc[:,"comp_score"] == "pos"])
neg_height = len(treatments_filtered2[treatments_filtered2.loc[:,"comp_score"] == "neg"])
plt.bar(["positive", "negative"], [pos_height, neg_height])
# Add some text for labels, title and custom x-axis tick labels, etc.
plt.ylabel('Number of papers')
plt.yticks(np.arange(0,22, 2))
plt.xlabel('Sentiment')
plt.title('Sentiment of papers related to COVID-19')
```

```
Text(0.5, 1.0, 'Sentiment of papers related to COVID-19')
```

V. DISCUSSION

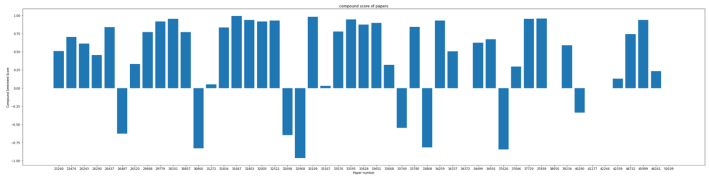
Observations and Results :

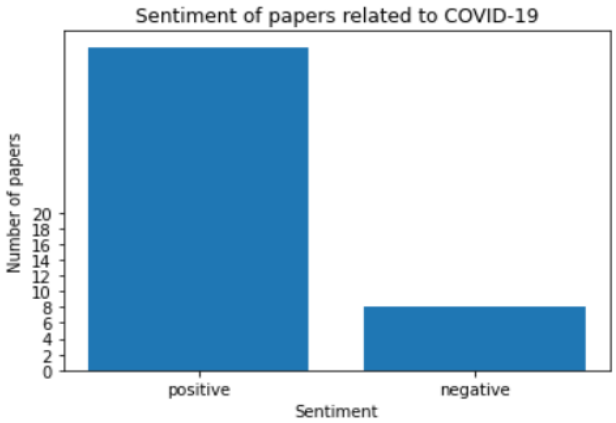
We had a wide range of results in the study article abstracts, from around -0.95 to 0.95. (-1 to 0 being a negative sentiment and 0-1 being a positive sentiment). Remdesivir, convalescent plasma, interferon, and zanamivir were the therapies we discovered to have a favorable attitude. Our tool may be used to scan future documents on a wide scale in order to look for potential remedies for this dilemma.

The compound score is the sum of positive, negative and neutral scores which is then normalized between -1(most extreme negative) and +1(most extreme positive). The more compound score closer to +1, the higher the positivity of the text. We had a wide range of outputs across the abstracts of the research papers, going from about -0.95 to 0.95 (-1 to 0 being a negative sentiment and 0-1 being a positive sentiment). The treatments we found to have a positive sentiment were remdesivir, convalescent plasma, interferon-β, and zanamivir.

```
paper_numbers = treatments_filtered2.index
x = np.array(paper_numbers)
paper_numbers_string = []
for i in x:
    paper_numbers_string.append(str(i))
scores = treatments_filtered2.loc[:,"compound"]
figure(figsize=(40,10))
plt.bar(paper_numbers_string, scores)
plt.ylabel('Compound Sentiment Score')
plt.xlabel('Paper number')
plt.title('compound score of papers')
```

```
Text(0.5, 1.0, 'compound score of papers')
```





VI. Conclusion and Future work

The goal of this study is to check the validity of different Covid 19 treatments that were suggested during the initial stage of outbreak by scanning through research papers related to Covid 19. This study was done using sentiment analysis by determining the tone of the paper and concluding whether the treatment was success or a failure. remdesivir, convalescent plasma, interferon-β, and zanamivir had a positive sentiment. Our study is limited to the title and abstract only. In future we would like to consider the entire text for further analysis by doing so it would give deeper classification to check if the treatment prescribed was efficient or not. The number of uploads are increasing at a drastic rate every day. The major work that could be worked on is the content improvement, improving the updates from weekly to daily. New features such as displaying citations, tables as well while filtering data. Moreover, a system could be built with an option to interact with users based on filtering the information.

Papers Cited:

[1]. Research lines on the impact of the COVID-19 pandemic on business. A text mining analysis

[2]. Covid-19 vaccine hesitancy: Text mining, sentiment analysis and machine learning on COVID-19 vaccination Twitter dataset.

[3]. COVID-19 related discrimination in Japan
A preliminary analysis utilizing text-mining

[4]. Text mining approaches for dealing with the rapidly expanding literature on COVID-19.

VII. REFERENCES

[1]. https://www.geeksforgeeks.org

[2].Research lines on the impact of the COVID-19 pandemic on business. A text mining analysis Patricia Carracedo a,*, Rosa Puertas b, Luisa Martí ba Universidad Internacional de Valencia, Area ´ de empresa, c/Pintor Sorolla, 21, Valencia 46022, Spain b Universitat Polit`ecnica de Val`encia, Departamento de Economía y Ciencias Sociales, Camino de Vera, s/n 46002 Valencia Spain

[3].Covid-19 vaccine hesitancy: Text mining, sentiment analysis and machine learning on COVID-19 vaccination Twitter dataset

[4].COVID-19 related discrimination in Japan A preliminary analysis utilizing text-mining

[5]. Text mining approaches for dealing with the rapidly expanding literature on COVID-19

[6]. https://www.investopedia.com/terms/d/datamining.asp

[7]. https://blog.hubspot.com/website/data-mining

[8]. https://www.geeksforgeeks.org/text-mining-in-data-mining/

[9]. https://www.simplilearn.com/what-is-text-mining-in-data-mining-article

[10]. https://www.youtube.com/watch?v=JB8khWIKtV0

[11].Dataset:https://www.kaggle.com/datasets/allen-institute-for-ai/CORD-19-research-challenge?resource=download&select=metadata.csv