



**CLASSIFICATION AND KEYWORD IDENTIFICATION OF COVID-19  
MISINFORMATION ON SOCIAL MEDIA: A FRAMEWORK FOR SEMANTIC  
ANALYSIS**

THESIS

Grace Y. Smith, First Lieutenant, USAF

AFIT-ENC-MS-22-M-002

**DEPARTMENT OF THE AIR FORCE  
AIR UNIVERSITY**

**AIR FORCE INSTITUTE OF TECHNOLOGY**

**Wright-Patterson Air Force Base, Ohio**

**DISTRIBUTION STATEMENT A.**

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the United States Government. This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

AFIT-ENC-MS-22-M-002

CLASSIFICATION AND KEYWORD IDENTIFICATION OF COVID-19  
MISINFORMATION ON SOCIAL MEDIA: A FRAMEWORK FOR SEMANTIC  
ANALYSIS

THESIS

Presented to the Faculty

Department of Mathematics and Statistics

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the  
Degree of Master of Science in Applied Mathematics

Grace Y. Smith, BS

First Lieutenant, USAF

March 2022

**DISTRIBUTION STATEMENT A.**  
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

CLASSIFICATION AND KEYWORD IDENTIFICATION OF COVID-19  
MISINFORMATION ON SOCIAL MEDIA: A FRAMEWORK FOR SEMANTIC  
ANALYSIS

Grace Y. Smith, BS

First Lieutenant, USAF

Committee Membership:

Christine M. Schubert Kabban, PhD  
Chair

Kenneth M. Hopkinson, PhD  
Member

Huaining Cheng, PhD  
Member

Mark E. Oxley, PhD  
Member

## **Abstract**

The growing surge of misinformation among COVID-19 communication can pose great hindrance to truth, magnify distrust in policy makers and/or degrade authorities' credibility, and it can even harm public health. Classification of textual context on social media data relating to COVID-19 is an effective tool to combat misinformation on social media platforms. In this research, Twitter data was leveraged to (1) develop classification methods to detect misinformation and identify Tweet sentiment with respect to COVID-19 and (2) develop a human-in-the-loop interactive framework to enable identification of keywords associated with social context, here, being misinformation regarding COVID-19. (1) Six fusion-based classification models were built fusing three classical machine learning algorithms. The best performing models were selected to detect misinformation and to classify sentiment. We found the public reacted more positively towards COVID-19 misinformation and positive sentiment increased in August 2020 relative to April 2020 for all but political or biased related misinformation. (2) The most semantically similar keywords were chosen via distribution representations of topics and recommended by optimal ROC curves. The interactive framework recommended 21 and 22 keywords related to conspiracy and unreliable misinformation, respectively, and are most semantically similar to the user inquiry "COVID start lab."

AFIT-ENC-MS-22-M-002

*To My Husband and Daughter*

## **Acknowledgments**

“Ask, and it will be given to you; seek, and you will find; knock, and it will be opened to you. [Mathew 7:7]

I asked for a degree, and my Lord Jesus Christ answered with AFIT. But He didn't just send me to AFIT bare handed, rather, as He did for all other aspects of my life, He has planned and prepared ahead of time everything that I will need to get to the other side. After 18 long months of training, I am standing on the other side, feeling extremely humble and increasingly reverent for my God. Looking back at my AFIT journey, there are so many kind souls for whom I will forever be grateful. First, my deepest gratitude goes to my committee.

My sponsor, Dr. Cheng, graciously offered a joint research opportunity for me to explore and ultimately turn the research into my thesis, months before I even started my AFIT journey. He has been supportive by all means I could possibly imagine. He is even more understanding than I could ever ask of him. When from time to time I had to tell him that the project is above my intelligence, he patiently guided me towards other obtainable yet challenging research methods. His passion about this project coupled with his astute guidance has encouraged me in continuing exploring the beauty of research.

Dr. Hopkinson is my secondary sponsor since the joint research I was working on belongs to him and Dr. Cheng. If Dr. Cheng supplies me with ideas, then Dr. Hopkinson equips me with resources and solutions. Dr. Hopkinson has always been there to support me from the very beginning. Never did he turn away my request no matter how big or

small. I will always be grateful for his high expectation and diplomatic encouragement to extend my limit in achieving my highest potential.

Dr. Oxley was highly recommended to be my advisor in the beginning of my research. Although my research interest switched relatively soon after I entered AFIT, I am still grateful for Dr. Oxley's guidance in the beginning of my research and his continue oversight of the rest of my work. It is a pure joy to spend an hour on discussion over the definition of hyperplane. It takes a true mathematician for that level of care, and I respect and treasure it with all my heart.

Lastly, to my thesis advisor, Dr. Schubert Kabban. I saved the acknowledgment of her last because she has had the greatest impact during my pursue at AFIT. It is an honor and a blessing to have the opportunity to learn from her because her erudite discussion is always enriching. Not only did she meet with me weekly for my thesis, guiding me through challenging problems, but she also provided me with equipment for my work. She also has been such an understanding and supportive instructor and an advisor. I still remember vividly, during the time of stress and hopelessness, I struggled to keep up with three classes and research, she expressed no disappointment and offered alternatives to keep my education moving forward and at the same time offered emotional support. My appreciation for her grew stronger as my thesis approached its end. During the time of my career uncertainty, I was in great distress. When I shared with her that I have been distracted from writing my thesis by the on-going movements that could potentially affect my career, and my concern about my thesis might be the last work I complete while in uniform, she offered her deepest sympathy and encouraged me



to find focus even though there is a roaring storm swirling around me. I don't take her understanding and support lightly, not at the slightest. Days and months can go by, I can and will still remember and treasure those times of encouragement.

Besides my committee, I would like to extend my gratitude to my family, especially, my daughter and my husband. To my sweet pea, you are the biggest blessing God has given me. You were only a little over one year old when 妈妈 started AFIT, and you have grown from a baby depending on your parents for everything to an independent little girl who now can care for your sick 妈妈. I could never have asked more from you, my pea, for you have learned to be resilient at such a young age. You are wise beyond your age and with your determination, there is nothing that can stop you from achieving your greatest potential. Never lose sight of God as He is your light and your Heavenly Father who loves you more than you can possibly imagine. 妈妈 will always love you, my sweet pea. To my husband, as our first dance on our wedding 12 years ago sings, I can't imagine living without you. You are the most beautiful and perfect gift from God. You love me unconditionally and provide me with everything that I need to be a wife, a new mother, and a friend. Through the darkest times you have not forsaken me, nor have you rebuked me for my wrong doings. You said you love me because God loves first. You truly have shone the light of Christ. I am forever grateful for you and looking forward to living our days together as we grow old. I love you dearly, Honey.

My Heavenly Father, hallowed be thy name. You have shown to me your sovereignty and greatest love over and over again, especially during times of my human despal and unbelief. For some tasks during my time at AFIT, I cried out to you in my

deepest desperation as it was impossible to complete the tasks that were asked of me. You carried me with great compassion, poured out your infinite wisdom over me, and strengthened me with a healthy body and a sound mind to complete the mission that seemed to human eyes to be impossible. Father, forgive me when I gave into my weakness and strengthen me and my faith to stand firm. Father, show us more of the mercy and love of our crucified and risen Savior, the Lord Jesus Christ. Give us more of your Spirit, that we might live unto Your glory each moment of the day. In Jesus's name, Amen.

Grace Y. Smith

## Table of Contents

	Page
Abstract.....	iv
Table of Contents .....	x
List of Figures .....	xii
List of Tables .....	xv
I. Introduction.....	1
1.1 General Issue.....	1
1.2 Problem Statement .....	5
1.3 Research Objectives .....	5
II. Background .....	9
2.1 Chapter Overview .....	9
2.2 Natural Language Processing .....	9
2.3 Machine Learning .....	13
2.4 Fusion .....	25
2.5 Distributed Representations of Topics .....	30
2.6 Methodologies Applicable to COVID-19 Related Literatures .....	36
2.7 Summary.....	40
III. Sensor Fusion for Context Analysis in Social Media COVID-19 Data .....	42
3.1 Chapter Overview .....	42
3.2 Abstract.....	42
3.3 Introduction.....	43
3.4 Methods .....	45
3.5 Results (Step 6) .....	57
3.6 Discussion and Conclusion (Step 7).....	66
3.7 Acknowledgment .....	68
IV. A Framework for Keywords Identification Via Semantic Analysis in Application to COVID-19 Misinformation on Social Media .....	69
4.1 Chapter Overview .....	69
4.2 Introduction.....	69
4.3 Background.....	71
4.4 Methodology .....	76
4.5 Results .....	90
4.6 Conclusions.....	104
V. Conclusions and Recommendations .....	110
5.1 Chapter Overview .....	110

5.2	Conclusions of Research .....	110
5.3	Significance of Research .....	113
5.4	Recommendations for Future Research.....	114
Bibliography .....		118

## List of Figures

	Page
Figure 1. A Support Vector Classifier Fit to a Small Training Dataset Shown on a Two-Dimensional Space .....	23
Figure 2. The Boolean AND Rule Truth Table .....	26
Figure 3. The Boolean OR Rule Truth Table .....	27
Figure 4. Plate Notation for LDA with Dirichlet-Distributed Topic-Word Distribution Fitted to a Small Dataset .....	34
Figure 5. Customized Data Science Trajectory Inspired by [64] for COVID-19 Twitter Sentiment and Misinformation Analysis .....	45
Figure 6. Tweet Count of the Binary Class for Each Misinformation Category .....	50
Figure 7. Tweet Count of the Multi-Class for Two Combined Misinformation Categories: Unreliable and Political/Biased, Clickbait and Political/Biased.....	51
Figure 8. Venn Diagram for Binary Class Fusion Rules Fusing Three Algorithms .....	55
Figure 9. True Positive Rate and False Positive Rate for Sentiment Dataset .....	58
Figure 10. Prediction Accuracy with 95% Confidence Interval by Algorithm for Sentiment Dataset.....	58
Figure 11. True Positive Rate and False Positive Rate for Predicting Unreliable .....	60
Figure 12. True Positive Rate and False Positive Rate for Predicting Conspiracy.....	60
Figure 13. True Positive Rate and False Positive Rate for Predicting Clickbait .....	60
Figure 14. True Positive Rate and False Positive Rate for Predicting Political/Biased .....	61
Figure 15. Accuracy with 95% Confidence Interval for Predicting Unreliable .....	61
Figure 16. Accuracy with 95% Confidence Interval for Predicting Conspiracy .....	62
Figure 17. Accuracy with 95% Confidence Interval for Predicting Clickbait.....	62
Figure 18. Accuracy with 95% Confidence Interval for Predicting Political/Biased .....	63
Figure 19. True Positive Rate for Four Labels by Algorithm for Combinations Unreliable and Political/Biased. Data Labels Are Shown for Combined Misinformation Category (Blue Squares).....	64
Figure 20. True Positive Rate for Four Labels by Algorithm for Combinations Clickbait and Political/Biased. Data Labels Are Shown for Combined Misinformation Category (Blue Squares).....	64

Figure 21. Overall Accuracy with 95% Confidence Interval for Predicting Combinations Unreliable and Political/Biased .....	65
Figure 22. Overall Accuracy with 95% Confidence Interval for Predicting Combinations Clickbait and Political/Biased .....	65
Figure 23. Percentage of Positive Sentiment for Each Misinformation Category. Blue Circle is for Early COVID-19 Outbreak from Feb 1 to Apr 29 and Orange Triangle is the 5th Month into COVID-19 from July 25 to Aug 29 .....	66
Figure 24. Human-in-the-Loop Interactive Framework Workflow. Human Interaction Occurs in Two Greyed Out Steps .....	78
Figure 25. Confusion Matrix .....	89
Figure 26. Shape Shows a 300 Dimensional Document Vectors Reduced into 2 Dimensions in UMAP. Colors Indicates Dense Areas Identified by HDBSCAN. Parameters: 30 n_neighbors and 15 min_cluster_size. ....	92
Figure 27. Shape Shows a 300 Dimensional Document Vectors Reduced into 2 Dimensions in UMAP. Colors Indicates Dense Areas Identified by HDBSCAN. Parameters: 45 n_neighbors and 45 min_cluster_size. ....	92
Figure 28. Shape Shows a 300 Dimensional Document Vectors Reduced into 2 Dimensions in UMAP. Colors Indicates Dense Areas Identified by HDBSCAN. Parameters: 60 n_neighbors and 45 min_cluster_size. ....	93
Figure 29. Keyword Semantic Quality by Various NLP Tasks.....	95
Figure 30. ROC Curve for Top2vec Model Predicting Conspiracy Employed Stop Words Removal and Lemmatization .....	96
Figure 31. ROC Curve for Top2vec Model Predicting Conspiracy Employed Lemmatization.....	97
Figure 32. Top 50 Conspiracy Keywords that Are Most Similar to Words “Covid”, “Start”, and “Lab” .....	98
Figure 33. Top 50 Unreliable Keywords that Are Most Similar to Words “Covid”, “Start”, and “Lab” .....	99
Figure 34. Top 50 Political/Biased Keywords that Are Most Similar to Words “Covid”, “Start”, and “Lab” .....	99
Figure 35. Top 50 Clickbait Keywords that Are Most Similar to Words “Covid”, “Start”, and “Lab” .....	100
Figure 36. Conspiracy Misinformation ROC Curve for 50 Sets Keywords.....	101
Figure 37. Unreliable Misinformation ROC Curve for 50 Sets Keywords .....	101
Figure 38. Political/Biased Misinformation ROC Curve for 50 Sets Keywords .....	101
Figure 39. Clickbait Misinformation ROC Curve for 50 Sets Keywords .....	102

Figure 40. 21 Recommended Conspiracy Misinformation Keywords Similar to Human Inquiry “Covid”, “Start”, and “Lab” .....	103
Figure 41. 22 Recommended Unreliable Misinformation Keywords Similar to Human Inquiry “Covid”, “Start”, and “Lab” .....	103

## List of Tables

	Page
Table 1. Sample Tweets from Sentiment Dataset .....	47
Table 2. Sample Tweets from Misinformation Dataset .....	48
Table 3. Data Preparation: Pre and Post Text-Preprocessing .....	52
Table 4. Fusion Rules for Both Sets of Combined Labels: Political/Biased with Unreliable or with Clickbait .....	56
Table 5. Hyperparameter Fine-Tuning Combinations Part 1 .....	84
Table 6. Hyperparameter Fine-Tuning Combinations Part 2 .....	86
Table 7. Recommended Keywords Most Probable and Semantically Similar to Conspiracy and Unreliable Misinformation .....	104



# **CLASSIFICATION AND KEYWORD IDENTIFICATION OF COVID-19 MISINFORMATION ON SOCIAL MEDIA: A FRAMEWORK FOR SEMANTIC ANALYSIS**

## **I. Introduction**

### **1.1 General Issue**

We are living in a world where information is shared at real-time speed and the flux of information continues to grow enormously. There were 147.1 million mentions of COVID-19 on social media at a global level in a span of one week during the early outbreak of the pandemic going from March 16th to March 22nd, 2020 [1]. That breaks down to 243 mentions of COVID-19 on social media at any given second throughout the whole world during that single week. Such advanced communication technology provides researchers with unprecedented amount of social and health information for the benefits of scientific findings such as explaining human behaviors and health measures [2].

However, great influxes of information come with large drawbacks. The higher volume of social media information produces a lower signal-to-noise ratio which results in an immense challenge identifying factual and pertinent information [3]. At his call for a whole-of-society effort to confront health misinformation about the COVID-19 pandemic, the United States Surgeon General (U.S. SG), Dr. Vivek Murthy, recognized that the “rapidly changing information environment has made it easier for misinformation to spread at unprecedented speed and scale” [4]. Indeed, the World Health Organization (WHO) anticipated the spread of information during the pandemic to be a crisis of its

own relatively early. On Feb 2, 2020, WHO signaled a byproduct of the pandemic is the massive flow of information, an “infodemic” [5].

A consistent, concise, and universal definition of misinformation remains undetermined. The definition varies from individual to individual and group to group. One definition for misinformation offered by researchers in the field of communication is “information considered incorrect based on the best available evidence from relevant experts at the time” [6]. According to the United States (U.S.) Department of Homeland Security (DHS) in its annual Homeland Threat Assessment, recognized misinformation is a “foreign use of false or misleading information” [7]. The Department of Defense (DoD) understood misinformation as the “unintentional dissemination of false information” [8]. The U.S. SG’s definition of misinformation seems to combine the three definitions above as “information that is false, inaccurate, or misleading according to the best available evidence at the time” [4]. When misinformation is “spread intentionally to serve a malicious purpose”, it is considered “disinformation” according to the U.S. SG [4].

What is the big deal about disinformation and misinformation? Disinformation and misinformation is a threat to the national security and can cause harm to public health. The 2021 Annual Threat Assessment of the U.S. Intelligence Community recognized online disinformation as “a significant threat to the security of U.S. and allied networks and data” [9]. DHS characterized online disinformation and misinformation among COVID-19 as one of the foreign influence activities to weaken America both domestically and abroad “through efforts to sow discord, distract, shape public sentiment,

and undermine trust in Western democratic institutions and processes” [7]. One of the examples is during the first six to seven months (2020 January to July) of the COVID-19 outbreak, malicious actors exploited misleading narratives about the origin of COVID-19, claiming the virus was engineered as a biological weapon to achieve their geopolitical agendas [10]. In September 2020, WHO and other international organizations published a joint statement acknowledging that misinformation and disinformation among COVID-19 can be harmful to an individual’s health both physically and mentally, misinformation destroys lives, and disinformation polarizes public’s opinions [11]. The U.S. SG assessed the impact of health misinformation as a serious threat to public health because it can “cause confusion, sow mistrust, harm people’s health, and undermine public health efforts” [4]. A joint statement by Mr. Christopher Maier, Acting Assistant Secretary of Defense for Special Operations and Low-Intensity Conflict, Mr. Neill Tipton, Director of Defense Intelligence (Collections and Special Programs), and Mr. James Sullivan, Defense Intelligence Agency’s Defense Intelligence Officer for Cyber, before the House Armed Services Committee Subcommittee on Intelligence and Special Operations, stated that disinformation and misinformation is a critical threat to force protection as the U.S. Soldiers, Sailors, Marines, Airmen, Guardians, civilians, and their family are directly targeted by malign actors [8]. The joint statement also recognized that disinformation and misinformation is one of today’s greatest challenges not just to the DoD, but also to the U.S. [8]. The negative impact of the spread of disinformation and misinformation is undeniable. Immediate actions are required to address this ongoing issue as discussed in the next section.

Combating COVID-19 disinformation and misinformation requires actions not just from the stakeholders such as the government, news media outlets, social media platforms, and the public, but rather from the society as a unit standing together to fight the battle. John Hopkins Center for Health Security recently published a report calling for a national strategy to combat threats of COVID-19 health disinformation and misinformation. The report proposed a solution to dissolve this threat: ensuring a whole-of-nation response via multisector and multiagency collective supports from government, academia, and private sectors [12]. The U.S. SG specified various sectors of the society to act upon the call for a whole-of-society effort in confronting COVID-19 health misinformation. Specifically, researchers and research institutions are called to increase vigilance on health questions, concerns, and misinformation via different mediums of information flow and study approaches [4].

From a defense perspective, the DoD has been supporting the following efforts to combat disinformation and misinformation via supports from the Intelligence Community, interagency collaboration, and partnerships [8]. For more specific on-going efforts, the DoD and IC have been providing intelligence support to the Operations in the Information Environment (OIE) and have been providing intelligence dissemination to support Combatant Command Messaging. In particular, nine Combatant Commanders signed a memorandum known as the “36-star memo” in January 2020. The memorandum requested increased intelligence support for “messaging and countering disinformation operations as part of great power competition.” The Office of the Under Secretary of Defense for Intelligence and Security and the joint DoD-Director of National

Intelligence responded with efforts in support of OIE. Though this line of effort of responding to the “36-star memo” was completed recently in September 2021, many follow-on initiatives have continued, especially efforts with focus in Open-Source intelligence.

## **1.2 Problem Statement**

In light of DoD’s increasing demand on Open-Source intelligence in combating COVID-19 disinformation and misinformation, this thesis work aims to address questions that may contribute to any on-going efforts that have been put forth by the DoD. This work studied online social media posts from the social media platform Twitter during the first year of the COVID-19 pandemic in 2020. The questions of interest are as follow.

1. What is the general public’s sentiment toward COVID-19 misinformation?
2. Are there any changes in sentiment of the general public toward COVID-19 misinformation over time?
3. Is there any system available with which humans may interact regarding specific types of COVID-19 misinformation? If not, can we build one?
4. Can we build a framework that allows humans to query a topic of interest on COVID-19 misinformation and retrieve topic related keywords and posts?

## **1.3 Research Objectives**

There are many types of COVID-19 misinformation with various prevalence on social media. This thesis work adopted [13] to categorize misinformation as four types: unreliable, conspiracy, clickbait, and political or biased. In addition, a Twitter post may be considered having more than one type of misinformation. That is, a Tweet can be

labelled as both unreliable and political or biased misinformation. The general public's sentiment is categorized as positive sentiment or negative sentiment. Thus, a Tweet has both contextual and sentimental characteristics. For example, one might post negative conspiracy and political or biased misinformation while others may spread positive political or biased misinformation on Twitter.

Most publications on social media COVID-19 misinformation detection and diffusion stop at the foundation of classifying fake news and general discussion on dispersion of fake news. Some offered dashboards for visualization of the fake news propagation through time and space. However, to the best of the author's knowledge to date, there has not been any publication on a user oriented/interactive system that allows users to search topic of interests relating COVID-19 misinformation on social media; this gap is addressed in the following objectives, specifically, Objective 3. Therefore, to answer the research questions in the previous section, the below objectives were set and achieved in this thesis.

Objective 1: Provide knowledge discovery of general public's sentiment toward four types of misinformation regarding COVID-19 news during the early outbreak of the pandemic from March 9 to April 24, 2020.

Objective 2: Detect sentiment changes regarding COVID-19 misinformation from the early outbreak February 1 - April 29, 2020 to a summer month of July 25 - August 29, 2020.

Objective 3: Construct a human-in-the-loop framework for enabling an interactive process to ingest a human input for a topic of interest then provide both

recommended keywords semantically similar to and accurately related to the human input as well as related documents. Specifically, the human-in-the-loop framework digests any dataset in a form of text through natural language processing. It then takes advantage of a text mining algorithm for topic modeling and semantic search in order to take a user's topic of interest in a form of either keywords or a sentence and return keywords that are semantically similar to the user input topic. The novel aspect of this framework is that it then makes recommendations on the ideal number of keywords as well as identifying such keywords along with each word's probability of being in a target category. The selection of the ideal set of keywords is based on the best classification performance. That is, the ideal set of keywords scores the highest in accurately being contained in the context of documents containing at least one of the ideal set of keywords whose document is correctly identified for a specific targeted category. The framework ideally should work for any dataset comprised of natural language communication; this thesis illustrated the proof of concept and applied this framework to a COVID-19 Twitter dataset.

This document is organized as follows: (1) Chapter II provides background on the topics and techniques used to achieve the research objectives; (2) Objectives 1 and 2 were achieved and documented in Chapter III which is a reprint of a conference paper [14] presented at the 2021 Institute of Electrical and Electronics Engineers (IEEE) National Aerospace & Electronics Conference; (3) the realization of Objective 3 is shown in Chapter IV which is a planned submission to the IEEE Transactions on Computational

Social Systems journal; and (4) final discussion and conclusions are provided in Chapter V.



## **II. Background**

### **2.1 Chapter Overview**

This chapter covers four methods employed in the thesis as well as applications of these methods in recent literatures related to COVID-19. In Sections 2.2 to 2.5, each section begins with a literature review on the method and then transitions to the detail of the method. These methods were used in various combinations to achieve the research objectives. Specifically, Chapter III applied natural language processing (NLP), machine learning (ML), and fusion while Chapter IV exploited NLP, ML, and distributed representations of topics. Section 2.6 is the application of the related methods and Section 2.7 is the summary of this chapter.

### **2.2 Natural Language Processing**

#### ***2.2.1 Literature Review.***

Computational linguistics, also known as natural language processing (NLP), is a subfield of computer science which attempts to understand, learn, and produce one or more human languages [15]. Natural language processing may process not only text, but also speech, image, and video. The essential challenge in processing natural language in all forms may contribute to the ubiquitous ambiguity found at all levels of the problem. James Allen provided five challenging ambiguities that all natural language process as follow [15].

- Simple lexical ambiguity. (e.g. “duck” can be a noun referring to a bird or a verb meaning to avoid something thrown.)

- Structural or syntactic ambiguity. (e.g. in “I saw the man with a telescope,” the telescope might be used for the viewing or might be held by the man being observed.)
- Semantic ambiguity. (e.g. “go” as a verb has well over 10 distinct meanings in any dictionary.)
- Pragmatic ambiguity. (e.g. “Can you lift that rock?” may be a yes/no question or a request to lift the rock.)
- Referential ambiguity. (e.g. “Jack met Sam at the station. He was feeling ill...,” it is not clear who is ill, although the remainder of the sentence might suggest a preferred interpretation.)

Despite the challenges, many applications to natural language processing have made significant improvements over the past two decades, especially in machine translation, machine reading, spoken dialogue systems and conversational agents, social media mining, and analysis and generation of speech state [16]. Other applications to natural language processing includes, but is not limited to the following according to Towards Data Science [17].

- Retrieval. ([Google](#) finds relevant and similar results.)
- Information extraction. ([Gmail](#) structures events from emails.)
- Machine translation. ([Google Translate](#) translates language from one language to another.)
- Text simplification. ([Rewordify](#) simplifies the meaning of sentences.)
- Sentiment analysis. ([Hater News](#) gives us the sentiment of the user.)
- Text summarization. ([Smmry](#) gives a summary of sentences.)

- Spam filter. (Gmail filters spam emails separately.)
- Automatic prediction. (Google Search predicts user search results.)
- Automatic correction. (Google Keyboard and [Grammarly](#) correct words otherwise spelled wrong.)
- Speech recognition. (Google [WebSpeech](#) or [Vocalware](#).)
- Question answering.
- Natural language generation. (Generation of text from image or video.)

In this work, social media data mining, text preprocessing, context analysis including sentimental context and textual context, and semantic search were direct beneficiaries of natural language processing.

### ***2.2.2 Baseline Text Preprocessing.***

Baseline text-preprocessing applied in this thesis work used regular expression to remove and replace certain words or characters yet preserved semantically meaningful information. The baseline text-preprocessing includes removing non-alphabetic characters and non-informative words; replacing emojis and smileys with the word “happy” or “sad” accordingly; replacing contractions such as “didn’t” with its long form “did not”; case folding which converts all words to lower cases; removing non-alphabetic words such as numbers and symbols; removing non-ASCII characters, mentions, urls, retweet “RT”, and single letters; replacing punctuations with a space; replacing three or more identical consecutive letters with two letters.

### 2.2.3 *Normalization.*

Various normalization methods can be applied after the baseline text-preprocessing step. In particular, tokenization, stop words removal, and lemmatization are often seen as normalization methods. Tokenization is a process to break down a string of text into a smaller unit such as words, and the words are called tokens. Text in the English language is separated by a white space, therefore, it is a common practice to tokenize English text by a white space. However, separating text by a white space may incur issues when inferring meaning, for instance, separating the phrase “rock n roll” into three individual tokens “rock”, “n”, “roll”.

Stop words are a set of commonly used words that carry little semantic information. This research made use of the Natural Language Toolkit (NLTK) [18] Stop Words list which contains 179 stop words. Stop words in this list include words such as pronouns “I”, “them”; prepositional words such as “from”, “on”; contractions words such as “couldn’t”, “couldn’t”. Notice there is no letter t for the contraction example “couldn’t”.

Lemmatization is a text process which reduces words to their stems, i.e., removes affixes of words to obtain their root form. There are two types of lemmatizations applied in this research. First, Porter Stemmer is a tool for removing the more common morphological and inflexional endings from words in English [19]. Porter Stemmer uses its own rules for deciding how to remove affixes. For example, the word “lying” is a variation of the word “lie”. However, a limitation of this method includes instances in which some resulting tokens might not appear as English words. For example, the

process might reduce the word “diabetes” to “diabet” where “es” is removed since “es” could be suffix of a plural form. Second, WordNet Lemmatizer is a tool for removing affixes only if the resulting word is contained in the WordNet dictionary [20]. WordNet lemmatizer converts the word “women” to “woman” while it does not alter the word “lying” as in Porter Stemmer.

## **2.3 Machine Learning**

### ***2.3.1 Literature Review.***

Naïve Bayes, logistic regression, and support vector classifier algorithms are supervised classical machine learning algorithms that predict an output of a class based on inputs and the corresponding ground truth labels. All three of these machine learning classifiers employed in this research are linear classifiers producing the estimated class based on a linear combination of the features. Naïve Bayes is a generative model while both logistic regression and support vector classifier are discriminative models. Generative or discriminative depends on the process of obtaining the output, i.e., predicting a class to which a document belongs. The prediction of a class depends on the conditional probability  $P(c|d)$  where  $c$  is a class and  $d$  is a document. Naïve Bayes is a generative model as it does not compute the probability directly, rather, it computes the probabilities of a prior and a likelihood. A generative model such as naïve Bayes takes advantage of the likelihood such that features of a document can be generated under the condition of knowing what class to which each feature belongs in the document. A discriminative model attempts to compute the conditional probability  $P(c|d)$  directly in hope of learning to assign a high weight to document features such that its ability to

discriminate between classes can be improved. The description of the naïve Bayes [21] and logistic regression [22] models in the sections below are based on the third edition of a working book, Speech and Language Processing, by Daniel Jurafsky and James Martin [23]. The description of a support vector classifier is based on the book “An Introduction to Statistical Learning with Applications in R” by Gareth et al. [24].

### 2.3.2 Naïve Bayes.

As applied to the data used in this research, the naïve Bayes algorithm contains a probabilistic classifier technique selecting a class with the highest computed posterior probability for a given Tweet by applying Bayes’ rule with the bag of words assumption and conditional independence assumption. First, a probabilistic classifier means that for a Tweet  $t$ , out of all classes  $c \in C$ , the classifier returns the class  $c$  that has the maximum posterior probability conditioning on the Tweet  $t$ . Therefore, naïve Bayes is estimating  $c$  by

$$c \approx \hat{c} = \max_{c \in C} P(c|t). \quad (1)$$

By Bayes rule,

$$P(y|x) = \frac{P(x|y)P(y)}{P(x)}, \quad (2)$$

and  $\hat{c}$  becomes

$$c \approx \hat{c} = \max_{c \in C} P(c|t) = \max_{c \in C} \frac{P(t|c)P(c)}{P(t)}. \quad (3)$$

Since the probability of a Tweet,  $P(t)$ , is a constant for each class,  $P(t)$  is set to 1 and is therefore dropped from Equation (3) which becomes

$$c \approx \hat{c} = \max_{c \in \mathcal{C}} P(c|t) = \max_{c \in \mathcal{C}} P(t|c)P(c). \quad (4)$$

In short, naïve Bayes is selecting the highest posterior probability,  $P(c|t)$ , by selecting the highest product of two probabilities: the likelihood of the Tweet occurring for class  $c$ ,  $P(t|c)$ , and the prior probability of the class  $c$ ,  $P(c)$ .

Without loss of generality, suppose a Tweet is represented as a set of words or features  $w_1, w_2, w_3, \dots, w_n$  such that Equation (4) becomes

$$c \approx \hat{c} = \max_{c \in \mathcal{C}} P(w_1, w_2, w_3, \dots, w_n|c)P(c). \quad (5)$$

Then estimating  $P(w_1, w_2, w_3, \dots, w_n|c)$  requires two assumptions: (1) the bag of words assumption, that is, the order or the position of the words do not matter in a Tweet; (2) the naïve Bayes assumption which is the conditional independence assumption given in Equation (6) for generic events  $A$ ,  $B$  and  $C$  in which  $A$  and  $B$  are expressed as conditionally independent of event  $C$ :

$$P(A \cap B|C) = P(A|C)P(B|C). \quad (6)$$

Assuming conditional independence, then Equation (5) becomes

$$c \approx \hat{c} = \max_{c \in \mathcal{C}} P(c)P(w_1|c)P(w_2|c), \dots, P(w_n|c). \quad (7)$$

Therefore, the equation for the probability associated with a particular class using the naïve Bayes classifier can be written as  $c_{NB}$  where

$$c_{NB} = \max_{c \in \mathcal{C}} P(c) \prod_{w \in W} P(w|c). \quad (8)$$

Indexing each word in the training set  $W$  and defining  $I$  as the set of indexes in Equation (8) yields

$$c_{NB} = \max_{c \in \mathcal{C}} P(c) \prod_{i \in I} P(w_i|c). \quad (9)$$

Equation (9) could cause a computational issue called overflow or underflow if there are tens of thousands of features in a language model which is very common in practice.

Overflow or underflow occur when a number exceeds the value range for a data type that a standard computer can store or represent. To avoid this computational issue, the logarithm is applied as follows

$$c_{NB} = \max_{c \in C} \log P(c) + \sum_{i \in I} \log P(w_i | c). \quad (10)$$

To calculate the probability of a class  $P(c)$ , first let  $T_c$  be the number of Tweets in class  $c$  and let  $N_t$  be the total number of Tweets. Then  $P(c)$  is the percentage of Tweets in training dataset  $W$  that are in each class  $c$  and estimated as

$$P(c) \approx \hat{P}(c) = \frac{T_c}{N_t}. \quad (11)$$

There are multiple ways to calculate  $P(w_i | c)$ , here, a specific algorithm was used as a convention. Specifically, the multinomial naïve Bayes algorithm [25] from Scikit-learn [26] was employed as a classifying algorithm. Furthermore, the `predict_proba` method from the `MultinomialNB` module was selected in calculating  $P(w_i | c)$  which is formulated as in Equation (12).

$$P(w_i | c) = \frac{N_{ci} + \alpha}{N_c + n\alpha} = \frac{\sum_{w \in W} w_i + \alpha}{\sum_{i=1}^n N_{ci} + n\alpha}, \quad (12)$$

where  $w_i$  is the  $i$ th feature/word,  $c_i$  is the  $i$ th class,  $n$  is the number of features/words,  $W$  is the training dataset,  $N_{ci} = \sum_{w \in W} w_i$  is the number of times the  $i$ th feature/word appears in class  $c$  in the training dataset  $W$ ,  $N_c = \sum_{i=1}^n N_{ci}$  is the total number of times each feature/word appear in class  $c$  in the training dataset  $W$ , that is,  $N_c$  is the word count in class  $c$ ,  $\alpha$  is a smoothing prior accounting for features not present in the training sample



to prevent zero probabilities in calculation. Note that this smoothing prior has a default value of  $\alpha = 1$  which is called Laplace smoothing.

### 2.3.3 *Logistic Regression.*

Logistic regression is a discriminative classifier that computes the probability of assigning a class  $c$  to a Tweet  $t$ ,  $P(c|t)$ . Compared with the naïve Bayes algorithm, the most distinct difference is that naïve Bayes is a generative classifier where Tweets are generated by words which are generated by sampling from the conditional probability  $P(t|c)$ . The naïve Bayes classifier estimates  $P(c|t)$  by estimating the product of a likelihood probability and a prior probability without computing the conditional probability  $P(c|t)$  directly. In contrast, logistic regression is a discriminative model which aims to learn the appropriate class by putting more weights on the words such that the model is able to discriminate between classes despite the fact that the model was not able to generate an example of one of the classes.

Both naïve Bayes classifier and logistic regression classifier are probabilistic classifiers that employ supervised machine learning. There are four components for such a probabilistic machine learning classifier: (1) an input represented by a feature; (2) an output determined by a classification function; (3) an objective function for learning; and (4) an algorithm for optimizing the objective function. A brief summary of these four components specific to logistic regression follows.

(1) An input represented by a feature. Any supervised machine learning classifier requires  $i = 1, \dots, m$  pairs of input/ output in the training corpus,  $(x^{(i)}, y^{(i)})$ .

A vector representing features for each  $x^{(i)}$  can be written as  $[x_1, x_2, \dots, x_n]$ , and the  $i$ th feature is denoted by  $x_i$ .

(2) An output determined by a classification function. For logistic regression, sigmoid and softmax are the classification functions for binary classes and multi-classes, respectively. For the binary case with two class outcomes,  $y_1$  and  $y_2$ , the classification function makes a prediction based on probabilities of an observation  $x$  assigned to all possible classes  $y$ ,  $P(y|x)$ , where  $y \in \{y_1, y_2\}$ . The logistic regression learns a vector of weights and bias from a training dataset to indicate the importance of each feature. Each weight  $w_i$  is a real number signifying the importance of an input feature  $x_i$  to the classification function that determines the output label for a specific class as  $\hat{y} = y_1$  or  $\hat{y} = y_2$ . Bias is known as the intercept. The bias  $b$  is a real value that adds to the weighted input. The weighted sum combining weights and the bias is governed by the linear function  $z$  as shown in Equation (13).

$$z = (\sum_{i=1}^n w_i x_i) + b. \quad (13)$$

The sum product of weight  $w_i$  and  $x_i$ ,  $\sum_{i=1}^n w_i x_i$ , can be represented as  $\vec{w} \cdot \vec{x}$  where  $\vec{w} = [w_1, w_2, \dots, w_n]$  and  $\vec{x} = [x_1, x_2, \dots, x_n]$ , then  $z$  becomes

$$z = \vec{w} \cdot \vec{x} + b. \quad (14)$$

The sigmoid function maps the value of the weighted sum  $z$  to an interval  $[0,1]$ , representing a candidate probability estimate for a specific class. The sigmoid is also known as the logistic function denoted as  $\sigma(z)$  and shown as

$$\sigma(z) = \frac{1}{1+e^{-z}} = \frac{1}{1+\exp(-z)}. \quad (15)$$

$\sigma(z)$  returns a value between 0 and 1. To ensure  $\sigma(z)$  is a probability, we make the probabilities of an observation assigned to all possible classes sums to 1, i.e.,  $P(y = y_1|x) + P(y = y_2|x) = 1$ . Then, the probability of an observation being in class  $y_1$  and class  $y_2$  are shown in Equation (16) and Equation (17), respectively. Note that Equation (16) and Equation (17) are the probabilities for a function on  $y$  that labels  $y$  as outcome class  $y_1$  and class  $y_2$ , respectively.

$$\begin{aligned}
P(y = y_1) &= \sigma(z) \\
&= \sigma(\vec{w} \cdot \vec{x} + b) \\
&= \frac{1}{1 + e^{-(\vec{w} \cdot \vec{x} + b)}} \\
&= \frac{1}{1 + \exp(-(\vec{w} \cdot \vec{x} + b))}.
\end{aligned} \tag{16}$$

And applying the sigmoid function property  $1 - \sigma(x) = \sigma(-x)$ , we have

$$\begin{aligned}
P(y = y_2) &= 1 - P(y = y_1) \\
&= 1 - \sigma(z) \\
&= 1 - \frac{1}{1 + \exp(-(\vec{w} \cdot \vec{x} + b))} \\
&= \frac{\exp(-(\vec{w} \cdot \vec{x} + b))}{1 + \exp(-(\vec{w} \cdot \vec{x} + b))} \\
&= \sigma(-(\vec{w} \cdot \vec{x} + b)).
\end{aligned} \tag{17}$$

A decision boundary is an arbitrary value between 0 and 1 to guide the classification function in predicting the estimated class  $\hat{y}$ . In particular, suppose the decision boundary is 0.5, the classification function assigns the estimated class  $\hat{y}$  to  $y_1$  if the probability of an observation in  $y_1$  class is greater than 0.5. Thus, we have

$$\hat{y} = \begin{cases} y_1 & \text{if } P(y = y_1|x) > 0.5 \\ y_2 & \text{otherwise.} \end{cases} \quad (18)$$

(3) An objective function for learning. A common objective function for learning weights in a machine learning approach for logistic regression is the cross-entropy loss function. Generally, a loss function measures the difference between the classifier output  $\hat{y}$  and the ground truth output  $y$  and is denoted as  $L(\hat{y}, y)$ . The cross-entropy loss function is the negative log likelihood loss where the log probability of the true  $y$  labels in the training dataset is maximized, known as the conditional maximum likelihood estimation. Therefore, the goal is to maximize the probability of the true class label  $P(y|x)$  and is shown as

$$\begin{aligned} \text{Maximize: } \log(P(y|x)) &= \log(\hat{y}^y (1 - \hat{y})^{1-y}) \\ &= y \log(\hat{y}) + (1 - y) \log(1 - \hat{y}). \end{aligned} \quad (19)$$

Then flipping the sign to turn the probability into a loss to be minimized as the following

$$\begin{aligned} \text{Minimize: } L_{CE}(\hat{y}, y) &= -\log(P(y|x)) \\ &= -(y \log(\hat{y}) + (1 - y) \log(1 - \hat{y})) \\ &= -(y \log(\sigma(\vec{w} \cdot \vec{x} + b)) + (1 - y) \log(1 - \sigma(\vec{w} \cdot \vec{x} + b))). \end{aligned} \quad (20)$$

(4) An algorithm for optimizing the objective function. The last step is to minimize the cross-entropy loss function via stochastic gradient descent to identify the optimal weights and bias. Let  $\theta$  denote the set of parameters in the loss function, then,  $\theta = \{w, b\}$  is the parameterized parameter for the logistic regression cross-entropy loss function. The goal is to find the average of the minimized loss function identified by the set of weights,  $\hat{\theta}$ .

$$\hat{\theta} = \min_{\theta} \frac{1}{m} \sum_{i=1}^m L_{CE}(f(x^i; \theta), y^i). \quad (21)$$

Since the logistic regression loss function is a convex function, the global minimum can be identified by the gradient descent of the loss function. That is, to find the gradient of the loss function at the current position then move in the opposite direction. The magnitude of the amount of move in gradient descent is the slope of  $\frac{d}{dw} f(x; w)$  weighted by a learning rate  $\eta$ . In each dimension  $w_i$ , the slope  $\nabla_{\theta} L(f(x; \theta), y)$  is the partial derivative  $\frac{\partial}{\partial w_i}$  of the loss function  $L(f(x; \theta), y)$ . Finally, to update  $\theta$  based on the gradient is

$$\theta_{t+1} = \theta_t - \eta \nabla L(f(x; \theta), y). \quad (22)$$

Therefore, the derivative of the cross-entropy loss function becomes

$$\frac{\partial L_{CE}(\hat{y}, y)}{\partial w_j} = (\sigma(\vec{w} \cdot \vec{x} + b) - y) x_j. \quad (23)$$

For this work, the Logistic Regression algorithm [27] from Scikit-learn [26] was employed to perform the classification task.

#### 2.3.4 Support Vector Classifier.

The support vector classifier is another discriminative model where the output of the estimated class is determined by the position of a test observation relative to a specific hyperplane meant to discriminate between the classes of interest. A hyperplane in a linear space is a co-dimensional one linear space that can be translated. A  $p$ -dimensional hyperplane can be defined as

$$\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p = 0. \quad (24)$$

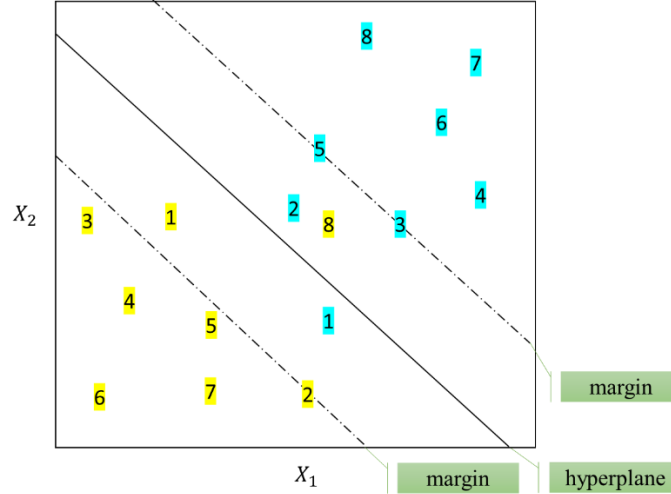
If a point  $X = (X_1, X_2, \dots, X_p)^T$  in  $p$ -dimensional space satisfies Equation (24) then  $X$  lies on the hyperplane. If a point does not satisfy Equation (24), instead,

$$\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p > 0, \quad (25)$$

then the point lies above the hyperplane. If the lefthand side of Equation (24) is less than zero, then a point lies below the hyperplane. Training a classifier that is solely based on a separating hyperplane requires perfectly separable training observations which in practice rarely occurs. Therefore, a room for error around the hyperplane is more practical. That is, creating a bandwidth or margin around the hyperplane to allow a few misclassifications of observations during the training of the classifier is a viable solution to issues risen by a perfectly separating hyperplane.

The support vector classifier, also known as the soft margin classifier, allows some training observations to be on the incorrect side of the margin, or even on the incorrect side of the hyperplane during the training phase. The margin is soft since it allows a few violations of training observations to be on the incorrect side of the hyperplane. As an illustration, Figure 1 presents a support vector classifier trained using a small dataset. The black solid line is the hyperplane, and the two dashed lines are margins. Majority of the observations fall on the correct sides of the margins and only a few observations are on the wrong sides of the margin. Yellow observations 3, 4, 5, 6, 7 are on the correct side of the margin, observation 2 is on the margin, observation 1 is on the wrong side of the margin, and the worst is when observation 8 is on the wrong side of the hyperplane. Teal observations 4, 6, 7, 8 are on the right side of the margin,

observations 3, 5 are on the margin, observation 2 is on the wrong side of the margin while observation 1 is on the wrong side of the hyperplane.



**Figure 1. A Support Vector Classifier Fit to a Small Training Dataset Shown on a Two-Dimensional Space**

To summarize, a support vector classifier classifies a test observation by the side of the hyperplane on which it falls. The hyperplane is trained to separate the training observations as much as possible while allowing room for a few misclassifications. To train a support vector classifier is a maximization task summarized as

$$\max_{\beta_0, \beta_1, \beta_2, \dots, \beta_p, \epsilon_1, \epsilon_2, \dots, \epsilon_p, M} M \quad (26)$$

$$\text{subject to } \sum_{j=1}^p \beta_j^2 = 1 \quad (27)$$

$$y_i(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_p x_{ip}) \geq M(1 - \epsilon_i) \quad (28)$$

$$\epsilon_i \geq 0, \sum_{i=1}^n \epsilon_i \leq C \quad (29)$$

where  $x_1, x_2, \dots, x_n \in \mathbb{R}^p$  are  $n$  training observations and  $y_1, y_2, \dots, y_n \in \{-1, 1\}$  are the ground truth labels associated with the training observations. Labels of  $-1$  and  $1$  are two classes each observation to which can be assigned.  $M$  is the width of the margin,  $\epsilon_1, \epsilon_2, \dots, \epsilon_p$  are slack variables with non-negative real values that allow each individual observations to be on the wrong side of the margin or the hyperplane, and  $C$  is a non-negative tunable hyperparameter determines the amount of tolerances for violation to the margin. Finally, the side of the hyperplane on which a test observation  $x'$  falls is determined by the sign of  $f(x')$

$$f(x') = \beta_0 + \beta_1 x'_1 + \beta_2 x'_2 + \dots + \beta_p x'_p. \quad (30)$$

One interesting property of support vector classifiers is that only the observations lie on the margin or that violate to the margin affect the hyperplane. All other observations being on the correct side of the margin do not play a role in shaping the margin. This is where the name support vectors come from, the observations that affect the hyperplane are the support vectors since they affect the support vector classifier. Therefore, depending on the tunable hyperparameter  $C$ , the number of support vectors may incur bias-variance trade-off of the support vector classifier. For a small value of  $C$ , the width of the margin  $M$  will be small, and the number of support vectors that lie directly on the margin or on the wrong side of the margin will be small, therefore only a few numbers of training observations, here being the support vectors, are used to determine the hyperplane. This scenario leads to a classifier with high variance but low bias and highly fit to training data. Large  $C$  value results in large  $M$ , large number of support vectors, and a classifier that is less fit to training data with large bias and small



variance. For this work, Linear Support Vector Classifier (LinearSVC) [28] from Scikit-learn [26] was employed to perform the classification task.

## 2.4 Fusion

### 2.4.1 Literature Review.

The concept of data fusion is hardly anything new. In 1997, Hall and Llinas provided a well-known definition of data fusion: “[d]ata fusion techniques combine data from multiple sensors, and related information from associated databases, to achieve improved accuracies and more specific inferences than could be achieved by the use of a single sensor alone” [29]. Multiple literature have shown the overall accuracy or Receiver Operating Characteristic (ROC) curves of fusion outcome performs better than that of an individual source [30], [31], [32]. There are numerous ways to combine multiple individual sources. Depending on the methods chosen for such a combination task, some fused outcomes may lead to a better performance than that of a single source while some lead to worse ones [33]. In this work, we considered fusing the outcomes of three machine learning classifiers using four types of fusion rules, namely, two Boolean based-fusion rules: the logical AND rule and the logical OR rule, majority vote fusion rule, and sensor dominance fusion rule. A brief summary of each rule follows.

### 2.4.2 Logical AND Fusion Rule.

Suppose there is a label set  $S = \{m, n\}$  where  $m, n$  have categorical labels such that  $m$  represents “misinformation” and  $n$  represents “non-misinformation”. The logical

AND rule applies a binary operation denoted by  $\wedge$  on  $S$  and its results can be found in the AND truth table shown in Figure 2.

AND	$B_\phi(e)$	
	$\wedge$	
$A_\theta(e)$	$m$	$n$
	$n$	$n$

**Figure 2. The Boolean AND Rule Truth Table**

Therefore, the new classifier  $R_{\theta,\phi}^{AND}$  is defined by the point-wise logical AND rule on each element output, that is,

$$R_{\theta,\phi}^{AND}(e) = A_\theta(e) \wedge B_\phi(e) \quad \text{for all } e \in E, \quad (31)$$

where  $A, B$  are two individual classifiers whose parameters are  $\theta$  and  $\phi$ , respectively;  $e$  is an event outcome and  $E$  is a set of all event outcomes. When the full range of classifier parameters are considered, then a new classifier family  $\mathcal{R}^{AND}$  is generated as

$$\mathcal{R}^{AND} = \{R_{\theta,\phi}^{AND} : \theta \in \Theta, \phi \in \Phi\}. \quad (32)$$

Therefore, the new classifier produced under the logical AND rule,  $R_{\theta,\phi}^{AND}(e)$ , labels an event  $e$  as misinformation  $m$  when both classifiers  $A_\theta(e)$  and  $B_\phi(e)$  label the event  $e$  as a misinformation  $m$ . That is,  $R_{\theta,\phi}^{AND}(e) = m$  only when  $A_\theta(e) = m$  and  $B_\phi(e) = m$ . If either  $A_\theta(e)$  or  $B_\phi(e)$  or both  $A_\theta(e)$  and  $B_\phi(e)$  labels an event as non-misinformation  $n$ , then the fused classifier  $R_{\theta,\phi}^{AND}(e)$  will label the event as non-misinformation.

In a situation where three individual classifiers were used, the binary operation  $\wedge$  on three sets can be reduced to operation on two set by set commutative and associative properties as follows. Suppose three classifiers are  $X_\alpha(e)$ ,  $Y_\beta(e)$ , and  $Z_\gamma(e)$ , then the results of ANDing three classifiers is congruent to that of ANDing two classifiers as shown below.

$$\begin{aligned}
& X_\alpha(e) \wedge Y_\beta(e) \wedge Z_\gamma(e) \\
&= (X_\alpha(e) \wedge Y_\beta(e)) \wedge Z_\gamma(e) \\
&= X_\alpha(e) \wedge (Y_\beta(e) \wedge Z_\gamma(e)) \\
&\cong A_\theta(e) \wedge B_\phi(e).
\end{aligned} \tag{33}$$

#### 2.4.3 Logical OR Fusion Rule.

Using the same set notation as for the logical AND fusion rule, the logical OR rule applies a binary operation denoted by  $\vee$  on  $S$  and its results can be found in the OR truth table as shown in Figure 3.

OR		$B_\phi(e)$	
	$\vee$	$m$	$n$
$A_\theta(e)$	$m$	$m$	$m$
	$n$	$m$	$n$

**Figure 3. The Boolean OR Rule Truth Table**

Therefore, the new classifier  $R_{\theta,\phi}^{OR}$  is defined by the point-wise logical OR rule on each element output, that is,

$$R_{\theta,\phi}^{OR}(e) = A_{\theta}(e) \vee B_{\phi}(e) \text{ for all } e \in E. \quad (34)$$

A new classifier family  $\mathcal{R}^{OR}$  is generated as

$$\mathcal{R}^{OR} = \{R_{\theta,\phi}^{OR} : \theta \in \Theta, \phi \in \Phi\}. \quad (35)$$

Therefore, the new classifier produced under the logical OR rule,  $R_{\theta,\phi}^{OR}(e)$ , labels an event  $e$  as misinformation  $m$  when either classifier  $A_{\theta}(e)$  or  $B_{\phi}(e)$  or both label the event  $e$  as a misinformation  $m$ . That is,  $R_{\theta,\phi}^{OR}(e) = m$  when  $A_{\theta}(e) = m$  or  $B_{\phi}(e) = m$  or both  $A_{\theta}(e) = m$  and  $B_{\phi}(e) = m$ . If both  $A_{\theta}(e)$  and  $B_{\phi}(e)$  label an event as non-misinformation  $n$ , then the fused classifier  $R_{\theta,\phi}^{OR}(e)$  will be labeled as non-misinformation.

Similarly, in a situation where three individual classifiers were used, the binary operation  $\vee$  on three sets can be reduced to operation on two set by set commutative and associative properties as follows. Suppose three classifiers are  $X_{\alpha}(e)$ ,  $Y_{\beta}(e)$ , and  $Z_{\gamma}(e)$ , then the results of ORing three classifiers is congruent to that of ORing two classifiers as shown below.

$$\begin{aligned} & X_{\alpha}(e) \vee Y_{\beta}(e) \vee Z_{\gamma}(e) \\ &= (X_{\alpha}(e) \vee Y_{\beta}(e)) \vee Z_{\gamma}(e) \\ &= X_{\alpha}(e) \vee (Y_{\beta}(e) \vee Z_{\gamma}(e)) \\ &\cong A_{\theta}(e) \vee B_{\phi}(e). \end{aligned} \quad (36)$$

#### 2.4.4 Majority Vote Fusion Rule.

Building upon the logical AND and OR fusion rules, then for the three classifiers  $X_{\alpha}(e)$ ,  $Y_{\beta}(e)$ , and  $Z_{\gamma}(e)$ , the majority vote fusion rule applies binary operations  $\vee$  and  $\wedge$

on  $S$  such  $\vee$  is applied after applying  $\wedge$  to each distinct pair sets. Then the new classifier  $R_{A,B,\Gamma}^{MV}(e)$  for three classifiers becomes

$$R_{A,B,\Gamma}^{MV}(e) = (X_\alpha(e) \wedge Y_\beta(e)) \vee (X_\alpha(e) \wedge Z_\gamma(e)) \vee (Y_\beta(e) \wedge Z_\gamma(e)) \quad . \quad (37)$$

for all  $e \in E$ .

A new classifier family  $\mathcal{R}^{MV}$  under majority vote is generated as

$$\mathcal{R}^{MV} = \{R_{A,B,\Gamma}^{MV}: \alpha \in A, \beta \in B, \gamma \in \Gamma\} \quad . \quad (38)$$

Therefore, the new classifier produced under the majority vote rule,  $R_{A,B,\Gamma}^{MV}(e)$ , labels an event  $e$  as misinformation  $m$  when majority of classifiers agree on an event outcome.

That is,  $R_{A,B,\Gamma}^{MV}(e) = m$  when  $X_\alpha(e) = m$  and  $Y_\beta(e) = m$ , or  $X_\alpha(e) = m$  and  $Z_\gamma(e) = m$ , or  $Y_\beta(e) = m$  and  $Z_\gamma(e) = m$ , or  $X_\alpha(e) = m$  and  $Y_\beta(e) = m$  and  $Z_\gamma(e) = m$ .

#### 2.4.5 Sensor Dominance Fusion Rule.

The sensor dominance rule applies binary operations  $\vee$  and  $\wedge$  on  $S$  such that  $\vee$  is applied after applying  $\wedge$  to a pair of non-dominating sets. Then the new classifier  $R_{A,B,\Gamma}^{SD}(e)$  for three classifiers becomes

$$R_{A,B,\Gamma}^{SD}(e) = X_\alpha(e) \vee (Y_\beta(e) \wedge Z_\gamma(e)) \text{ for all } e \in E, \quad (39)$$

where  $X_\alpha(e)$  is a dominating classifier. A new classifier family  $\mathcal{R}^{SD}$  under sensor dominance is generated as

$$\mathcal{R}^{SD} = \{R_{A,B,\Gamma}^{SD}: \alpha \in A, \beta \in B, \gamma \in \Gamma\}. \quad (40)$$

Therefore, the new classifier produced under the sensor dominance rule,  $R_{A,B,\Gamma}^{SD}(e)$ , labels an event  $e$  as misinformation  $m$  when the dominating classifier labels the event as

misinformation or both non-dominating classifiers labels the event as misinformation.

That is,  $R_{A,B,\Gamma}^{SD}(e) = m$  when  $X_\alpha(e) = m$  or both  $Y_\beta(e) = m$  and  $Z_\gamma(e) = m$ .

## 2.5 Distributed Representations of Topics

### 2.5.1 Literature Review.

Generally speaking, topic modeling is a task of natural language processing (NLP) discovering latent semantic structures in a large corpus. Topic modeling can help identify themes or topics such as politics or health within a large volume of text. Distributed representations of topics was used as a central building block in constructing the framework in this thesis via a topic modeling and semantic search algorithm employed here, top2vec [34]. Although the framework did not use the most popular function of the top2vec algorithm (topic modeling), it took advantage of the powerful semantic search function. Since the top2vec algorithm was motivated by improving the topic modeling method, a brief review of four topic modeling methods follows: Latent Semantic Analysis (LSA), Probabilistic Latent Semantic Analysis (PLSA), Latent Dirichlet Allocation (LDA), and distributed representations of topics (top2vec) which is shown in Section 2.5.2.

Popular topic modeling can be traced back to 1990 when Latent Semantic Analysis (LSA) was introduced by Deerwester et al. [35]. LSA lives in vector space and uses eigenvectors and eigenvalues from Singular Value Decomposition to approximate a matrix containing word counts per document. In particular, LSA is approximating any rectangular matrix  $M$  of  $t \times d$  dimension where  $t$  is the terms found in corpus and  $d$  is

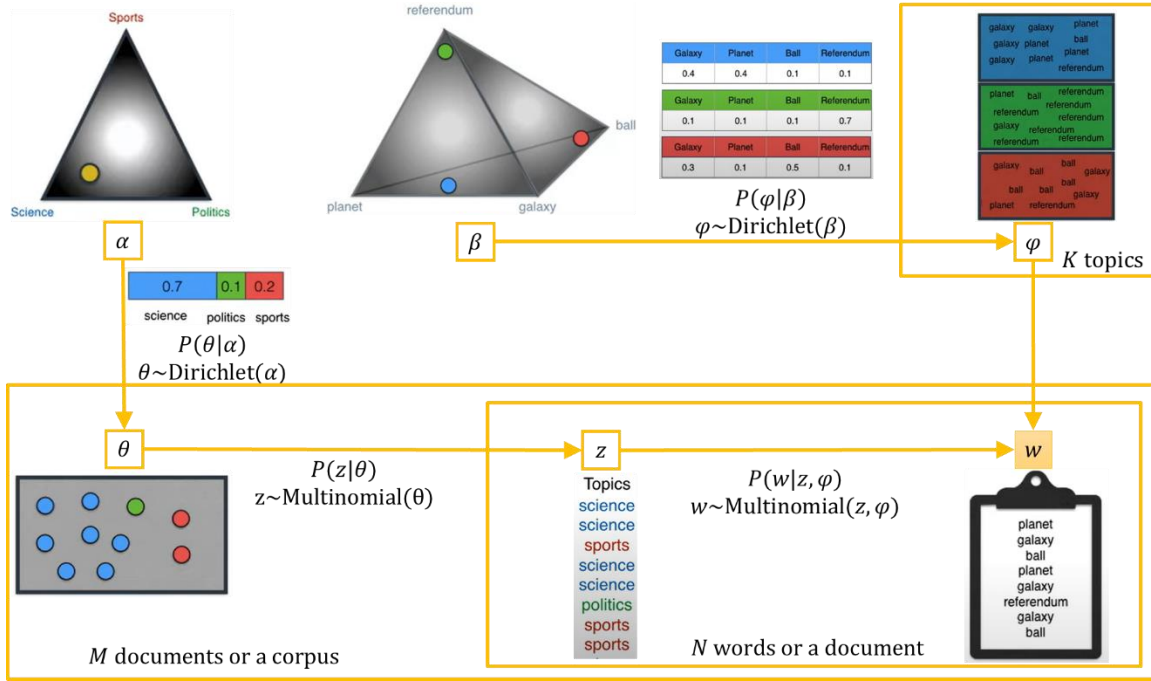
documents via decomposing  $M$  as a product of three matrices:  $M = TSD$ . Such a decomposition is called singular value decomposition because matrices  $T$  and  $D$  have orthonormal columns and  $S$  is diagonal. Furthermore,  $T$  and  $D$  are matrices of left and right singular vectors and  $S$  is the diagonal matrix of singular values. The approximation of  $M$  is accomplished by keeping the first  $k$  highest values of the singular values in the diagonal matrix  $S$  and setting the remaining smaller values to zero. Geometrically, the “rows of the reduced matrices of singular vectors are taken as coordinates of points representing the documents and terms in a  $k$  dimensional [factor] space.” [35] Thus, the approximation of  $M$  has the best possible least-square-fit to  $M$  by choosing an ideal  $k$ , i.e., number of topics being modeled. There are a few limitations of LSA. First, as Deerwester et al. stated in their work, “the choice of  $k$  is critical to our work” as a small value of  $k$  might undermine the real structure of the original dataset and a large value of  $k$  might lead the model to overfit “noise or irrelevant detail in the data.” However, this  $k$ , the choice of dimension or the number of topics, is assumed to be known while in reality it hardly is. Second, the LSA has a polysemy problem in which each polysemy word is only represented as only one point in the vector space. That is, “a word with more than one entirely different meaning (e.g., “bank”), is represented as a weighted average of the different meanings.” Besides dimension and polysemy issue, stemming, phrasal entries, and similarity measure posed as challenges for LSA due to LSA’s nature of representation in raw vector methods. Furthermore, LSA has a strict distribution assumption that words and documents form a joint gaussian model while in practice, a Poisson distribution has been observed instead.

About a decade later, Thomas Hofmann greatly improved LSA to Probabilistic Latent Semantic Analysis (PLSA) [36]. PLSA made the evolution from a vector space to a probabilistic generative model where a document is generated and then that document generates words. A model's parameters are determined by Monte Carlo simulation together with an Expectation/Maximization step used to determine the initial parameters. However, there are a few drawbacks with PLSA, in particular, documents are generated from the existing documents; new documents cannot be generated and thus cannot be estimated.

From 2003 to present, Latent Dirichlet Allocation (LDA) [37] remains a popular method for topic modeling. LDA is a fully generative probabilistic model of a corpus whose documents are represented as random mixtures of latent topics and each topic is represented as a distribution of words. The goal of LDA is to identify components of a corpus with the highest probability of a corpus and documents. LDA generative process is represented in a plate notation [37] fitted to a small dataset is shown in Figure 4. Figure 4 is an illustration of the plate notation where the illustration is inspired by a video overview of LDA by Luis G. Serrano [38]. The orange boxes are “plates” representing replicates which are repeated entities. The bottom outer plate represents documents, while the inner plate represents the repeated word position in a given document, and each position is associated with a choice of topics and words. Suppose we have a corpus consisting of  $M$  number of documents and each document contains  $N$  number of words while  $K$  is a predetermined number of topics. LDA model has two Dirichlet distributions where  $\alpha$  is the parameter of the Dirichlet prior on the document-topic distribution which



is represented as a triangle in the figure and  $\beta$  is the parameter of the Dirichlet prior on the topic-word distribution which is illustrated as a tetrahedron. The parameters  $\alpha$  and  $\beta$  are corpus-level parameters assumed to be sampled once in the process of generating a corpus.  $\theta$  is the topic distribution for a document – document-level variables sampled once per document.  $\varphi$  is the word distribution for a topic – topic-level variables sampled once per topic. The random variable  $z$  follows a multinomial distribution and consists of a list of topics.  $w$  is a list of words where each word is chosen from a multinomial probability conditioned on the topic,  $P(w|z, \varphi)$ . The variables  $z$  and  $w$  are word-level variables and are sampled once for each word in each document. Note that  $w$  is greyed out because words in  $w$  are the only observable variables while all other variables are latent variables. Following the generative process, components of a corpus with the highest probability of a corpus and documents can be identified. As a result, LDA gives more information on the word for each topic. However, LDA assumes the dimensionality  $k$  of the Dirichlet distribution, and thus the dimensionality of the topic variable  $z$ , that is, the number of topics, to be known and fixed while in practice it is rarely known. A newer method, distributed representations of topics by top2vec, addressed this issue and waived the requirement of such an assumption. The next section discusses distributed representations of topics.



**Figure 4. Plate Notation for LDA with Dirichlet-Distributed Topic-Word**

### Distribution Fitted to a Small Dataset

#### 2.5.2 Top2vec.

Different from the probabilistic generative models such as PLSA and LDA, distributed representations of topics by top2vec capitalizes on the well-known distributed representation of documents and words and finds topic vectors in the jointly embedded document and word semantic space [34]. Finding topic vectors is the core function of top2vec, and it requires three existing algorithms and four steps to achieve this goal. Next four paragraphs discuss these four steps: (1) create jointly embedded document and word vectors; (2) create lower dimensional embedding of document vectors; (3) find dense areas of documents; (4) finally calculate the centroid of document vectors, that is, a topic vector, in each dense area.

First, top2vec has three options to learn jointly embedded document and word vectors, one of which used in our research, the Distributed Bag of Words (DBOW) as found in the embedded doc2vec [39] function. The DBOW structure is similar to the word2vec skip-gram model [40] where context word is used to predict its surrounding words within the context window. The difference between DBOW and the skip-gram is that DBOW uses document vector to predict the surrounding words in the context window. In particular, by accessing this feature, the top2vec function first builds on an embedding space where distance between document vectors and word vectors measures their semantic relationships. This semantic relationship is characterized by cosine similarity. Cosine similarity is the cosine of the angle between two vectors; it is also a normalized dot product so that vector magnitude such as word frequency does not affect the cosine similarity score. Therefore, on the semantic space, document vectors cluster closer to each other if they share high semantic similarities and scatter away from each other if they have low similarity scores. Also, the word vectors positioned around document vectors are representative of documents nearby.

The second step of calculating topic vectors is to perform dimension reduction on the jointly document and word embedding semantic space. Within the top2vec function, the Uniform Manifold Approximation and Projection for Dimension Reduction (UMAP) [41] may be used to avoid the curse of dimensionality that sparse document vectors scatter in the high-dimensional semantic embedding space.

The third step is to identify dense areas of documents in the embedded semantic space. A dense area can be identified via Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) [41].

Finally, topic vectors are calculated. So far, a joint document and word embedding semantic space is created, its dimensions is reduced, and density-based clustering is identified. Next, from a dense area where multiple document vectors cluster together sharing a common topic or theme, a topic vector is calculated by taking the arithmetic mean of a dense area's clustering document vectors. Therefore, top2vec finds topic words by recognizing the word vectors located nearest a topic vector via the cosine similarity scores to the topic vector in the embedded space.

## **2.6 Methodologies Applicable to COVID-19 Related Literatures**

Since the dawn of COVID-19, a plethora of research has been conducted worldwide on various topics among the pandemic. Research areas among COVID-19 include, but are not limited to, policing [42] [43] [44], mental health [45] [46] [47], countering misinformation on social media [48], misinformation detection [13] [14] [49] [50] [51], misinformation diffusion [13] [52] [53], and descriptive analysis such as sentiment analysis [13] [14] [54] [55] and topic modeling [13] [56] [57]. Four COVID-19 related works to which the methodologies in this research are applicable are briefly discussed in the next four paragraphs. Specifically, these include a summary of COVID-19 related literatures on topic modeling, sentiment analysis, misinformation diffusion and detection.

Liu et al. investigated the most popular topics among media-directed health communications and timeliness of Chinese media reporting during the first two months of the COVID-19 outbreak in China [56]. Liu et al. analyzed 7791 Chinese media reports collected via a Chinese media content database, WiseSearch, between January 1, 2020, and February 20, 2020. Latent Dirichlet allocation with a prespecified 20 topics was applied to model topics of the media reports. They found the top two most popular themes were prevention and control procedures, and medical treatment and research. They concluded that there was a time lag between Chinese mass media news reporting and the major developments of the spread of the virus.

Basiri et al. conducted a studied on sentimental context of social media posts during the early outbreak of COVID-19 in these eight countries: United States, China, Iran, Italy, Spain, Australia, England, and Canada [54]. Basiri et al. used the Stanford Sentiment140 dataset that classifies a post as expressing either positive sentiment or negative sentiment [58] to train five machine learning base learners: naïve Bayes support vector machines, convolutional neural network, bidirectional gated recurrent network, fastText, and DistilBERT. Then, a stacked generalization mechanism was used to train a meta learner fusing the five base learners. Finally, Basiri et al. collected Twitter data from the eight countries during the timeframe 2020-01-24 to 2020-04-21 and applied the meta learns to the Twitter data. They found that the general Twitter users expressed the highest negative sentiment when infected cases or mortality rate increased. They also found that the highest positive sentiment appeared when the highest recoveries were

reported. As an interesting finding, there was less fluctuations in the percent of Tweets' sentiments for English speaking countries.

Singh et al. is another descriptive analysis of the ecosystem of the information sources relating to COVID-19 shared by Twitter users [52]. They analyzed the network link structure among three groups of information source on Twitter: high-quality health sources (credible), traditional news sources, and low-quality information sources (misinformation). Singh et al. extracted URLs from Twitter posts via hashtags relating to COVID-19 between 2020-01-16 and 2020-04-15 which amounts to 11.2 million Tweets, 1.5 million quotes, and 54.5 million retweets that were shared. Then, they extracted the domains from the URLs and built an information sources network displaying the connections based on the number of times a domain from an URL is shared among the three groups of information sources. Singh et al. found that while posts that share URL whose domain contains misinformation make references to posts that share URL whose domain comes from credible sources and vice versa, misinformation URLs are shared at a greater rate than credible URLs. Also, the highest connectivity of news sources indicates the important role of news outlets made available to information consumers on social media platforms. One limitation that Singh et al. pointed out and should be noted is that the content from a shared URL was not considered when determining the credibility of the information sources. That is, a post is considered credible high-quality health sources solely based upon the domain of the source such as the Centers for Disease Control and Prevention while in this study a post is labeled misinformation if a post contains an URL

that points to a website identified as propagating misinformation specifically related to COVID-19.

Sharma et al. presented a dashboard tracking sentiments, topics, and trends as well as misinformation diffusion and detection on Twitter social media relating to COVID-19 [13]. They collected Twitter posts from March 1, 2020 to June 5, 2020 totaling to 54.32 million of English Tweets worldwide. For misinformation detection, Tweets with URLs were categorized into fake news or not fake news by fact-checking the domain of URLs in the Tweets. In other words, Tweets were classified as fake news based on the news source but not the news content. Out of the fake news category, Tweets were further classified into four subcategories. For misinformation diffusion, original fake news was tracked via retweet/reply in a directed graph. The dashboard offers a few examples of highly circulated fake news regarding fake news' geospatial and temporal tracing along with topics, sentiments and trends on Twitter at a given time.

In conclusion, most publications on social media COVID-19 misinformation detection and diffusion stop at the foundation of classifying fake news and general discussion on dispersion of fake news. Some offered dashboard for visualization of the fake news propagation through time and space. However, to the best of the author's knowledge to date, there has not been any publication on user oriented/interactive system that allows users to search topic of interests relating COVID-19 misinformation on social media. Thus, this research aims to build a human-in-the-loop framework for enabling an interactive process to ingest a human input for a topic of interest then provide both recommended keywords semantically similar to and accurately related to the human input

as well as related documents. Specifically, the human-in-the-loop framework digests any dataset in a form of text through natural language processing. It then takes advantage of a text mining algorithm for topic modeling and semantic search in order to take a user's topic of interest in a form of either keywords or a sentence and return keywords that are semantically similar to the user input topic. The novel aspect of this framework is that it then makes recommendations on the ideal number of keywords as well as identifying such keywords along with each word's probability of being in a target category. The selection of the ideal set of keywords is based on the best classification performance. That is, the ideal set of keywords scores the highest in accurately being contained in the context of documents containing at least one of the ideal set of keywords whose document is correctly identified for a specific targeted category. The framework ideally should work for any dataset comprised of natural language communication; this thesis illustrated the proof of concept and applied this framework to a COVID-19 Twitter dataset.

## **2.7 Summary**

The above methods reviewed in this chapter were integrated to achieve the objectives mentioned in Chapter I. The first realization of integrating natural language processing (NLP), machine learning (ML), and fusion methods was used to address Objective 1 and 2. The results of these two objectives were presented at the IEEE National Aerospace & Electronics Conference which is reprinted with minor revisions in Chapter III. Integration of NLP, ML, and distributed representations of topics were formulated to achieve Objective 3. The full process and results can be found in Chapter



IV which is a planned submission to IEEE Transactions on Computational Social Systems journal.

### **III. Sensor Fusion for Context Analysis in Social Media COVID-19 Data**

#### **3.1 Chapter Overview**

This chapter is a reprint (with minor revisions) of a conference paper presented at the 2021 IEEE National Aerospace & Electronics Conference (NAECON) [14]. This chapter demonstrated the application of natural language processing, machine learning, and fusion in achieving Objective 1 and 2. The entirety of the paper except the Bibliography section begins with the next section. The NAECON paper Bibliography section may be found in thesis supplementary material under Bibliography.

#### **3.2 Abstract**

The growing surge of misinformation among COVID-19 can pose great hindrance to truth, it can magnify distrust in policy makers and/or degrade authorities' credibility, and it can even harm public health. Classification of textual context on social media data relating to COVID-19 is an effective tool to combat misinformation on social media platforms. In this research, Twitter data was leveraged to develop classification methods to detect misinformation and identify Tweet sentiment with respect to COVID-19. Six fusion-based classification models were built fusing three classical machine learning algorithms: multinomial naïve Bayes, logistic regression, and support vector classifier. The best performing models were selected to detect misinformation and to classify sentiment on Tweets that were created during early outbreak of COVID-19 pandemic and the fifth month into pandemic. We found that majority of the public held positive sentiment toward all six types of misinformation news on Twitter social media platform. Except political or biased news, general public expressed more positively toward

unreliable, conspiracy, clickbait, unreliable with political/biased, and clickbait with political/biased news later in the summer month than earlier during the outbreak. The results provide decision or policy makers valuable knowledge gain in public opinion towards various types of misinformation spreading over social media.

Keywords—sensor fusion, sentiment analysis, misinformation analysis, social media, COVID-19

### **3.3 Introduction**

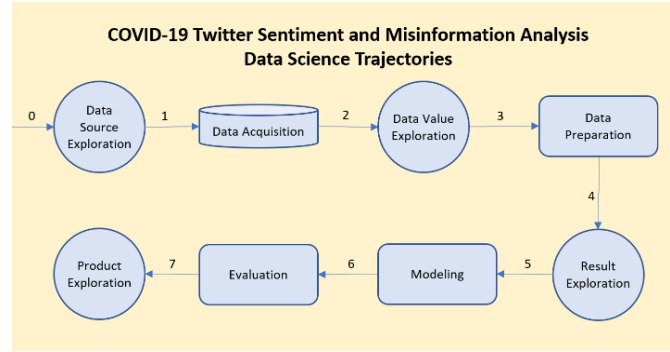
On Feb 2, 2020, the World Health Organization identified information among then Novel Coronavirus as a massive “infodemic” [59]. Nine main themes of COVID-19 disinformation were summarized by [60]. One of the main disinformation themes is medical science such as symptoms, diagnosis, and treatments. Timely and accurate information is crucial in disease control and prevention, especially in a world of instant news and feeds. However, disinformation and misinformation of COVID-19 have been spreading rapidly through social media networks, influencing public’s emotion and action towards certain types of disinformation or misinformation. Combating misinformation on social media platforms is an ongoing battle. Researchers leveraged immense open-source data to conduct various analyses pertaining to COVID-19. Hossain et al. released a dataset containing 6761 expert-annotated Twitter Tweets to support misinformation detection of COVID-19 statements [61]. S. Boon-Itt and Y. Skunkan discovered the public’s awareness and perception of the COVID-19 pandemic with respect to public health through the lens of topic modeling and sentiment analysis on Twitter data [62]. Jang et al. also employed topic modeling for sentiment analysis of the COVID-19

Tweets, but differ in using human-in-loop aspect-based sentiment analysis [63]. Most researchers took advantage of machine learning (ML) algorithms for classification tasks, and some found incorporating sensor fusion to be useful in various ways. Sensor fusion is a process combining various sensory data as a mean to achieve better performance. Basiri et al. developed a novel classification model fusing five ML algorithms using a stacked generalization method to classify sentiment of COVID-19 Tweets in eight countries [54]. The performance of the fusion-based model surpassed all other ML algorithms.

This research seeks to achieve two objectives: 1. Provide knowledge discovery of general public's sentiment toward different types of misinformation regarding COVID-19 news. 2. detect sentiment changes regarding the COVID-19 pandemic and its misinformation over time. To answer these two questions, we first developed two separate classification methods: (1) the classification of sentimental context and (2) the classification of textual context, i.e., misinformation narrative. Six fusion-based classification models were built fusing three classical ML algorithms: multinomial naïve Bayes (NB), logistic regression (LR), and support vector classifier (SVC) with a linear kernel. Six fusion rules based on Boolean mathematical expressions for AND, OR, majority vote, and sensor dominance were applied. We then compared the six fusion-based models among the three ML algorithms and selected the best performing model to be the classification method for sentiment context classification and for misinformation narrative classification. The best performing model was selected based on model accuracy and true positive rate. Lastly, we applied unsupervised ML to two sets of

COVID-19 Tweets employing sentimental context classification and misinformation narrative classification.

Our research procedure followed the data science trajectory (DST) introduced by [64]. Figure 5 is a customized DST for this research. In this paper, section 3.4 covers data source exploration, data acquisition, data value exploration, data preparation, result exploration, and modeling. Section 3.5 goes over evaluation, and section 3.6 finishes with production exploration.



**Figure 5. Customized Data Science Trajectory Inspired by [64] for COVID-19 Twitter Sentiment and Misinformation Analysis**

### 3.4 Methods

#### 3.4.1 Data Source Exploration (Step 0) and Data Acquisition (Step 1).

Due to popularity among the public and readily accessible datasets, social media Twitter Tweets were identified as data source for this study (Step 0). We acquired four Twitter datasets to examine sentiment towards misinformation regarding COVID-19 (Step 1). Dataset (i) has 1.6 million Tweets between April 6 and June 25, 2009, with two sentiment labels: negative and positive. Dataset (ii) contains 39,675 Tweets from March 9 to April 24, 2020, with four COVID-19 narratives, i.e., four misinformation categories: unreliable, conspiracy, clickbait, and political/biased. Dataset (iii) are COVID-19 Tweets

during early outbreak of the pandemic from February 1 to April 29, 2020 and contains no truth labels with respect to sentiment or narrative. Dataset (iv) includes COVID-19 Tweets during a summer month from July 25 to August 29, 2020, again contains no truth labels for sentiment or narrative. Dataset (i) [65], (iii) [66], and (iv) [67] were acquired from Kaggle, an online community of data scientists who share data. Dataset (ii) was retrieved from the Twitter application programming interface (API) service using Tweet ids in [13]. Of the 65,150 source Tweet IDs acquired from [13], only 39,675 Tweets were successfully retrieved from Twitter API due to a large change in Tweet status from public to private or from Tweet deletion.

#### ***3.4.2 Data Value Exploration (Step 2).***

Datasets (i) and (ii) were used to build a sentiment classifier and a misinformation classifier, respectively. Datasets (iii) and (iv) were used as the application for our research objectives. These details follow.

##### ***3.4.2.1. Sentiment dataset or dataset (i).***

A classifier for sentiment was created using dataset (i). We randomly sequestered 20% of the data (320,000 Tweets) as a test dataset and used the remaining 80% (1,280,000 Tweets) as training and validation datasets. Table 1 shows a snippet of the sentiment dataset in which the sentiment column contains the assigned truth label. Positive sentiment is not limited to happy or joyous and can include approval (1<sup>st</sup> Tweet) and somewhat neutral sentiment (3<sup>rd</sup> Tweet). Negative sentiment includes disapproval (2<sup>nd</sup> Tweet) and disappointment (last Tweet). Positive and negative sentiments were split 50/50 for both the test, training and validation datasets. The average length for a Tweet is

13.58 characters with a standard deviation of 7 characters. The 75<sup>th</sup> percentile is 19 characters.

**Table 1. Sample Tweets from Sentiment Dataset**

Sentiment	Tweets
Positive	@viviansessoms Short version - it's like Twitterberry, but BETTER. <a href="http://ubertwitter.com/">http://ubertwitter.com/</a>
Negative	hates prank callers at 10 o'clock in the morning especially when they try to put on an indian accent and they sound scottish/jamacan
Positive	@yateoh Hello twitter noob What phone do u have at the moment? tweet me via web 1st lah
Negative	@shaddih I emailed the billshare author to ask if the site would stay online for a long time, he never wrote back

#### **3.4.2.2. Misinformation dataset or dataset (ii).**


A classifier for misinformation was created using dataset (ii). Following the same 80%-20% data splitting procedure as for dataset (i), the test set for misinformation had 7,935 Tweets and the training/validation sets had 31,740 Tweets. The misinformation dataset labels were generated by [13] using three fact-checking sources: Media Bias/Fact Check [68], NewsGuard [69], and Zimdars [70]. Each Tweet fell into one or more misinformation categories defined as follows.

- Unreliable: Includes false, partially false, rumorous, and/or unverified news.
- Conspiracy: Contains conspiracy theories and false/questionable scientific claims.

- Clickbait: Misleading Tweets to attract attention for reliable or unreliable news.
- Political/biased: Biased Tweets supporting political agendas for reliable or unreliable news.

Each Tweet could be labeled with more than one misinformation category, which resulted in 14 combinations of misinformation labels. Table 2 shows samples for four out of the 14 combination labels. Due to Tweets containing possibly more than one label, we created a misinformation classifier based on two labeling methods: individual label and combined label.

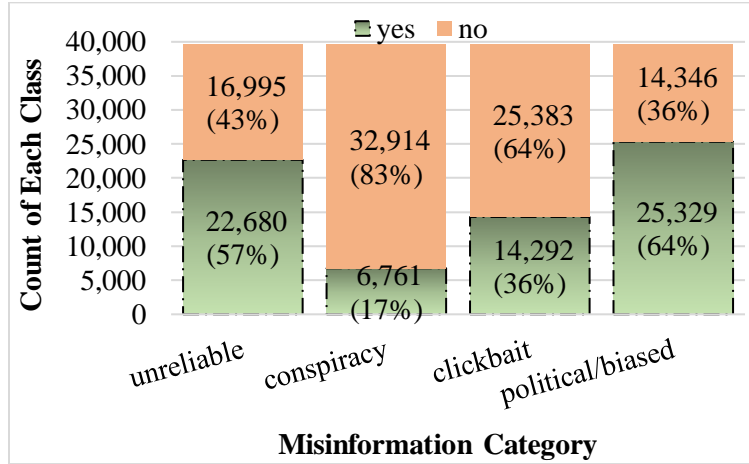
**Table 2. Sample Tweets from Misinformation Dataset**

Unreliable	Conspiracy	Clickbait	Political /Biased	Tweets
unreliable				Imagine that! Democrats lying about a national emergency to try to damage President Trump, and the corporate propaganda media nodding their heads and wiping their chins. Biden repeated the lie at the last debate. Hard to believe!  <a href="https://t.co/zeqNDvXRh7">https://t.co/zeqNDvXRh7</a>
unreliable				'Our hospitals are on their knees': Italian doctor is warning over #coronavirus <a href="https://t.co/vieNqJ2QEG">https://t.co/vieNqJ2QEG</a>



	conspiracy			New York Firefighters Won't Respond To Coronavirus Calls   Zero Hedge <a href="https://t.co/v2JIYl0gW">https://t.co/v2JIYl0gW</a>
	conspiracy			This is very disturbing.\nWhy Is the US Apparently Not Testing for the COVID-19 Coronavirus? - <a href="https://t.co/6DVLpxlNyK">https://t.co/6DVLpxlNyK</a>
unreliable			political/ biased	Coronavirus is exposing Trump's unsuitability to handle a crisis - Washington Examiner <a href="https://t.co/chy4c4fNbd">https://t.co/chy4c4fNbd</a> via @GoogleNews
unreliable			political/ biased	Italy Extends Quarantine to the Entire Country Over Coronavirus\n\n🔥 How to hurt your economy <a href="https://t.co/qfKDHnu2zG">https://t.co/qfKDHnu2zG</a>
		clickbait	political/ biased	Brutal new ad contrasts Trump's coronavirus happy talk with accelerating number of US infections - <a href="https://t.co/cz1hWw1k4M">https://t.co/cz1hWw1k4M</a>
		clickbait	political/ biased	Here's the anti-Trump coronavirus ad we were all eager to see <a href="https://t.co/TWMk9vniOI">https://t.co/TWMk9vniOI</a>

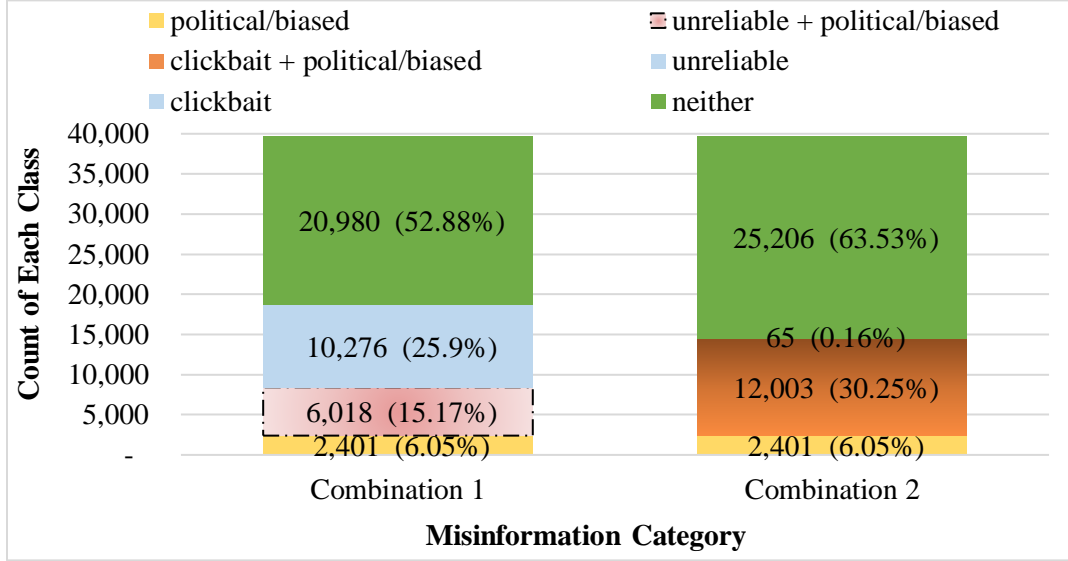
Individual label. For the individual label analysis, we developed four classifiers each designated to classify one of the four misinformation categories. For each classifier, we reconstructed the truth label to be a binary class labeling “yes” if the Tweet contains the type of misinformation category and “no” if it does not. For example, from Table 2 if we want to develop an “unreliable” classifier, we relabel the first two Tweets and the 5<sup>th</sup> and 6<sup>th</sup> Tweets as “yes” and the rest as “no”. Figure 6 displays the binary class counts for each misinformation classifier. Overall, the classes are relatively balanced with a slight skew towards “no” in conspiracy class.



**Figure 6. Tweet Count of the Binary Class for Each Misinformation Category**

Combined label. From the whole dataset (ii), the two highest frequencies of multi-misinformation labels are the pairs of clickbait and political/biased (12,003 Tweets, 30.25%) and unreliable and political/biased (6,018 Tweets, 15.17%). For these two combinations, we developed a classifier to classify four misinformation outcomes: the Tweet is in both misinformation categories, the Tweet is in one but not the other category, or the Tweet is in neither category. Frequency counts of the four outcomes for both combinations are given in Figure 7. For both combinations, more than half the data

is labeled as “neither”, whereas the combined clickbait and political/biased label contains about 25% of the data, but in contrast, the combined unreliable and political/biased label is rarer (< 1%).



**Figure 7. Tweet Count of the Multi-Class for Two Combined Misinformation Categories: Unreliable and Political/Biased, Clickbait and Political/Biased**

### 3.4.3 Data Preparation/Text-preprocessing (Step 3).

Text preprocessing is a crucial step in preparing social media data because the raw text is unstructured and extremely noisy. All four dataset Tweets underwent baseline text-preprocessing (BTP) step which includes using regular expressions to remove non-alphabetic characters and symbols, etc. The first column in Table 3 is a snippet of three original Tweets from dataset (i) and the second column shows examples of our BTP. Depending on the ML algorithm for the sentiment classifier, three additional normalization methods were applied after this baseline text-preprocessing step: Natural Language Toolkit (NLTK) Stop Words, Porter Stemmer [19], and WordNet Lemmatizer [20]. NB used only BTP, LR included the Porter Stemmer processing (Table 3 3<sup>rd</sup>

column), and SVC used all three normalization methods (Table 3 4<sup>th</sup> column). Datasets (ii), (iii), and (iv) used only BTP.

**Table 3. Data Preparation: Pre and Post Text-Preprocessing**

<b>Original Text</b>	<b>Normalized Text – Baseline</b>	<b>Normalized Text – Partial</b>	<b>Normalized Text – Full</b>
Amazing, many of this current &quot;cabinet&quot; appear to have believed that our Tax system is definitely voluntary	amazing many of this current quot cabinet quot appear to have believed that our tax system is definitely voluntary	amaz mani of thi current quot cabinet quot appear to have believ that our tax system is definit voluntari	amaz current quot cabinet quot appear believ tax definit voluntari
blasted internet is soooo slow due to this storm, everything is taking double time to load can't seem to access most of the pages !!!!	blasted internet is soo slow due to this storm everything is taking double time to load cannot seem to access most of the pages	blast internet is soo slow due to thi storm everyth is take doubl time to load can not seem to access most of the page	blast internet soooo slow storm take doubl time load access page
@mattblissett im gutted really i am!	im gutted really am	im gut realli am	im gut realli

#### **3.4.4 Classification Algorithms (Steps 4 and 5).**

##### **3.4.4.1. Feature extraction.**

We considered two ways to extract features for our classification tasks: (1) word counts using Scikit-learn Count Vectorizer [71] and (2) weighted word counts using a measure of how often words appear in Tweets using term frequency inverse document frequency (TFIDF). The NB algorithm was developed using Count Vectorizer feature extraction and both the LR and SVC were developed Scikit-learn TFIDF Vectorizer [72].

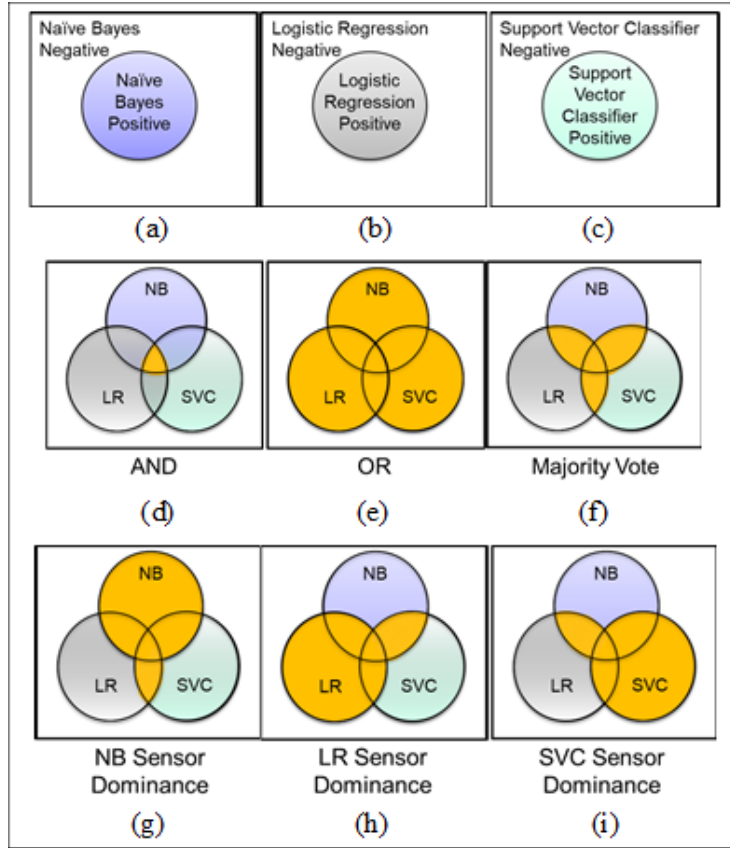
#### **3.4.4.2.      *Algorithms.***

We selected three classical ML algorithms as sentiment and misinformation classifier candidates and compared their performance for dataset (i) and (ii). (1) NB algorithm contains a probabilistic classifier technique selecting a class with the highest computed posterior probability for a given Tweet by applying Bayes' rule with the bag of words assumption and conditional independence assumption. Specifically, we used the Scikit-learn multinomial naïve Bayes (NB) [25] as the classifying algorithm. (2) Like NB, LR contains a probabilistic classifier, but differs in that the classifier is comprised of a set of tunable parameters, weights, and bias. For the LR algorithm, we used Scikit-learn linear model logistic regression [27]. (3) SVC contains a classifier that fits either a linear or nonlinear boundary between classes by expanding the input feature space using kernel functions. For this research, we used Scikit-learn SVC with a linear kernel [28]. We built a pipeline streamlining TFIDF Vectorizer with LR and SVC and found the optimal hyperparameter set using five-fold cross validation in grid search. Elastic net was used as a penalty to prevent overfitting the LR model.

#### **3.4.4.3.      *Fusion rules (Step 5).***

For each sentiment dataset and misinformation dataset, the NB, LR and SVC were used to classify outcomes. We then used Boolean fusion rules to ensemble these three algorithms and evaluated the merit of these rules in prediction performance. There are six fusion rules introduced for this research. Sentiment analysis and the first part (individual labels) of misinformation analysis follow the individual label fusion rules for binary outcomes while the second part (multi-label) of misinformation analysis adhere to combined label fusion rules.

Individual label fusion rules. These rules apply to the binary outcome (yes/no for a given label). Figure 8 illustrates these individual label fusion rules via a Venn diagram. Figure 8 (a) through (c) represent Tweets predicted by each of the three algorithms as positive sentiment and are put inside the circle while negative sentiment is outside of the circle. Figure 8 (d) shows the AND rule which will only predict positive sentiment when all three algorithms predict positive. Figure 8 (e) demonstrates that the OR rule predicts positive sentiment when at least one algorithm predicts positive sentiment. Figure 8 (f) Majority Vote rule predicts positive sentiment when at least two algorithms predict positive sentiment. The last three fusion rules (Figure 8 (g)-(i)) are based on sensor dominance. The prediction takes on the dominating algorithm prediction unless the other two algorithms both disagree. For example, NB sensor dominance fusion rule predicts positive sentiment when NB predicts positive sentiment unless both LR and SVC predict negative sentiment.



**Figure 8. Venn Diagram for Binary Class Fusion Rules Fusing Three Algorithms**

Combined label fusion rules. These rules were used in the second part of the misinformation analysis for the combined labels of (1) political/biased with unreliable and (2) political/biased with clickbait. Recall, each combination produces four labels, therefore, the Boolean rules had to be extended to four potential outcomes using label ordering. Combinations include label = 1 (both unreliable (or clickbait) and political/biased); = 2 unreliable (or clickbait); = 3 political/biased; = 4 neither labels. Table 4 lists five fusion rules and their algorithm predictions for all four labels. The Boolean AND rule and sensor dominance were applied with the same logic as given in the individual label fusion rule, however, two types of OR rules were used depending on

assumed ordering. OR(1) emphasizes predicting unreliable (clickbait) misinformation Tweet over political/biased Tweets while OR(2) has these two reversed. Further, Majority Vote logic became more complicated when all three algorithms disagreed with each other. For example, we assigned a Tweet as a 3 if one algorithm predicted a Tweet as 3, one algorithm predicted a 1, and the third algorithm predicted a 4. The three algorithm predictions would be (3, 1, 4), and order within these predictions does not matter. Thus, (3, 1, 4) results in the same fused label as (1, 3, 4) which is listed in Table 4 under majority vote intersects political/biased.

**Table 4. Fusion Rules for Both Sets of Combined Labels: Political/Biased with Unreliable or with Clickbait**

<b>Fusion</b>	<b>Rules/Notes</b>	<b>1 Both</b>	<b>2 Unreliable (or Clickbait)</b>	<b>3 Political/biased</b>	<b>4 Neither</b>
AND	Predicts a label on which all three algorithms agree; if one or more algorithms does not agree then label neither.	(1, 1, 1) or only 1s.	(2, 2, 2) or only 2s.	(3, 3, 3) or only 3s.	All others except only 1s, 2s, and 3s.
OR(1)	Ordering: 4 Neither < 3 Political/biased < 2 Unreliable (or Clickbait) < 1 both	If there is a 1.	If there is no 1 but there is a 2.	If there are no 1 and 2, but there is a 3.	(4, 4, 4).
OR(2)	Ordering: 4 Neither < 2 Unreliable (or Clickbait) < 3 Political/biased < 1 both	If there is a 1.	If there are no 1 and 3, but there is a 2.	If there is no 1, but there is a 3.	(4, 4, 4).
Majority Vote	At least two algorithms agree on a label. If all three algorithms disagree and if there is (a) no 1, then label 4; (b) no 2, then label 3; (c) no 3, then label 2; (d) no 4, then label 1. Order does not matter inside the parentheses.	(1, 1, 1), (1, 1, 2), (1, 1, 3), (1, 1, 4), (1, 2, 3).	(2, 2, 1), (2, 2, 2), (2, 2, 3), (2, 2, 4), (1, 2, 4).	(3, 3, 1), (3, 3, 2), (3, 3, 3), (3, 3, 4), (1, 3, 4).	(4, 4, 1), (4, 4, 2), (4, 4, 3), (4, 4, 4), (2, 3, 4).



Sensor Dominance	Predict on dominating algorithm label unless the other two algorithms both agree on a different label. First number is the dominating algorithm prediction. (m, n) is a set of the other two algorithm predictions where m, n = 1, 2, 3, 4 and m is not equal to n.	1 (m, n), 1 (1, 1), 2 (1, 1), 3 (1, 1), 4 (1, 1).	2 (m, n), 1 (2, 2), 2 (2, 2), 3 (2, 2), 4 (2, 2).	3 (m, n), 1 (3, 3), 2 (3, 3), 3 (3, 3), 4 (3, 3).	4 (m, n), 1 (4, 4), 2 (4, 4), 3 (4, 4), 4 (4, 4).
------------------	---	---	---	---	---

#### 3.4.4.4. *Performance metrics.*

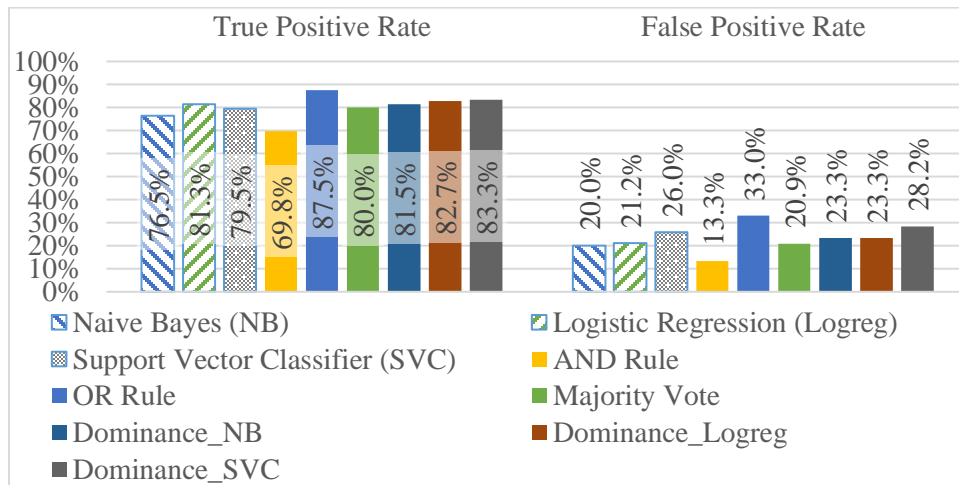
We calculated the true positive rate (TPR), false positive rate (FPR), and accuracy to evaluate algorithm performance. TPR, or hit rate, is the quotient of true positives (TPs) over sum of TPs and false negatives (FNs). The FPR, or false alarm rate, is the FPs divided by the sum of FPs and true negatives (TNs). Accuracy is the sum of TPs and TNs over the total number of observations.

### 3.5 Results (Step 6)

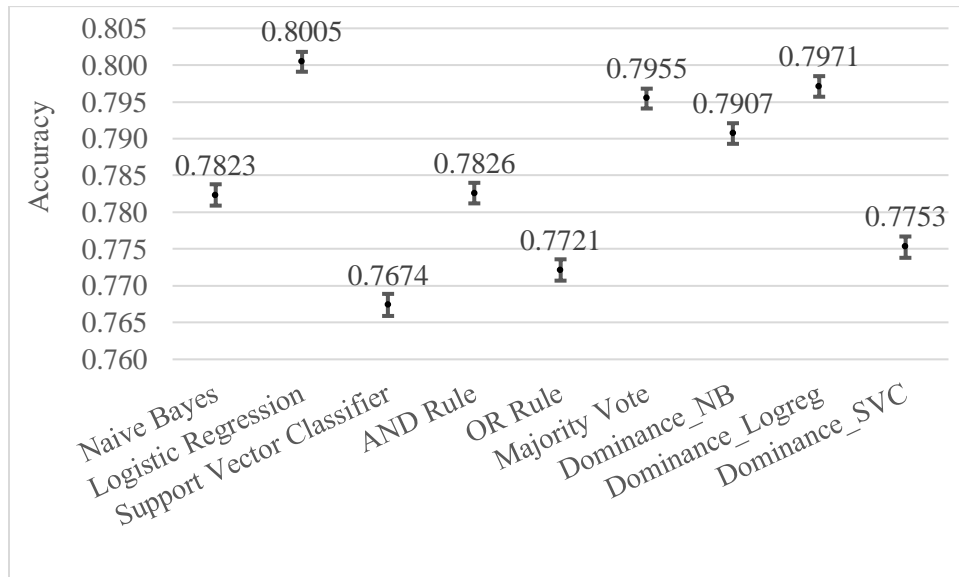
#### 3.5.1 *Sentiment Analysis Results: dataset (i).*

Figure 9 shows the TPR and FPR for sentiment dataset. Without fusion, LR scored the highest TPR (81.3%) and second best FPR (21.2%). NB had the lowest TPR and FPR. Of the fusion rules, the OR rule received the highest TPR, but also the highest FPR whereas the AND rule had the lowest TPR and FPR. If we value the hit rate more than false alarms, four out of six fusion rule algorithms performed better than all three algorithms without fusion. Thus, fusion outperformed individual algorithm in predicting true positives. Figure 10 displays 95% confidence interval (CI) accuracy of predicting sentiment. The three algorithms without fusion all performed fairly well with at least

76% accuracy. The best performing algorithm was LR having 80% accuracy. While SVC performed the worst (76%), it was still better than chance. The six fusion rules improved the NB and SVC algorithms. LR sensor dominance and majority vote were the best performing fusion rules, though, not significantly different. Although comparable, LR alone performed significantly better than any fusion rule in the sentiment analysis.



**Figure 9. True Positive Rate and False Positive Rate for Sentiment Dataset**

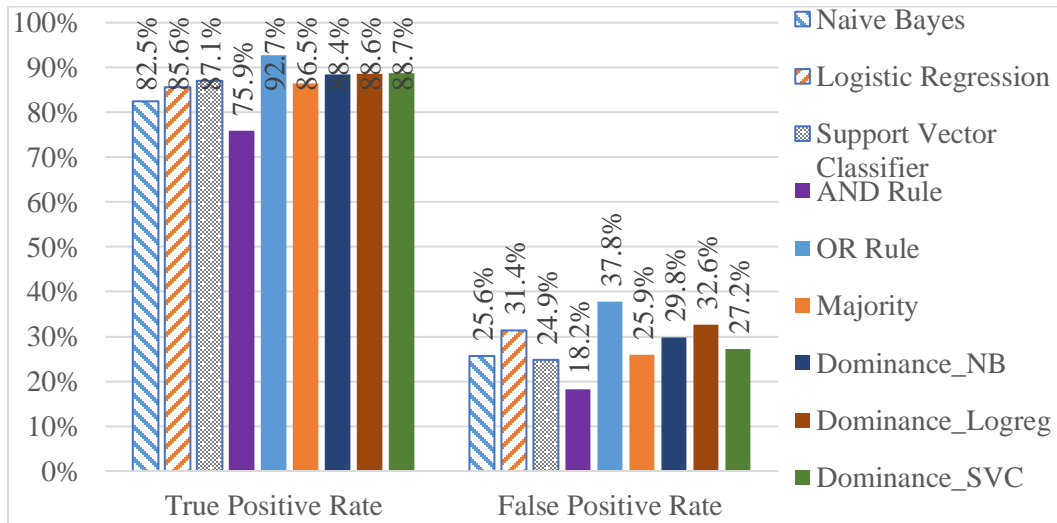


**Figure 10. Prediction Accuracy with 95% Confidence Interval by Algorithm for Sentiment Dataset**

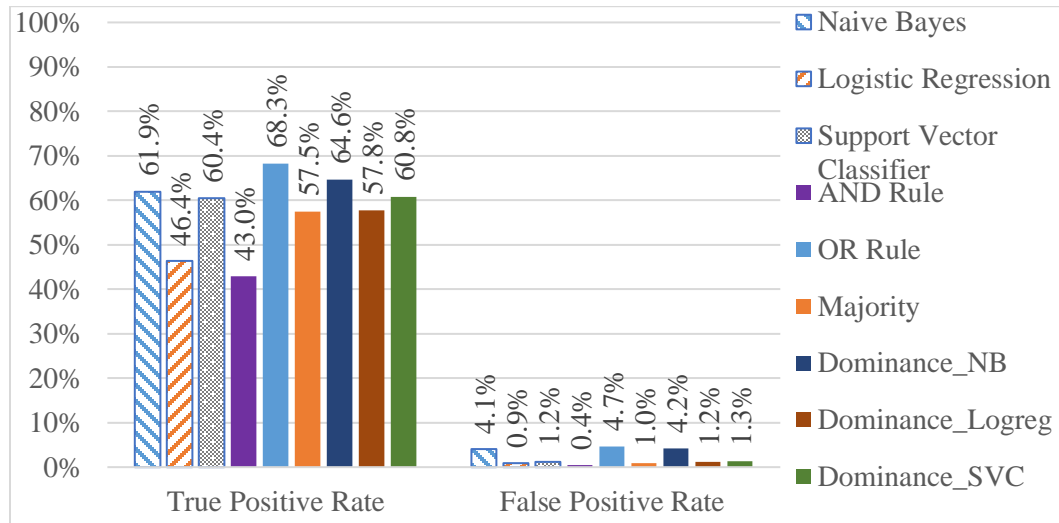
### 3.5.2 Misinformation Analysis Results: dataset (ii).

#### 3.5.2.1. Individual label results.

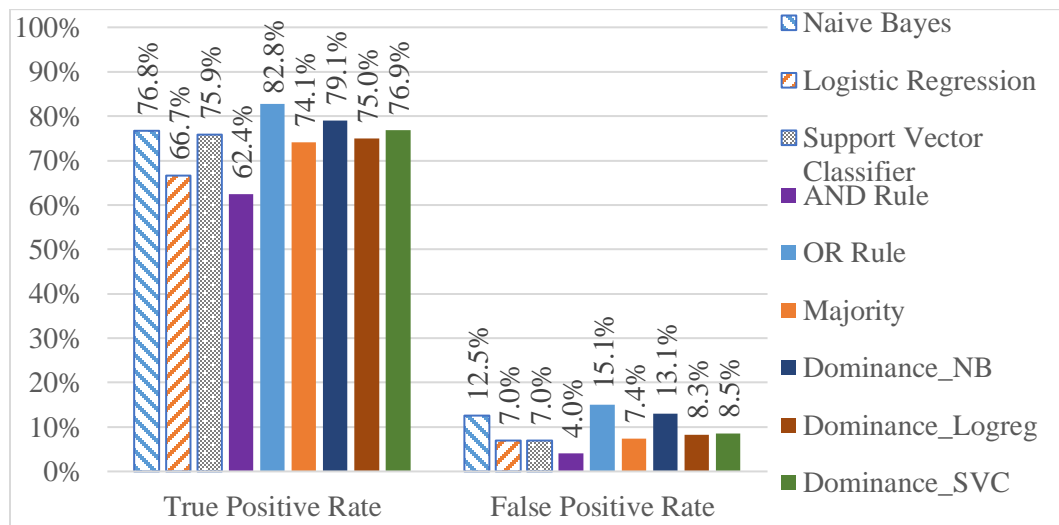
Figure 11 to Figure 14 present the TPR and FPR predicting each individual misinformation label across algorithms. For predicting each type of misinformation, the OR rule performed the best in TPR and the worst in FPR whereas the AND rule performed the best in FPR and the worst in TPR. For predicting unreliable misinformation Tweets, all algorithms scored above 80% in TPR except the AND rule (76%). Conspiracy misinformation, performed the worst holistically across algorithms when compared with predicting other three types of misinformation Tweets with respect to TPRs, however, produced the lowest FPRs overall. For clickbait, all algorithms produced TPRs above 74% except two that produced 66%. Political/biased TPRs were generally the highest across algorithms (above 90%) except for two algorithms (84% and 87%). Figure 15 to Figure 18 illustrate accuracy with 95% CI predicting each individual misinformation label across algorithms. For each plot, SVC performed the best. Notice that most fusion rules improve NB and LR.



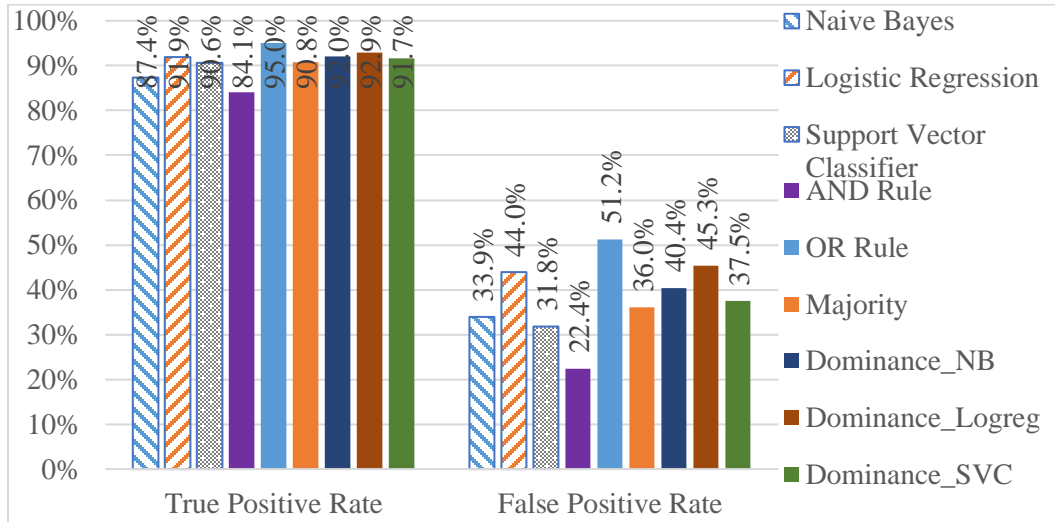
**Figure 11. True Positive Rate and False Positive Rate for Predicting Unreliable**



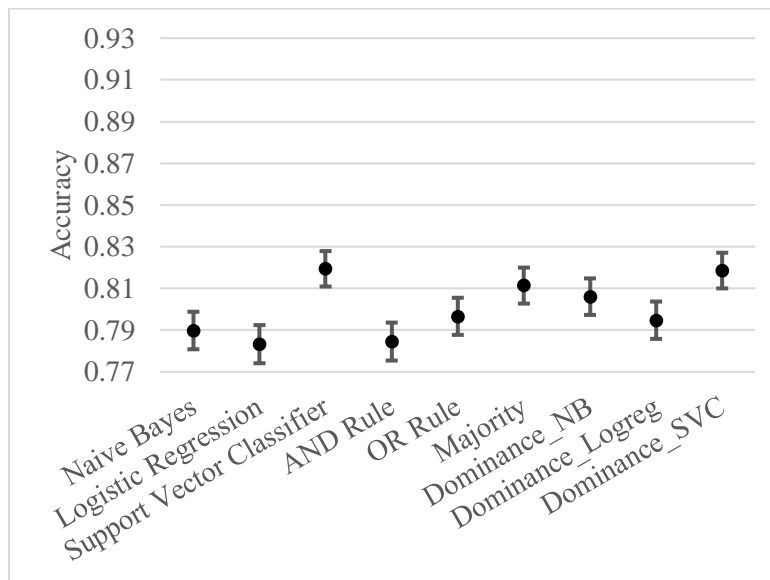
**Figure 12. True Positive Rate and False Positive Rate for Predicting Conspiracy**



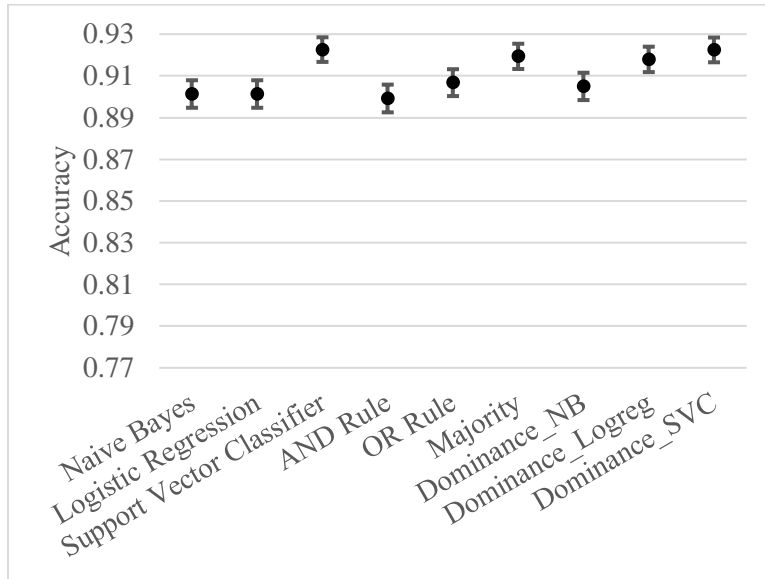
**Figure 13. True Positive Rate and False Positive Rate for Predicting Clickbait**



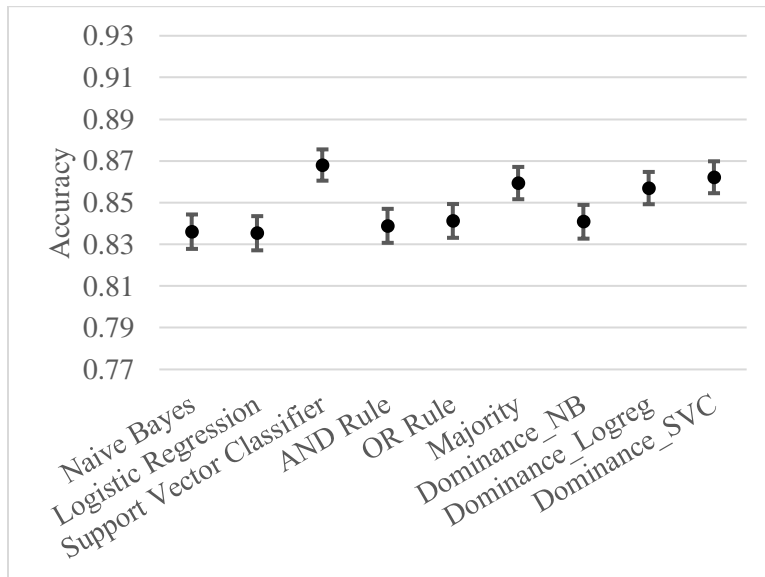
**Figure 14. True Positive Rate and False Positive Rate for Predicting Political/Biased**



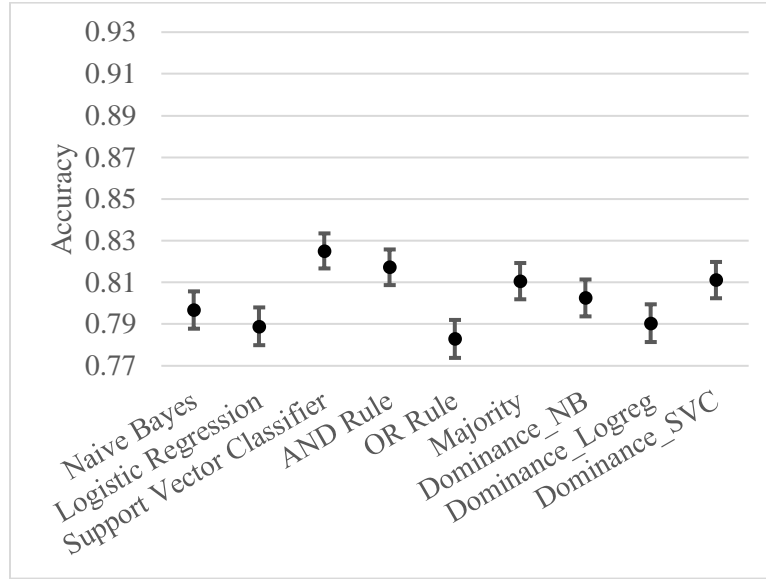
**Figure 15. Accuracy with 95% Confidence Interval for Predicting Unreliable**



**Figure 16. Accuracy with 95% Confidence Interval for Predicting Conspiracy**



**Figure 17. Accuracy with 95% Confidence Interval for Predicting Clickbait**

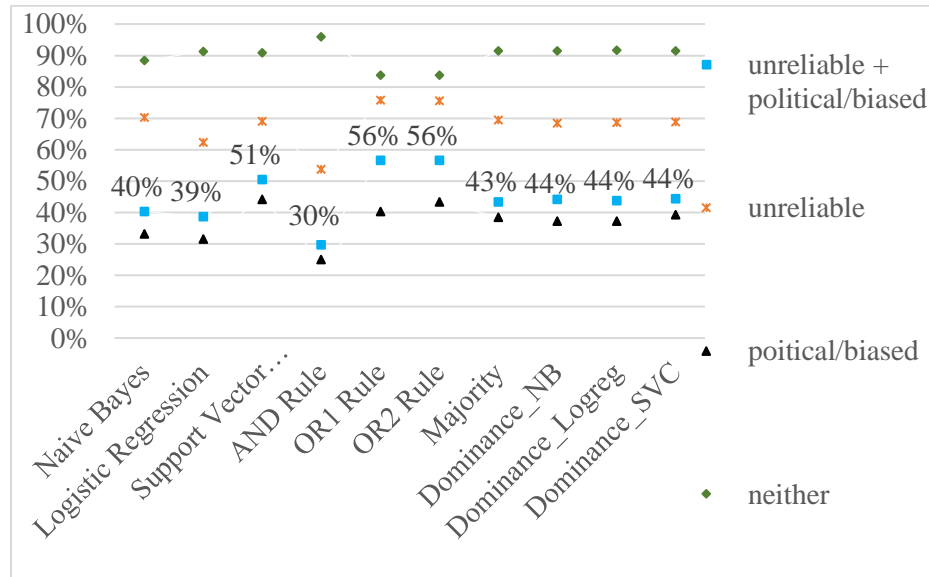


**Figure 18. Accuracy with 95% Confidence Interval for Predicting Political/Biased**

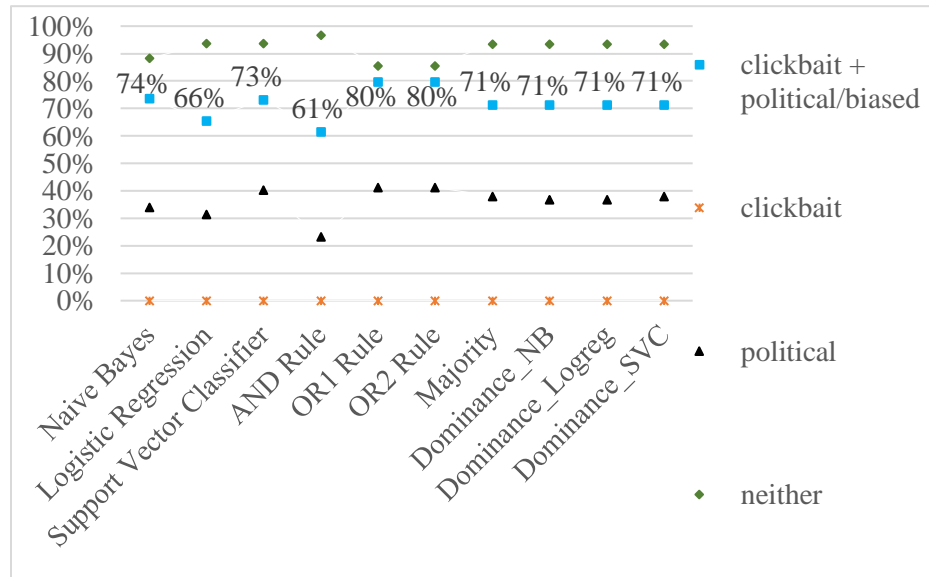
### 3.5.2.2. Combined label results.

Figure 19 and Figure 20 display the class specific TPRs for the four labels by algorithm for combination (1) and (2). Comparing all 10 algorithms, OR(1) and OR(2) fusion rules achieved highest TPR in predicting unreliable with political/biased in Figure 19 and clickbait with political/biased in Figure 20. Figure 21 and Figure 22 show the overall accuracy with 95% CI for predicting four labels in each combination. SVC performed the best among all algorithms in both combinations. For combination in Figure 21, all fusion rules except AND improved NB and LR. For combination in Figure 22, Majority Vote and the three sensor dominance fusion rules performed not significantly different than the best performing SVC algorithm. Though the OR rules scored the second lowest in overall accuracy in Figure 22, but they performed not significantly different than NB, LR, and the AND fusion rule. They also performed the

best in TPR seen in Figure 20. One may consider applying OR fusion rule in the combined labels prediction.

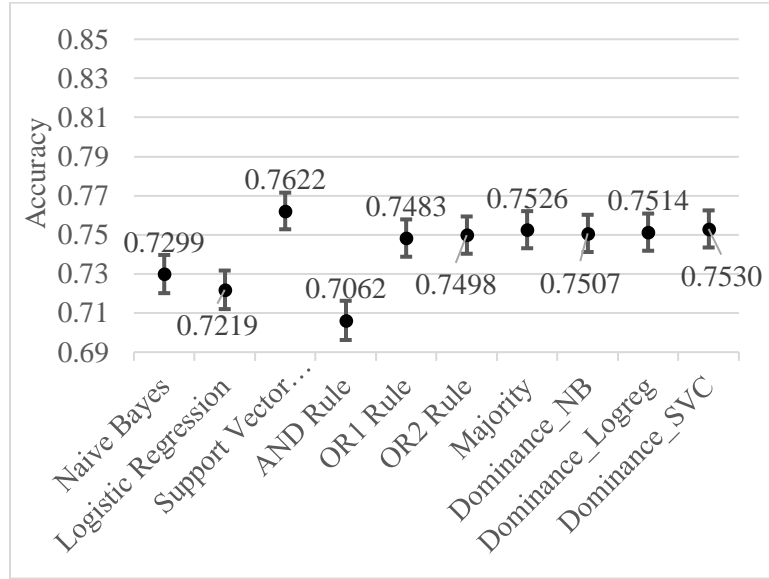


**Figure 19. True Positive Rate for Four Labels by Algorithm for Combinations Unreliable and Political/Biased. Data Labels Are Shown for Combined Misinformation Category (Blue Squares)**

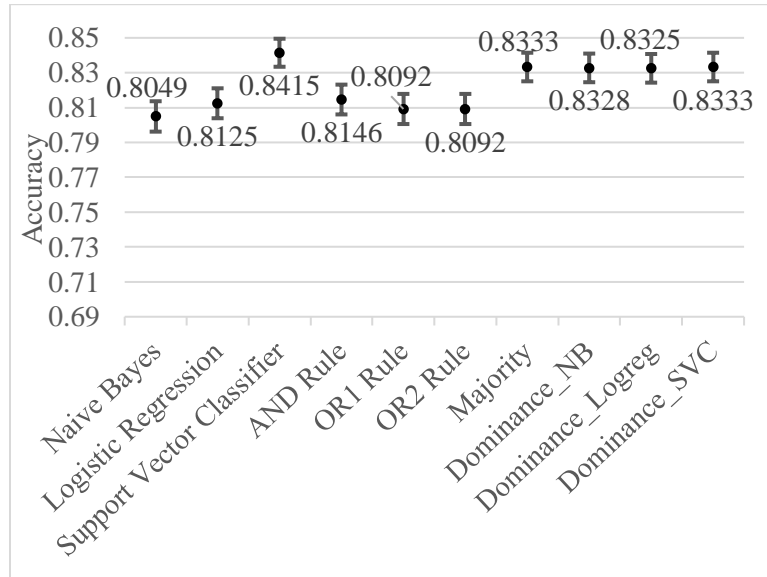


**Figure 20. True Positive Rate for Four Labels by Algorithm for Combinations Clickbait and Political/Biased. Data Labels Are Shown for Combined Misinformation Category (Blue Squares)**





**Figure 21. Overall Accuracy with 95% Confidence Interval for Predicting Combinations Unreliable and Political/Biased**

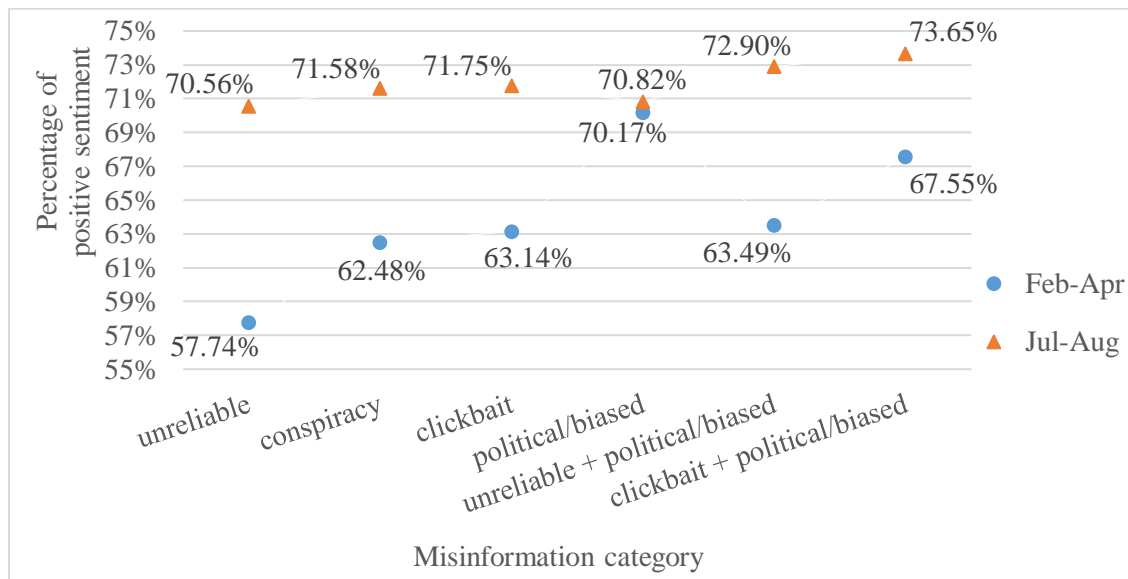


**Figure 22. Overall Accuracy with 95% Confidence Interval for Predicting Combinations Clickbait and Political/Biased**

### 3.5.2.3. Application results (datasets iii and iv).

Figure 23 presents percentage of positive sentiment for each misinformation category. Negative sentiment percentage is not shown but can be calculated as 100%

minus the positive sentiment percentage. By simple majority rule, i.e., over 50%, majority of the general public held positive sentiment towards all misinformation regarding COVID-19 news on Twitter social media for both during the early pandemic outbreak and a summer month in August 2020. For all misinformation category except political/biased, general public expressed more positively later in the summer month than earlier during the outbreak. General public's sentiment towards COVID-19 political or biased type of misinformation news remains relatively unchanged over time.



**Figure 23. Percentage of Positive Sentiment for Each Misinformation Category. Blue Circle is for Early COVID-19 Outbreak from Feb 1 to Apr 29 and Orange Triangle is the 5th Month into COVID-19 from July 25 to Aug 29**

### 3.6 Discussion and Conclusion (Step 7)

Classification of textual context (misinformation narrative) on social media data relating COVID-19 is an effective tool to combat misinformation on social media platforms. We took advantage of the large-scale Twitter data and developed two classification methods to classify sentimental context and misinformation narrative. The

results have been consistently showing fusion-based models can improve classification performance. For all analyses, fusion-based model outperformed the three classical ML algorithms in predicting TPR. Though no fusion-based model scored highest based on accuracy, several fusion-based accuracy scores were not significantly different than that of the best performing ML algorithm. Based on these performance metrics, we selected logistic regression algorithm as sentimental context classification method, support vector classifier as individual misinformation narrative classification method, and OR fusion-based algorithm as combined misinformation narrative classification method. Applying the selected classification methods to COVID-19 Tweets that were created during the early outbreak of the pandemic and the fifth month into the pandemic, we found that majority of the public held positive sentiment toward all six types of misinformation news on Twitter social media platform. It should be noted that positive sentiment includes expression of approval, hope, excitement, and even somewhat neutral in addition to sentiments such as happy or joyous. We also noticed that the over 70% of the public expressed positively towards all misinformation news at the fifth month into the pandemic. Vast majority (>70%) of the public Tweeted most positively toward political or biased misinformation news during the early outbreak of COVID-19, but the percentage of the positive sentiment toward the same misinformation news remained almost unchanged at the latter month. For all misinformation category except political/biased, general public expressed more positively later in the summer month than earlier during the outbreak.

Although most algorithms performed fairly well, there are a couple of ways we can explore in improving algorithm performance. For feature extraction, additional pruning method can help to further reduce the number of features. We also consider adding random forest as well as neural networks to our ML model candidates. Since misinformation categories labels were not distributed evenly with large amount of Tweets were labeled unreliable, we considered expand the training dataset by either regenerating misinformation labels using the existing algorithm or using other labeled dataset such as the one produced by [61].

### **3.7 Acknowledgment**

The views expressed are those of the authors and do not reflect the official guidance or position of the United States Government, the Department of Defense or of the United States Air Force. The material was assigned a clearance of CLEARED AS AMENDED on 21 Sep 2021. Originator Reference Number: 1030AFIT2021ENC09023. Case Number: 2021-0745 (original case number(s): MSC/PA-2021-0388; 88ABW-2021-0797).

## **IV. A Framework for Keywords Identification Via Semantic Analysis in Application to COVID-19 Misinformation on Social Media**

### **4.1 Chapter Overview**

This chapter addressed Objective 3 and is a planned submission to the IEEE Transactions on Computational Social Systems journal. This chapter demonstrated the application of natural language processing, machine learning, and distributed representations of topics in achieving Objective 3, a framework for keyword identification via semantic analysis.

### **4.2 Introduction**

Information consumers are susceptible to misinformation about the COVID-19 pandemic due to the fast-changing information environment [73]. Not only the United States Surgeon General, Dr. Vivek Murthy, recognized that COVID-19 misinformation has been spreading at unprecedented speed and scale [4], but also the World Health Organization (WHO) anticipated the spread of information during the pandemic to be a crisis of its own and characterized the massive flow of information as “infodemic” [5]. Just a few months after the WHO statement on infodemic, in September 2020, the WHO and other international organizations published a joint statement acknowledging that mis- and disinformation among COVID-19 can be harmful to an individual’s health both physically and mentally, misinformation destroys lives, and disinformation polarizes public’s opinions [11]. Dr. Murthy further assessed the impact of health misinformation as a serious threat to public health because it can “cause confusion, sow mistrust, harm people’s health, and undermine public health efforts” [4]. Dr. Murthy specified various

sectors of the society to act upon the call for a whole-of-society effort in confronting COVID-19 health misinformation. Specifically, researchers and research institutions are called to increase vigilance on health questions, concerns, and misinformation via different mediums of information flow and study approaches [4].

Since the dawn of COVID-19 existence, a plethora of research has been conducted worldwide on various topics among the pandemic. Research areas among COVID-19 include, but are not limited to, policing [42] [43] [44], mental health [45] [46] [47], countering misinformation on social media [48], misinformation detection [13] [14] [49] [50] [51], misinformation diffusion [13] [52] [53], and descriptive analysis such as sentiment analysis [13] [14] [54] [55] and topic modeling [13] [56] [57]. Most publications on social media COVID-19 misinformation detection and diffusion stop at the foundation of classifying fake news and general discussion on dispersion of fake news. Some offered dashboard for visualization of the fake news propagation through time and space.

However, to the best of the author's knowledge to date, there has not been any publication on user oriented/interactive process that allows users to search topics of interest relating COVID-19 misinformation on social media. Thus, this paper aims to build a human-in-the-loop framework for enabling an interactive process to ingest a human input for a topic of interest then provide both recommended keywords semantically similar to and accurately related to the human input as well as related documents. Specifically, the human-in-the-loop framework digests any dataset in a form of text through natural language processing. It then takes advantage of a text mining

algorithm for topic modeling and semantic search in order to take a user's topic of interest in a form of either keywords or a sentence and return keywords that are semantically similar to the user input topic. The novel aspect of this framework is that it then makes recommendations on the ideal number of keywords as well as identifying such keywords along with each word's probability of being in a target category. The selection of the ideal set of keywords is based on the best classification performance. That is, the ideal set of keywords scores the highest in accurately being contained in the context of documents containing at least one of the ideal set of keywords whose document is correctly identified for a specific targeted category. The framework ideally should work for any dataset comprised of natural language communication; this thesis illustrated the proof of concept and applied this framework to a COVID-19 Twitter dataset. A diagram of the interactive framework workflow is shown in Figure 24 and is described in detail in the Methodology section.

The next section will provide background on the semantic search algorithm employed in the data application, top2vec. Section 4.4 presents construction of the novel framework. Section 4.5 poses preliminary results of the COVID dataset application. Section 4.6 proposes conclusion and future work.

### **4.3 Background**

The human interactive framework developed in this work is an integration of natural language processing (NLP), classical machine learning (ML), and distributed representations of topics techniques. NLP and ML implementations were inherited from Smith et al. [14]. Distributed representations of topics was used as a central building

block in constructing the framework via a topic modeling and semantic search algorithm employed here, top2vec [34]. Although the framework did not use the most popular function of the top2vec algorithm (topic modeling), it took advantage of the powerful semantic search function. Since top2vec was motivated by improving the topic modeling method, a brief review of topic modeling follows.

Generally speaking, topic modeling is a task of NLP discovering latent semantic structures in a large corpus. Topic modeling can help identify themes or topics such as politics or health within a large volume of text. Four topic modeling methods will be discussed briefly: Latent Semantic Analysis (LSA), Probabilistic Latent Semantic Analysis (PLSA), Latent Dirichlet Allocation (LDA), and distributed representations of topics (top2vec).

Popular topic modeling can be traced back to 1990 when Latent Semantic Analysis (LSA) was introduced by Deerwester et al. [35]. LSA lives in vector space and uses eigenvectors and eigenvalues from Singular Value Decomposition to approximate a matrix containing word counts per document. In particular, LSA is approximating any rectangular matrix  $M$  of  $t \times d$  dimension where  $t$  is the terms found in corpus and  $d$  is documents via decomposing  $M$  as a product of three matrices:  $M = TSD$ . Such a decomposition is called singular value decomposition because matrices  $T$  and  $D$  have orthonormal columns and  $S$  is diagonal. Furthermore,  $T$  and  $D$  are matrices of left and right singular vectors and  $S$  is the diagonal matrix of singular values. The approximation of  $M$  is accomplished by keeping the first  $k$  highest values of the singular values in the diagonal matrix  $S$  and setting the remaining smaller values to zero. Geometrically, the



“rows of the reduced matrices of singular vectors are taken as coordinates of points representing the documents and terms in a  $k$  dimensional [factor] space.” [35] Thus, the approximation of  $M$  has the best possible least-square-fit to  $M$  by choosing an ideal  $k$ , i.e., number of topics being modeled. There are a few limitations of LSA. First, as Deerwester et al. stated in their work, “the choice of  $k$  is critical to our work” as a small value of  $k$  might undermine the real structure of the original dataset and a large value of  $k$  might lead the model to overfit “noise or irrelevant detail in the data.” However, this  $k$ , the choice of dimension or the number of topics, is assumed to be known while in reality it hardly is. Second, the LSA has a polysemy problem in which each polysemy word is only represented as only one point in the vector space. That is, “a word with more than one entirely different meaning (e.g., “bank”), is represented as a weighted average of the different meanings.” Besides dimension and polysemy issue, stemming, phrasal entries, and similarity measure posed as challenges for LSA due to LSA’s nature of representation in raw vector methods. Furthermore, LSA has strict distribution assumption that words and documents form a joint gaussian model while in practice, a Poisson distribution has been observed instead.

About a decade later, Thomas Hofmann greatly improved LSA to Probabilistic Latent Semantic Analysis (PLSA) [36]. PLSA made the evolution from a vector space to a probabilistic generative model where a document is generated and then that document generates words. A model's parameters are determined by Monte Carlo simulation together with an Expectation/Maximization step used to determine the initial parameters. However, there are a few drawbacks with PLSA, in particular, documents are generated

from the existing documents; new documents cannot be generated and thus cannot be estimated.

From 2003 to present, Latent Dirichlet Allocation (LDA) [37] remains a popular method for topic modeling. LDA is a fully generative probabilistic model of a corpus whose documents are represented as random mixtures of latent topics and each topic is represented as a distribution of words. The goal of LDA is to identify components of a corpus with the highest probability of a corpus and documents. LDA model has two Dirichlet distributions:  $\alpha$  is the parameter of the Dirichlet prior on the document-topic distribution;  $\beta$  is the parameter of the Dirichlet prior on the topic-word distribution. The parameters  $\alpha$  and  $\beta$  are corpus-level parameters assumed to be sampled once in the process of generating a corpus.  $\varphi$  is the word distribution for a topic – topic-level variables sampled once per topic. The random variable  $z$  follows a multinomial distribution where the parameter  $\theta$  is the topic distribution for a document – document-level variables sampled once per document.  $z$  consists of a list of topics.  $w$  is a list of words where each word is chosen from a multinomial probability conditioned on the topic,  $P(w|z, \varphi)$ . The variables  $z$  and  $w$  are word-level variables and are sampled once for each word in each document. In addition, distributions of words are not only generated by each topic, but also generated by the whole corpus. Thus, LDA gives more information on the word for each topic. However, LDA assumes the dimensionality of the Dirichlet distribution, that is, the number of topics, to be known and fixed while in practice it is rarely known. A newer method, distributed representations of topics by top2vec, addressed this issue and waived the requirement of such an assumption.

Different from the probabilistic generative models such as PLSA and LDA, distributed representations of topics by top2vec capitalizes on the well-known distributed representation of documents and words and finds topic vectors in the jointly embedded document and word semantic space [34]. Top2vec is a function written in Python that contains several features of topic modelling in one package. Finding topic vectors is the core function of top2vec, and it requires three existing algorithms and four steps to achieve this goal. The next four paragraphs discuss these four steps as applicable to our framework: (1) create jointly embedded document and word vectors; (2) create lower dimensional embedding of document vectors; (3) find dense areas of documents; (4) finally calculate the centroid of document vectors, that is, a topic vector, in each dense area.

First, top2vec has three options to learn jointly embedded document and word vectors, one of which used in our research, the Distributed Bag of Words (DBOW) as found in the embedded doc2vec function. The DBOW structure is similar to the word2vec skip-gram model where context word is used to predict its surrounding words within the context window. The difference between DBOW and the skip-gram is that DBOW uses document vector to predict the surrounding words in the context window. In particular, by accessing this feature, the top2vec function first builds on an embedding space where distance between document vectors and word vectors measures their semantic relationships. This semantic relationship is characterized by cosine similarity. Cosine similarity is the cosine of the angle between two vectors; it is also a normalized dot product so that vector magnitude such as word frequency does not affect the cosine

similarity score. Therefore, on the semantic space, document vectors cluster closer to each other if they share high semantic similarities and scatter away from each other if they have low similarity scores. Also, the word vectors positioned around document vectors are representative of documents nearby.

The second step of calculating topic vectors is to perform dimension reduction on the jointly document and word embedding semantic space. Within the top2vec function, the Uniform Manifold Approximation and Projection for Dimension Reduction (UMAP) may be used to avoid the curse of dimensionality that sparse document vectors scatter in the high-dimensional semantic embedding space.

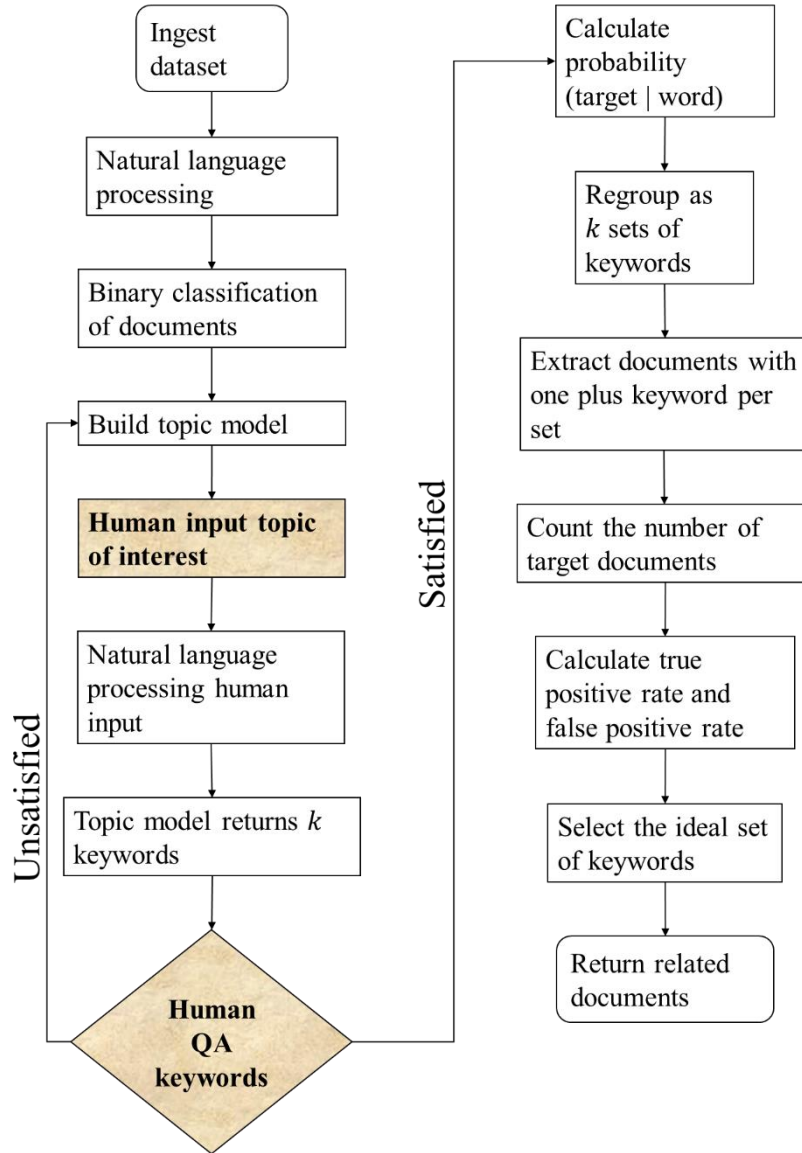
The third step is to identify dense areas of documents in the embedded semantic space. A dense area can be identified via Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN).

Finally, topic vectors may be calculated. So far, a jointly document and word embedding semantic space is created, and its dimensions is reduced, and density-based clustering is identified. Next, from a dense area where multiple document vectors cluster together sharing a common topic or theme, a topic vector is calculated by taking the arithmetic mean of a dense area's clustering document vectors. Therefore, in this process, topic words are recognized via cosine similarity scores of the word vectors located nearest a topic vector in the embedded space.

#### **4.4 Methodology**

This section will discuss each step of the human interactive framework workflow as shown in Figure 24. The novel part of building this framework lies in a few steps.

First, the framework processes misinformation classification through categorizing a document as a specific type of misinformation. Then, the first human interaction with the framework is entering an inquiry which is illustrated in Figure 24, the greyed-out rectangular box. The topic model (top2vec) returns  $k$  keywords ( $k = 50$  keywords were produced by top2vec) ranked by their similarity scores. The second human interaction with the framework occurs at deciding if the initial set of  $k$  keywords are satisfactory. This step is shown in the greyed-out diamond decision shape of Figure 24, human qualification assessment of the keywords. A criteria for judging satisfaction could be the quality of keywords. If the human is unsatisfied with the preliminary product, then the framework goes back to refine the topic model and then prompts the human to input an inquiry again. The loop continues until the human is satisfied with the keyword candidates, then it will end the loop and goes to the next step. The framework divides  $k$  keywords into  $k$  sets of keywords where the latter set of two consecutive sets contains one more keyword than the former set. Thus, the number of keywords in the  $k$  sets of keywords ranges from one keyword, two keywords, three keywords, all the way to  $k$  keywords. Next, for each set of keywords, documents containing at least one of those keywords are extracted. Finally, the system recommends an optimal set of keywords along with its mathematical properties for user. Each of the steps of the human interactive framework is now discussed in reference to the dataset we used for application.



**Figure 24. Human-in-the-Loop Interactive Framework Workflow. Human Interaction Occurs in Two Greyed Out Steps**

#### **4.4.1 Ingest Dataset.**

The dataset used in this analysis is the same dataset used for misinformation analysis in Smith et al.’s work [14] which acquired the original dataset from Sharma et al. [13]. Since the publication of Sharma et al. in October 2020 till mid-January 2022, there

were 69 papers cited Sharma et al. according to Google Scholar. Twelve of these papers either erroneously cited Sharma et al., duplicate each other, or published in a non-English language. Of the remaining 57 English papers that correctly cited Sharma et al., none used Sharma et al.’s dataset. Note that Smith et al. was the only work that took advantage of this dataset since Sharma et al.’s publication. The work in this paper continued to expand on Smith et al.’s work using the same dataset.

The dataset was retrieved from the Twitter application programming interface (API) service using tweet ids in [13] between 03-09-2020 and 04-24-2020. Of the 65,150 source tweet IDs acquired from [13], only 39,675 tweets were successfully retrieved from Twitter API due to a large change in tweet status from public to private or from tweet deletion. This set of data also comes with misinformation labels. The misinformation labels were generated using fact-checking sources categorizing each Tweet according to the domain of the URL shared in a Tweet. Each Tweet was labeled one or more of the four misinformation categories: unreliable, conspiracy, clickbait, and political/biased. More detail regarding misinformation labels can be found in [13] and [14].

#### ***4.4.2 Natural Language Processing.***

Natural language processing of text can help reduce noise and improve structure of the text. As the second step in the framework (Figure 24), the dataset underwent baseline text-preprocessing which includes using regular expression to replace emojis and smileys with the word “happy” or “sad” accordingly; case folding which converts all words to lowercase; replacing contractions such as “didn’t” with its long form “did not”; removing non-alphabetic characters such as numbers and symbols, non-ASCII characters,

mentions, urls, retweet “RT”, and single letters; replacing punctuations with a space; and replacing three or more identical consecutive letters with two letters. Various normalization methods can be applied after the baseline text-preprocessing step, in particular, stop words removal and lemmatization. We considered NLTK Stop Words list which contains 179 commonly used words that carry little semantic information. There are two types of lemmatizations we applied. First, Porter Stemmer is a process for removing the more common morphological and inflexional endings from words in English [19]. Second, WordNet Lemmatizer removes affixes only if the resulting word is in WordNet dictionary [20]. As a result of WordNet Lemmatizer, plural words such as “bats”, “babies”, and “geese” will be reduced down to its singular form “bat”, “baby”, and “goose”, respectively.

As an additional piece to the NLP step, various NLP tasks were tried in this work as a check point for quality keywords returned by the algorithm, top2vec. In particular, we compared and contrasted these NLP combinations during the fine-tuning stage: no NLP performed (process raw data), baseline NLP, lemmatization, Porter stemming, apply stop words then perform lemmatization, and lastly, apply stop words then perform stemming.

#### ***4.4.3 Binary Classification of Documents.***

Since some Tweets may be labeled with more than one misinformation category, for simplicity, Tweets were relabeled into a binary category either a yes or no regarding its misinformation narrative. That is, a Tweet is labeled a targeted misinformation narrative if the Tweet has that specified misinformation label. Therefore, this created



four sets of classification tasks since each Tweet can be thought of as either a specific target misinformation or not where a target misinformation being one of the four labels. For example, in Table 2, the first two Tweets and the 5<sup>th</sup> and 6<sup>th</sup> Tweets are labeled “unreliable”; the last four Tweets are labeled “political/biased”; the 3<sup>rd</sup> and 4<sup>th</sup> Tweets have only “conspiracy”; and the last two Tweets are also “clickbait”. Figure 6 displays the binary class counts for each misinformation classifier. Overall, the classes are relatively balanced with a slight skew towards “no” in the conspiracy class. The binary classification of documents is important for a later task evaluating and selecting the best performing set of keywords representing the most similar keywords to a human inquiry.

#### ***4.4.4 Build Topic Model.***

Once the training dataset underwent NLP, then the topic model was built. Here, we used specifically the top2vec algorithm which incorporated three preexisting functional algorithms of our specific interest to find topic vectors. This section will discuss how to choose the optimal set of hyperparameters to reach a best performing set of keywords as demonstrated for our dataset.

The first step is to build a jointly embedding semantic vector space. There are four options to achieve this goal. We selected the Bag of Words method to build this space (found in the doc2vec option) due to its ability to learn large and unique vocabulary dataset with better results. Further, doc2vec also trains the model from scratch which is different from the rest of the embedding options. Within doc2vec, training speed and min\_count were two hyperparameters finely tuned for this work. We compared training speeds “learn” and “deep-learn” then selected the “learn” option because it learned better

quality vectors than “fast-learn” option and took less time to train than “deep-learn” option. The `min_count` hyperparameter controls the minimum value of the total frequency of a word. `Min_count` removes rare words with total frequency less than a specified threshold in which higher values remove more rare words and lower values keep more rare words. The values considered for `min_count` are 10, 15, 30, and 50.

The second step of building a topic model is to perform a dimension reduction technique via UMAP. We fine-tuned two hyperparameter of UMAP, `n_neighbors` and `embedding_dimension`. `N_neighbors` is a number of nearest neighbors. It balances preserving global structure vs local structure in low dimension embedding. Lower values focus on local structure hence finding more dense areas, that is, more topics. The values considered for `n_neighbors` are 15, 30, 45, 60, and 75. The `embedding_dimension` is the number of dimensions in the reduced embedding space since UMAP scales well in large datasets with high dimensional data. The default value within the `top2vec` function for `embedding_dimension` is five, but we also considered reducing the embedding dimension to two for visualization and for parameter selection purposes.

The third step is to identify dense areas of document vectors via HDBSCAN. `Minimum_cluster_size` is a critical parameter determining clusters of different density in HDBSCAN. It is a minimum value, i.e., minimum number of documents, to be considered a cluster. Higher value tends to have more noise as the model merges unrelated documents. The numbers considered for this dataset are 15, 30, and 45.

A few combinations of the above hyperparameters are as follows. First, we took every combination of `min_count` with values of 10, 30, 50, `n_neighbors` with values of

15, 45, 75, and `minimum_cluster_size` with values of 15, 45, 75. We then generated models with these combinations, setting the `embedding_dimension` to 2 for visualization and selection purposes. After evaluation, four additional combinations with `embedding_dimension` set to 2 were considered. Initially, the parameters selected were based on the clustering visualization. However, since our human interactive system is concerned about the quality of the keywords, further parameter fine-tuning was considered based on either semantic quality of words or the performance of such words. Therefore, 20 more combinations were added to the hyperparameter fine-tuning task. This 20-combination fine-tuning was reached after the model returned the initial keyword set and is discussed further in Section 4.4.6. Table 5 below shows the combinations considered to fine tune hyperparameters for this section.

#### ***4.4.5 Human Input Topic of Interest and Natural Language Processing***

##### ***Human Input.***

Top2vec has a built-in function for semantic search of similar words to a user inquiry. To personalize this semantic search function, we added NLP to prepare the inquiry such that the inquiry undergoes a consistent NLP as the training dataset. In particular, the framework will normalize and tokenize a user inquiry into words. The best performing top2vec model will identify 50 most semantically similar words to a human inquiry of “COVID start lab.” This initial human inquiry has somewhat domain knowledge due to literature search in this thesis work. Therefore, the initial inquiry has a particular interest in the COVID-19 origin, especially regarding a laboratory. Keep in mind that the training dataset containing Tweets during the late March and mid-April in

the early outbreak of the pandemic. Therefore, the origin of COVID-19 started in a laboratory was considered a conspiracy during that timeframe.

**Table 5. Hyperparameter Fine-Tuning Combinations Part 1**

min_count	n_neighbors	minimum_cluster_size	embedding_dimension
10	15	15	2
10	15	45	2
10	15	75	2
10	45	15	2
10	45	45	2
10	45	75	2
10	75	15	2
10	75	45	2
10	75	75	2
30	15	15	2
30	15	45	2
30	15	75	2
30	45	15	2
30	45	45	2
30	45	75	2
30	75	15	2
30	75	45	2
30	75	75	2
50	15	15	2
50	15	45	2
50	15	75	2
50	45	15	2
50	45	45	2
50	45	75	2
50	75	15	2
50	75	45	2
50	75	75	2
50	45	30	2
50	60	15	2
50	60	30	2
50	60	45	2

#### ***4.4.6 Topic Model Returns $k$ Keywords and Human Test.***

When a user inputs an inquiry that has been normalized, the topic model returns the most similar words to the inquiry by using a word similarity score. In our application, we focused on returning the 50 most similar words,  $\{w_1, w_2, w_3, \dots, w_{50}\}$ . The first word has the highest cosine similarity score indicating that it is most semantically similar to the inquiry. This step is achieved by the topic model (top2vec) taking an average of the input word vectors and returning 50 semantically similar words surrounding that resulting vector. Recall, in Section 4.4.4, this work is concerned with the quality of the keywords returned by the topic model. Twenty combinations of the modeling hyperparameters were added to the hyperparameter fine-tuning task to further improve keyword quality. This step is indicated in Figure 24 “Unsatisfied” decision line going from “Human test” to “Build topic model”. Table 6 shows the combinations used for additional hyperparameter tuning.

#### ***4.4.7 Calculate Probability of Misinformation Class Given a Word.***

Once an initial set of satisfactory keywords is recognized, the system calculates the probability of a target misinformation class given one of the words in the set. For example, if “bioweapon” is in the set of keywords, and if the target misinformation class is conspiracy, then the system calculates the probability of the word “bioweapon” being categorized as conspiracy conditioning on the word. To perform this task, Bayes rule was followed and multinomial naïve Bayes algorithm [25] from Scikit-learn [26] was employed. Specifically, the predict\_proba method from the MultinomialNB module was

used. The conditional probability following Bayes rule shows in Equation (41) and predict\_proba method is formulated as in Equation (42).

**Table 6. Hyperparameter Fine-Tuning Combinations Part 2**

min_count	n_neighbors	minimum_cluster_size	embedding_dimension	normalization
15	50	30	2	raw dataset
15	50	40	2	raw dataset
15	50	50	2	raw dataset
15	50	30	5	raw dataset
15	50	40	5	raw dataset
15	50	50	5	raw dataset
15	60	30	5	raw dataset
15	60	40	5	raw dataset
15	60	50	5	raw dataset
50	15	15	5	raw dataset
50	15	30	5	raw dataset
50	15	45	5	raw dataset
50	60	15	5	raw dataset
50	60	30	5	raw dataset
50	60	45	5	raw dataset
50	60	45	2	baseline
50	60	45	2	lemma
50	60	45	2	stemming
50	60	45	2	stopword & lemma
50	60	45	2	stopword & stemming

$$P(y|x) = \frac{P(y \cap x)}{P(x)} = \frac{P(x|y)P(y)}{P(x|y)P(y) + P(x|y^c)P(y^c)}, \quad (41)$$

where

$$P(x|y) = \frac{N_{yi+\alpha}}{N_y + n\alpha} = \frac{\sum_{x \in T} x_i + \alpha}{\sum_{i=1}^n N_{yi} + n\alpha}, \quad (42)$$

$x$  is a feature or a word,  $y$  is a misinformation class,  $y^c$  is not a misinformation class,  $x_i$  is the  $i$ th feature/word,  $y_i$  is the  $i$ th class,  $n$  is the number of features/words,  $T$  is the

training dataset,  $N_{yi} = \sum_{x \in T} x_i$  is the number of times the  $i$ th feature/word appears in class  $y$  in the training dataset  $T$ ,  $N_y = \sum_{i=1}^n N_{yi}$  is the total number of times each feature/word appear in class  $y$  in the training dataset  $T$ , i.e., word count in class  $y$ ,  $\alpha$  is a smoothing prior accounting for features not present in the training sample to prevent zero probabilities in calculation. The common default value for alpha is  $\alpha = 1$  which is the Laplace smoothing.

#### ***4.4.8 Regroup as $k$ Sets of Keywords.***

After fine-tuning hyperparameters in the topic model, the system selected a set of 50 keywords (our application set the number of keywords to 50) that are similar to a user inquiry and higher quality in semantic meaning. Recall, these 50 keywords were listed in an order of highest similarity score to lowest. That is,  $\{w_1, w_2, w_3, \dots, w_{50}\}$  where  $w_1$  has the highest similarity score to human inquiry. Next, 50 sets of keywords were created such that the first set of keyword contains a keyword with the highest similarity score, the second set of keywords contains two keywords with top two highest similarity scores, so on and so forth, until the 50<sup>th</sup> set of keywords consisting of all 50 keywords with top 50 highest similarity scores. That is,  $\{w_1\}, \{w_1, w_2\}, \{w_1, w_2, w_3\}, \dots, \{w_1, w_2, w_3, \dots, w_{50}\}$ .

#### ***4.4.9 Extract Documents with One Plus Keyword Per Set.***

For each set of keywords(s), the system extracts documents, i.e., Tweets, which contains at least one or more of the keywords in the set. For example, for the first set of keywords, the process identifies the word “official” of having the highest similarity score

among all 50 keywords, then the process searches the normalized training dataset and extracts Tweets with the word “official”. Suppose the second set of keywords has the words “official” and “may”, then the system extracts Tweets having at least one of these two words. That is, the second set of Tweets may contain Tweets with “official” in the Tweet, or with “may” in the Tweet, or both “official” and “may” in the Tweet. As a result, 50 sets of Tweets can be written as  $\{d \in D: w_1 \in d\}, \{d \in D: w_1 \in d \cup w_2 \in d\}, \{d \in D: w_1 \in d \cup w_2 \in d \cup w_3 \in d\}, \dots, \{d \in D: \bigcup_{i=1}^{50} w_i \in d\}$  where  $d$  is a document in the corpus  $D$ .

#### ***4.4.10 Count the number of target documents and Calculate True Positive Rate and False Positive Rate.***

True positive rate (TPR) and false positive rate (FPR) are the performance metrics guiding selection of optimal keyword set to recommend to a user. TPR and FPR are calculated based on confusion matrix which is shown in Figure 25. First, to calculate TPR, the system counts the number of Tweets in each misinformation class for each set of Tweets. Recall, Section 4.4.3 lists four types of misinformation classes. Therefore, for each set of Tweets, there are four values counts where each count is the number of Tweets in each four misinformation classes. When the counts are divided by the number of Tweets in the training dataset size individually, one might think these four decimal values as a proportion of Tweets in a target misinformation class where the target misinformation class is one of the four misinformation labels: unreliable, conspiracy, clickbait, and political/biased. That is, define proportion =  $\frac{TP}{P+N}$  where  $TP$  is the number of true positives for the target misinformation label,  $P$  is the number of Tweets that are



labeled positive for the target misinformation label. One more count is needed to calculate TPR, and Figure 6 provides that count. Figure 6 shows the total number of Tweets labeled in one of the four misinformation classes out of the training dataset. Therefore, prevalence of each misinformation class is shown as the percentage labeled “yes” in Figure 6. Equivalently, prevalence of the target class is given as  $\text{prevalence} = \frac{P}{P+N} = \frac{TP+FN}{P+N}$ . Now, TPR for each misinformation class can be calculated as proportion over prevalence. FPR is calculated similarly to the TPR where proportion is the number of Tweets not in a target misinformation class over training dataset size and the prevalence is the percentage labeled “no” in Figure 6.

<b>Confusion Matrix</b>		Predicted	
		Negative	Positive
Actual	Negative (N)	True Negative (TN)	False Positive (FP)
	Positive (P)	False Negative (FN)	True Positive (TP)

**Figure 25. Confusion Matrix**

#### ***4.4.11 Select the Ideal Set of Keywords and Return Related Documents.***

An ideal set of keywords is selected based on the best classification performance, i.e., this set of keywords scores the highest in accurately being contained in the context of documents containing at least one of the ideal set of keywords whose document is correctly identified for a specific targeted category. In particular, this work set a threshold on FPR of 0.2 and chose the highest TPR among all 50 sets of Tweets. Once that particular set of Tweets is identified, the corresponding set of keywords along with

each word's probability of being in the target category are selected, and the set of Tweets is returned for user's information.

## 4.5 Results

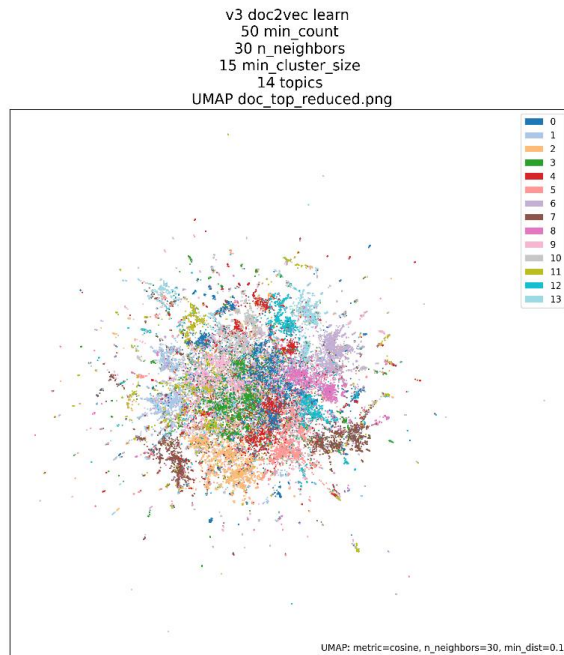
This section discusses two major results: model results from hyperparameter fine-tuning and the final product produced by the fine-tuned model. The best topic model was fine-tuned in three aspects including visualization of the two-dimensional document vector embedding space, quality examination of the initial set of 50 keywords returned by the topic model, and ROC performance between two final models. The best topic model had a high value of `min_count` and `n_neighbors`, medium value of `min_cluster_size`, 2 `bedding_dimension` with training dataset being processed through baseline NLP and lemmatization. An optimal set of keywords along with their probabilities of being classified into one of the misinformation narratives, their word counts, and the model performance metrics were displayed in a figure.

### 4.5.1 *Visualization Determinant.*

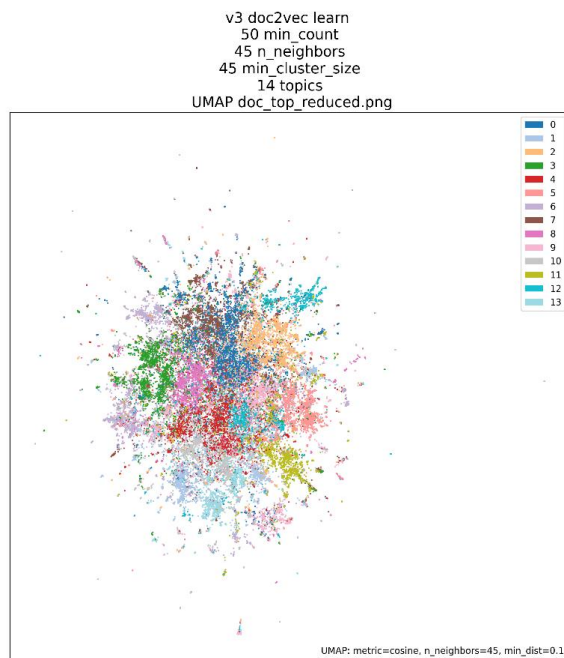
Section 4.4.4 introduced 31 combinations of four topic model hyperparameters which can be viewed in Table 5. Instead of showing all 31 results, the three most competitive model results are shown in Figure 26, Figure 27, and Figure 28. These three results are visualizations of the training dataset under different hyperparameter values. All three models have the following parameter values, 50 `min_count` and 14 topics, and they are different in `n_neighbors` and `min_cluster_size` with values (30, 15), (45, 45), and (60, 45) where the first and second element in parentheses corresponding to `n_neighbors` and `min_cluster_size`, respectively. These figures guided the choice of choosing the level

of these two varying parameters. The selection criteria included representations which resulted in denser, connected, and more clustered representations. In Figure 26, though it is dense, most of the coloring groups are disconnected where they scattered except perhaps topic 2, 6, and 8 where these three coloring groups remain relatively connected and not dislocated from one cluster to the other. Figure 27 is a slight improvement from Figure 26 that fewer coloring groups are disconnected for their main clustering. Yet, topic 6, 9, and 12 seem to form two geographically separated clusters. Both Figure 26 and Figure 27 seem to have larger outliers due to sparse points located on the far upper side, right and bottom directions. This is also the reason why the visualizations look smaller in size compared to the third visualization. Figure 28 might be the most connected relative to the previous two. Only topic 2 is disconnected and dispersed into three minor groupings. Topic 4 is observed to be scattered at the bottom and a few to the right of the figure and maybe several points to the left. The center coloring groups 1, 7, 9, 10, 11 are denser than their surrounding coloring groups, and this model handled noise better than the other two models due to less outliers. To conclude, the initial fine-tuning, hyperparameters in Figure 28 were selected, that is, 50 min\_count, 60 n\_neighbors, 45min\_cluster\_size, and 14 topics.

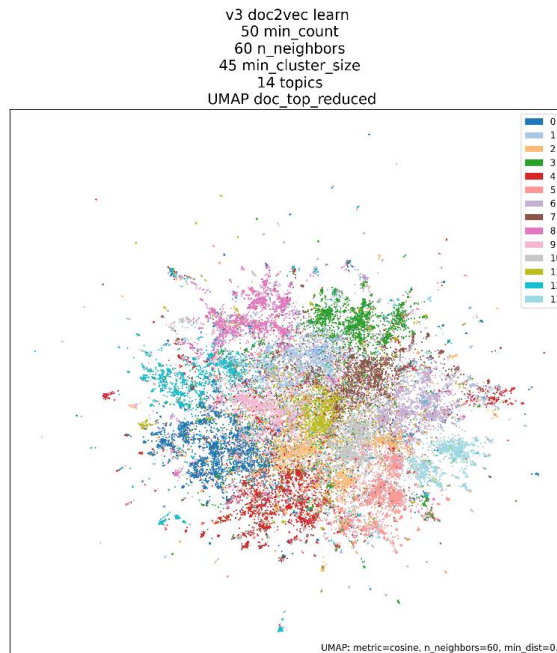
From above, we obtained the first best topic model based on visualization on the connectedness and denseness of the topics. The best set of hyperparameters from initial fine-tuning suggests that a lower level of min\_cluster\_size, medium level of n\_neighbors, and high level of min\_count works better in the training dataset.



**Figure 26. Shape Shows a 300 Dimensional Document Vectors Reduced into 2 Dimensions in UMAP. Colors Indicates Dense Areas Identified by HDBSCAN. Parameters: 30 n\_neighbors and 15 min\_cluster\_size.**



**Figure 27. Shape Shows a 300 Dimensional Document Vectors Reduced into 2 Dimensions in UMAP. Colors Indicates Dense Areas Identified by HDBSCAN. Parameters: 45 n\_neighbors and 45 min\_cluster\_size.**



**Figure 28. Shape Shows a 300 Dimensional Document Vectors Reduced into 2 Dimensions in UMAP. Colors Indicates Dense Areas Identified by HDBSCAN. Parameters: 60 n\_neighbors and 45 min\_cluster\_size.**

#### **4.5.2 Semantic Quality.**

Next, the model was refined with respect to the quality of the keywords returned by top2vec, and a slight change was implemented in the initial parameter set coupled with setting Embedding\_dimensions to five, taking suggestions from top2vec's two training datasets. Column 1 in Figure 29 shows one of the 12 models with five embedding\_dimensions. Most of the keywords in Column 1 have little to no semantic meanings, and other models with five embedding\_dimensions behaved similar to the one shown in Column 1. Thus, five embedding\_dimensions was rejected and two embedding\_dimensions stayed for further examination.

Up to this point in the analysis, all comparisons were done using raw data, that is, there was no NLP for the training data. As seen in Figure 29 Column 2, words such as

“covid-”, “covid\_”, “https”, and “rt” required some basic NLP in order to remove the non-informative characters. The next fine-tuning task falls on NLP where five combinations of NLP were compared and contrasted. Column 3 of Figure 29 shows the keywords returned by top2vec when the training dataset underwent baseline NLP. The light orange color highlights are problematic words. There are three forms of one root word “say” under baseline NLP. Column 4 shows resulting keywords after lemmatization has been performed on the training dataset. Once again, the words “origin” and “originated” have the same root word yet they show up twice in a set of 50 keywords. Column 5 shows the training dataset processed using Porter stemming NLP task. As warned, stemming various tense of words might result in words that are not in English dictionary, such as the ones highlighted in Column 5. Last two columns added NLTK stop word list prior lemmatization or stemming task in hopes of removing most of the low meaning words. As predicted, the settings in Column 7 also has the same issue as words in Column 5 due to stemming task. This leaves the settings in Column 6 which seems superior to all other alternatives. Thus, two NLP options were chosen for comparison in order to generate the most accurate top2vec model. One might note that in the last six models, the word “scientist” or “scientists” highlighted in yellow appeared in all models except the one in Column 6. The fact that removing stop words which in turn reduces the term (or word) dimension results in change of a term vector similarity score. A slight change in term dimension only affects 1/6 of the model results. This might suggest a relatively low sensitivity in term dimension reduction.

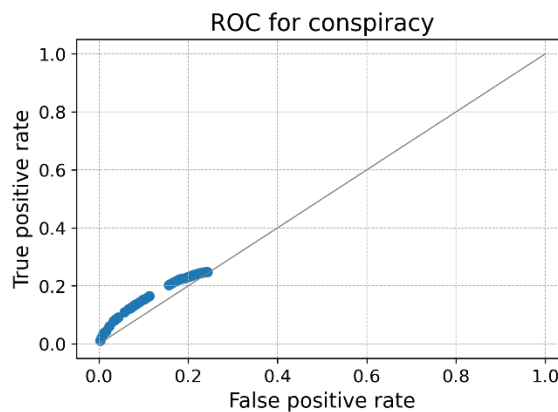
1	2	3	4	5	6	7
Raw data (no NLP)	Raw data (no NLP)	Baseline NLP	Lemma	Stemming	Stop word and lemma	Stop word and stemming
50 min_count	50	50	50	50	50	50
60 n_neighbors	60	60	60	60	60	60
45 min_cluster_size	45	45	45	45	45	45
5 embedding_dimension	2	2	2	2	2	2
top word word score	top word word score	top word word score	top word word score	top word word score	top word word score	top word word score
the 0.514581	human 0.534688	scientists 0.689287	official 0.834781	china 0.827585	icu 0.966344	virus 0.945976
world 0.474776	transmission 0.523938	claims 0.682807	may 0.809038	from 0.809425	lawmaker 0.963094	govt 0.941416
to 0.470502	wuhan 0.520984	human 0.660245	china 0.804913	could 0.790841	due 0.958018	believ 0.930644
rt 0.464707	chinese 0.505601	virus 0.659065	where 0.797299	virus 0.779259	almost 0.951562	california 0.91696
from 0.463909	believed 0.503015	where 0.645537	virus 0.796137	who 0.77645	matter 0.950537	diseas 0.916511
is 0.450505	where 0.496096	bat 0.627263	disease 0.786738	scientist 0.774712	french 0.948529	comment 0.909006
virus 0.437556	originated 0.4942	who 0.624183	information 0.784448	us 0.774022	thousand 0.946413	late 0.906706
it 0.435733	claims 0.488918	chinese 0.619289	which 0.782492	govern 0.766925	potential 0.946259	outbreak 0.90636
that 0.435465	usa 0.48411	transmissio 0.611106	advice 0.777463	evind 0.764886	medicine 0.946083	india 0.903816
wuhan 0.433317	china 0.479922	wuhan 0.606532	yet 0.774749	chines 0.758437	gov 0.946066	warn 0.903294
in 0.431929	suspected 0.478699	ago 0.605273	evidence 0.771874	infect 0.747821	data 0.94403	economi 0.902776
china 0.420441	who 0.478638	claimed 0.598194	wuhan 0.771406	outbreak 0.745989	yes 0.940255	ruli 0.90124
of 0.420237	novel 0.4768	months 0.594808	end 0.770095	wuhan 0.74478	based 0.938568	carri 0.899476
this 0.420149	from 0.455896	evidence 0.594507	say 0.768543	ago 0.734898	trust 0.938418	risk 0.895315
us 0.415314	france 0.452553	china 0.587784	population 0.762768	research 0.724676	person 0.937194	read 0.893884
but 0.411404	origin 0.45214	may 0.584394	chinese 0.762072	two 0.722451	reopen 0.936972	potenti 0.893866
and 0.407236	australia 0.451395	have 0.582774	scientist 0.760866	diseas 0.721498	increase 0.936661	link 0.892539
amp 0.404109	nih 0.449726	since 0.577072	control 0.760658	contain 0.72067	list 0.936177	list 0.890929
they 0.40071	japan 0.446178	weeks 0.576684	italian 0.759921	beij 0.718628	believe 0.935886	exclus 0.890787
with 0.39775	covid— 0.442264	two 0.576478	global 0.759719	may 0.717151	nothing 0.934909	weapon 0.890681
coronavir 0.39483	wuhanvirus 0.439981	originated 0.576181	infection 0.758826	have 0.714393	said 0.934493	total 0.89035
not 0.389889	covid_ 0.439957	scientist 0.574963	community 0.75105	earlier 0.709668	critical 0.93375	dead 0.889608
for 0.387079	came 0.431882	right 0.570256	cause 0.750205	minist 0.703229	place 0.933551	iran 0.888491
are 0.385308	https 0.431493	started 0.569788	show 0.748605	sourc 0.702562	source 0.932257	europ 0.888455
there 0.383069	scientists 0.431293	saying 0.560371	govt 0.748149	compar 0.70108	ship 0.931629	today 0.886653
news 0.382512	developed 0.428445	wuhanvirus 0.559525	top 0.746442	human 0.700993	military 0.93077	good 0.886624
chinese 0.379234	reports 0.427163	bioweapon 0.557564	transmission 0.743655	wa 0.699376	infection 0.930724	scientist 0.88649
who 0.377361	rt 0.425181	expert 0.555856	team 0.741968	shock 0.698628	course 0.930042	administr 0.886194
like 0.376568	mailonline 0.423855	within 0.553376	who 0.740219	which 0.696148	single 0.929786	see 0.885951
as 0.372538	french 0.420725	australia 0.55221	eu 0.7358	commun 0.695144	eu 0.929745	goe 0.883585
we 0.371234	corona 0.419136	earlier 0.549704	started 0.735097	ship 0.688776	community 0.929714	everyon 0.883095
so 0.370396	these 0.419057	outbreak 0.549177	ship 0.73501	one 0.688294	keep 0.929532	shut 0.882867
have 0.364953	not 0.418699	italian 0.546034	power 0.734161	possibl 0.687129	last 0.928671	complet 0.882604
via 0.362534	bioweapon 0.417653	researchers 0.545109	tucker 0.729373	caus 0.686883	also 0.928271	blast 0.881673
says 0.361551	leaked 0.41689	governmen 0.544347	right 0.728765	claim 0.683112	ago 0.927844	wonder 0.881193
usa 0.357972	iran 0.411804	australian 0.540761	high 0.727877	yet 0.682695	follow 0.92702	next 0.881192
no 0.356541	hiv 0.410157	based 0.539212	epidemic 0.724783	origin 0.681765	early 0.926747	gov 0.88119
human 0.355573	virus 0.407664	says 0.535162	infected 0.719657	in 0.681311	sure 0.92626	buy 0.880487
be 0.353661	army 0.405635	humans 0.534715	french 0.719008	month 0.679527	poll 0.925856	increas 0.880473
he 0.353444	in 0.405018	but 0.532707	spike 0.717981	some 0.676309	quarantine 0.923954	china 0.879512
if 0.345681	weeks 0.403361	us 0.529465	around 0.717711	carri 0.672813	propaganda 0.922251	fear 0.87795
at 0.34487	uk 0.402497	say 0.52741	once 0.715094	be 0.67065	move 0.922228	evind 0.87679
why 0.344834	that 0.400673	from 0.527267	reportedly 0.713265	amp 0.669148	dem 0.919607	live 0.875806
people 0.344401	bat 0.40066	were 0.525006	british 0.712041	shut 0.66723	begin 0.91947	enough 0.875543
time 0.335931	symptoms 0.400605	theblaze 0.522389	daily 0.709701	reason 0.667001	show 0.91833	lock 0.875435
now 0.335599	being 0.400288	symptoms 0.521602	originated 0.709287	french 0.666963	released 0.918192	low 0.87509
these 0.333642	but 0.398661	same 0.519925	origin 0.708489	that 0.666861	guy 0.917402	flight 0.874936
all 0.332341	research 0.397673	british 0.51886	in 0.706442	week 0.659739	caused 0.916944	stay 0.87478
on 0.330581	misleading 0.397042	last 0.518412	expert 0.704575	send 0.658288	people 0.91693	absolut 0.874521
work 0.329435	evidence 0.396828	world 0.5168	first 0.701445	been 0.655365	revealed 0.916524	decis 0.872761

Figure 29. Keyword Semantic Quality by Various NLP Tasks

#### 4.5.3 Performance Metric Determinant.

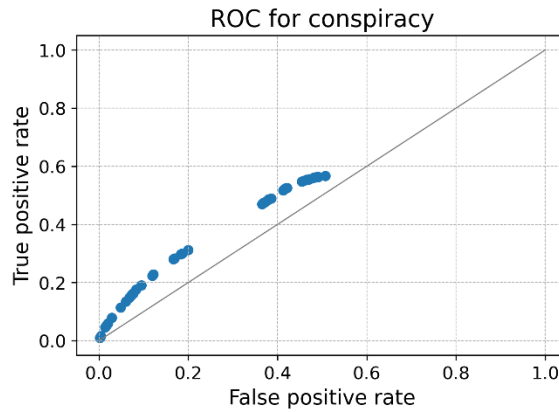
The final two top2vec models were compared by their performance metric, partial ROC curves. Both models have the same top2vec hyperparameter settings but are different in normalization. The partial ROC curves in each misinformation class were compared. Note that partial ROC curves were generated using only a subset of possible

settings and these partial ROC curves were used as an optimal setting could be found without the computational expense of creating the entire ROC curve. Conspiracy misinformation partial ROC curves for both models were chosen as the most drastic comparison. The model with both stop words and lemmatization NLP tasks, shown in Figure 30 did not perform as well as the model on which only lemmatization was performed as shown in Figure 31. The TPR in the latter model is about 30% higher than that of the first model when holding the FPR at a threshold of 0.2. Partial ROC curves in other misinformation classes performed in a relatively similar manner as they did in the conspiracy class for both topic models. Thus, the best top2vec model has lemmatization NLP task performed.



**Figure 30. ROC Curve for Top2vec Model Predicting Conspiracy Employed Stop Words Removal and Lemmatization**



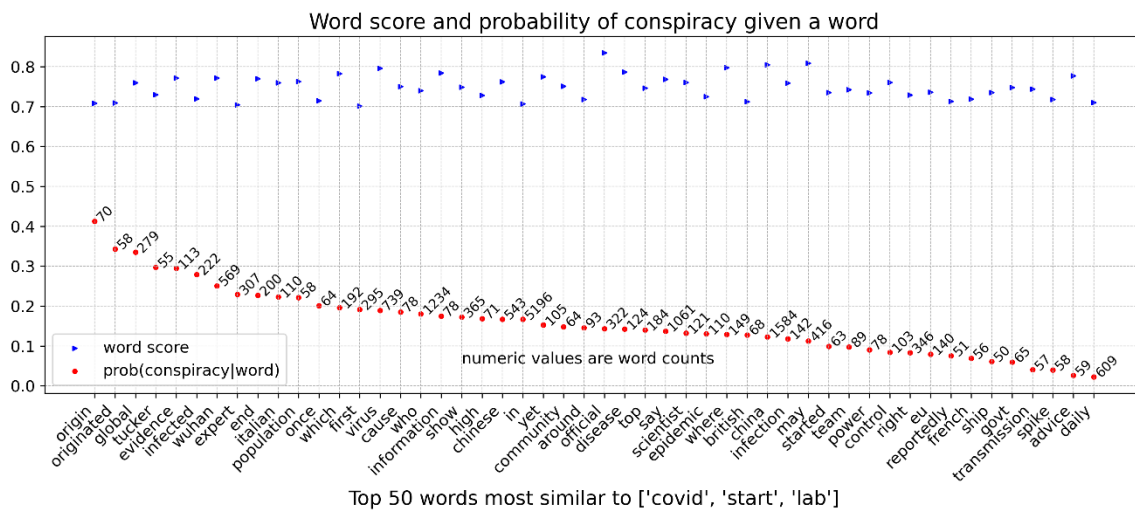


**Figure 31. ROC Curve for Top2vec Model Predicting Conspiracy Employed Lemmatization**

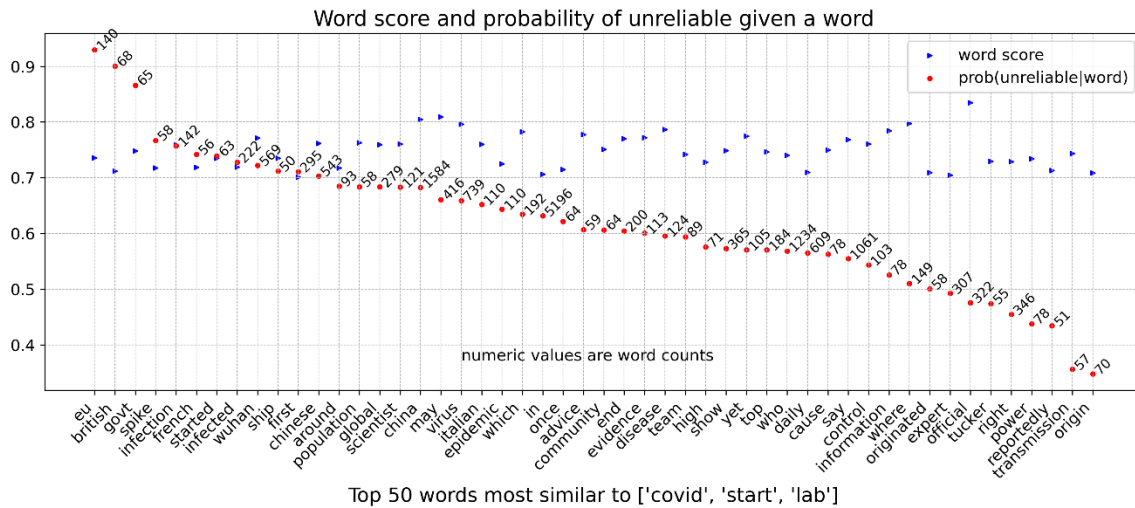
#### ***4.5.4 Best Keyword Set Selection.***

The best performing top2vec model identified the 50 most semantically similar words to a human inquiry of “COVID start lab.” This human initial inquiry has somewhat domain knowledge due to literature search in this thesis work. Therefore, the initial inquiry has a particular interest in the COVID-19 origin, especially regarding a laboratory. Keep in mind that the training dataset contains Tweets during the late March and mid-April in the early outbreak of the pandemic. Therefore, the origin of COVID-19 started in a laboratory was considered a conspiracy during that timeframe. Thus, Figure 32 shows the 50 keywords produced by the most fine-tuned top2vec model in a conspiracy misinformation class. Each word on the x-axis has an associated probability of being categorized as a conspiracy word. The words along the horizontal axis are ranked in a descending order of their conditional probabilities. The integer above the probability is the number of times that word appears in the training dataset. The blue triangle points are the word similarity scores measured by cosine similarity between the

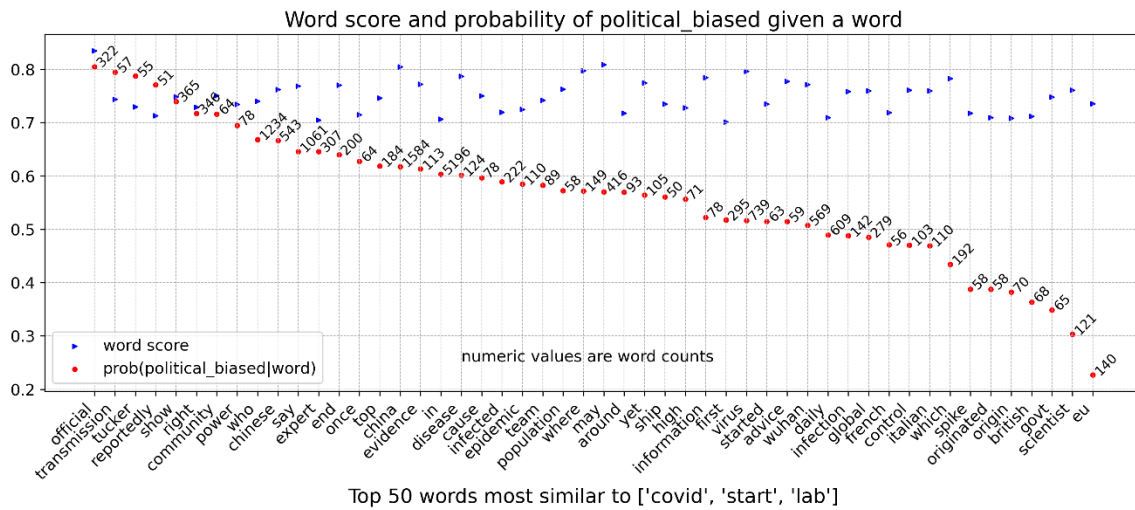
human initial inquiry and each keyword. Not to our surprise, the first two words having the root word “origin” are identified as conspiracy keywords that are most similar to the human inquiry. Keywords in other three misinformation classes are shown in Figure 33, Figure 34, and Figure 35. Note that the similarity score of each keyword in all four figures, Figure 32 to Figure 35, remain constant since the similarity is considered between words and human inquiry.



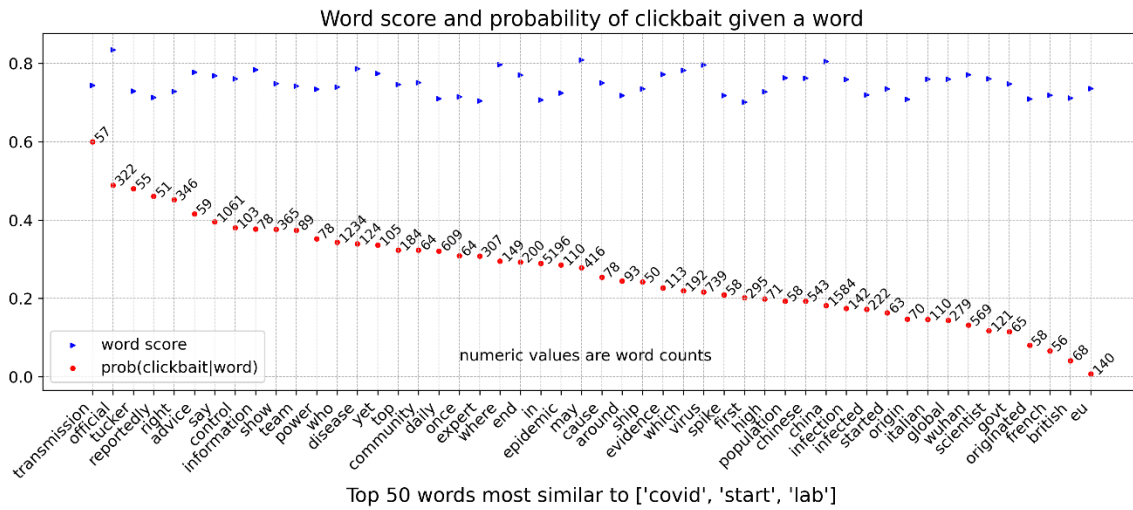
**Figure 32. Top 50 Conspiracy Keywords that Are Most Similar to Words “Covid”, “Start”, and “Lab”**



**Figure 33. Top 50 Unreliable Keywords that Are Most Similar to Words “Covid”, “Start”, and “Lab”**

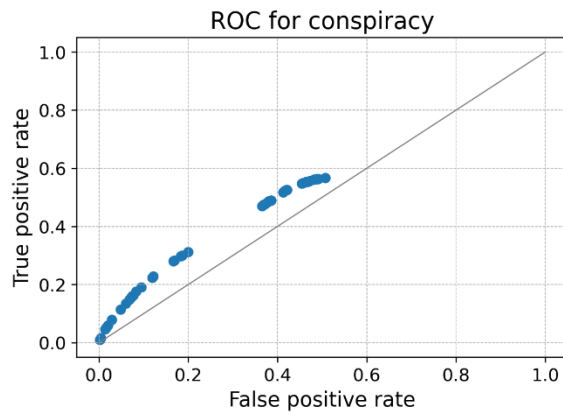


**Figure 34. Top 50 Political/Biased Keywords that Are Most Similar to Words “Covid”, “Start”, and “Lab”**

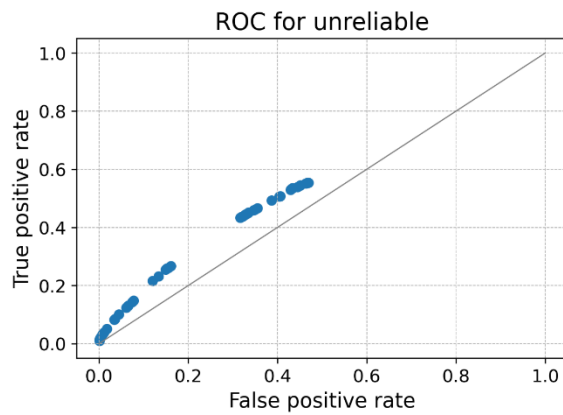


**Figure 35. Top 50 Clickbait Keywords that Are Most Similar to Words “Covid”, “Start”, and “Lab”**

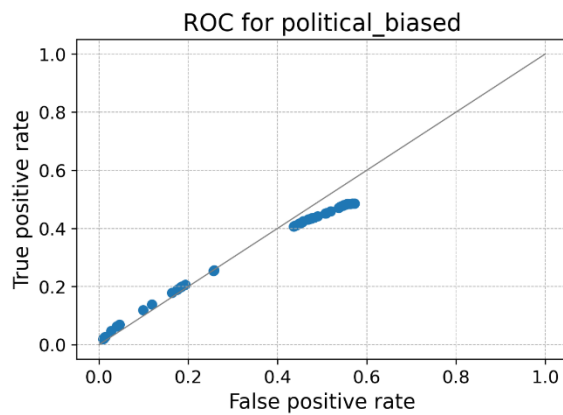
From the above four sets of keywords, for each misinformation class, 50 sets of keywords were created where for every two consecutive sets of keywords, the latter set has one more distinguish word than the previous set. The task at hand is to find the optimal number of keywords in each misinformation class such that Tweets containing at least one of the said set of keywords are identified correctly as being spreading a certain type of misinformation. Partial ROC curves shown from Figure 36 to Figure 39 provide an analytical measure for completing such task. To limit noise, FPR is set to 0.2, and highest TPR was identified. The intersection of maximum FPR that is less than 0.2 and maximum TPR is the number of optimal keywords.



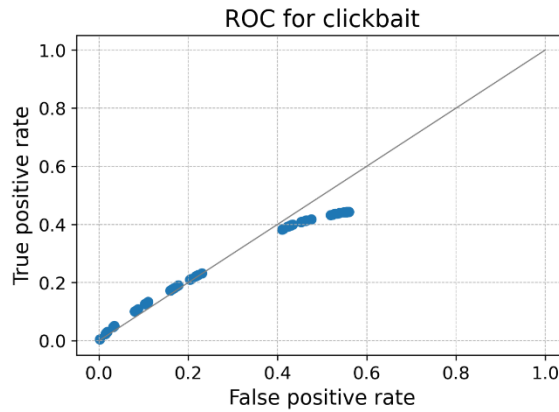
**Figure 36. Conspiracy Misinformation ROC Curve for 50 Sets Keywords**



**Figure 37. Unreliable Misinformation ROC Curve for 50 Sets Keywords**

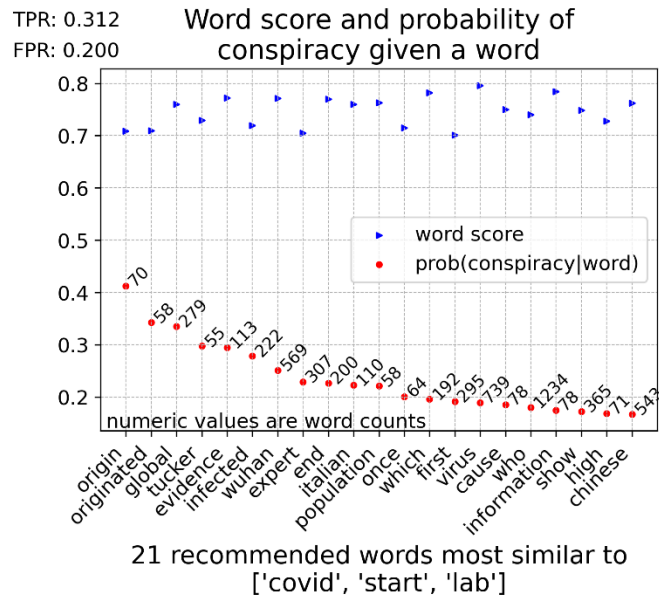


**Figure 38. Political/Biased Misinformation ROC Curve for 50 Sets Keywords**

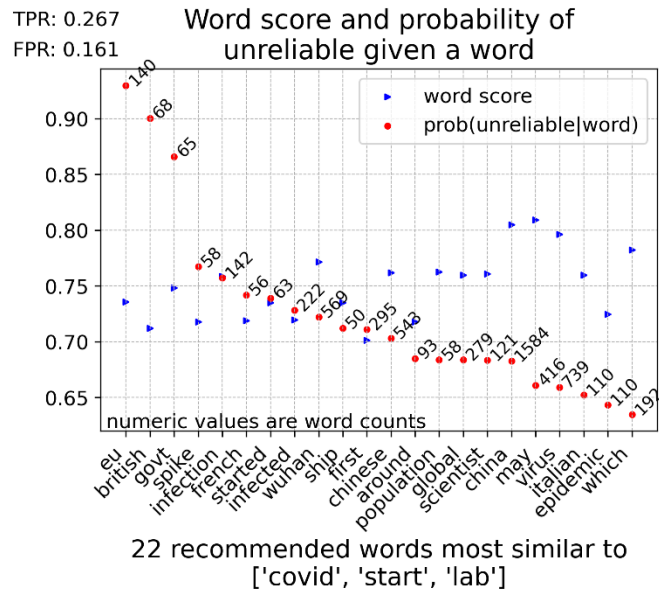


**Figure 39. Clickbait Misinformation ROC Curve for 50 Sets Keywords**

From the partial ROC curves above, it is obvious that the set of 50 keywords is better characterized as conspiracy or unreliable misinformation words rather than the other two misinformation categories. Thus, the recommended keyword set for identifying conspiracy misinformation that is similar to the human initial inquiry of “covid”, “start”, and “lab” contains 21 words that are shown in Figure 40. Figure 40 also displays the TPR of 0.312 and FPR of 0.2 on the upper left corner as the performance for these 21 recommended words. Similar observation was made for the recommended keyword set in the unreliable misinformation class as shown in Figure 41. TPR and FPR for the 22 recommended keywords being in the unreliable misinformation class are 0.267 and 0.161, respectively. Additional observation in the returned recommended keywords for both misinformation classes is that on average, roughly 40% of the keywords are found in both misinformation classes. The distinct keywords are underlined and shown in Table 7.



**Figure 40. 21 Recommended Conspiracy Misinformation Keywords Similar to Human Inquiry “Covid”, “Start”, and “Lab”**



**Figure 41. 22 Recommended Unreliable Misinformation Keywords Similar to Human Inquiry “Covid”, “Start”, and “Lab”**

**Table 7. Recommended Keywords Most Probable and Semantically Similar to Conspiracy and Unreliable Misinformation**

<b>Recommended words most probable class</b>	<b>Recommended words most semantically similar to covid”, “start”, “lab”</b>
<b>Conspiracy</b> (21 words, 57% are distinct)	<u>origin</u> , <u>originated</u> , global, <u>tucker</u> , <u>evidence</u> , infected, wuhan, <u>expert</u> , <u>end</u> , italian, population, <u>once</u> , which, first, virus, <u>cause</u> , <u>who</u> , <u>information</u> , <u>show</u> , <u>high</u> , chinese
<b>Unreliable</b> (22 words, 62% are distinct)	<u>eu</u> , <u>british</u> , <u>govt</u> , <u>spike</u> , <u>infection</u> , <u>french</u> , <u>started</u> , infected, wuhan, <u>ship</u> , first, chinese, <u>around</u> , population, global, <u>scientist</u> , <u>china</u> , <u>may</u> , virus, italian, <u>epidemic</u> , which

#### 4.6 Conclusions

This chapter presented a novel method for a human-in-the-loop interactive framework integrating natural language processing, machine learning, and distributed representations of topics to analytically recommend keywords that are similarly and accurately related to user’s topic of interest. In particular, the interactive framework digests an inquiry in a form of text from a user and systematically produces a set of keywords with the highest true positive rate with no greater than 0.2 false positive rate threshold in model performance metrics for each misinformation class, see Figure 36, Figure 37, Figure 38, and Figure 39. The system is designed for any texted based datasets ideally and is applied to an early COVID-19 Twitter dataset as proof-of-concept research. Figure 24 displays the human-in-the-loop interactive framework workflow for



this paper. An inquiry “COVID start lab” was a topic of interest of a user, in particular, interest of the origin of COVID-19 specifically coming from a laboratory. Keep in mind that the model was trained on a dataset containing Tweets during late March and mid-April in the early outbreak of the COVID-19 pandemic. Therefore, the origin of COVID-19 started in a laboratory was considered a conspiracy during that timeframe. The preliminary results show promising traits of the system. Out of the four types of misinformation classes, unreliable, conspiracy, clickbait, and political/biased, both unreliable (Figure 37) and conspiracy (Figure 36) classes performed better than the other two based on the partial ROC curves. This also indicates the semantic meaning of the user inquiry tends to have a similar conspiracy or unreliable misinformation rather than being in the clickbait or political/biased category. As a final product of the system, two sets of keywords were recommended for user’s information. The interactive framework recommended 21 words that are conspiracy related and most semantically similar to the user inquiry (Figure 40) as well as 22 words recognized to be unreliable and most semantically similar to the user inquiry (Figure 41). Both Figure 40 and Figure 41 further display the probability of each word being classified as a conspiracy and unreliable, respectively, along with each word count and word similarity score. Additionally, performance metrics, true positive rate and false positive rate, for selecting these two sets of keywords are displayed on the top left corner in Figure 40 and Figure 41 accordingly.

For future studies, one may consider a context analysis on the recommended keywords since there are about 40% of the words overlapped. It is also natural to consider a cost analysis for selecting the distinct keywords instead of selecting the whole

recommended set. Lastly, one may also investigate the improvement on a fully automated framework such that domain knowledge from a human is not required.

There are a few limitations and recommendations to this work. First, the evaluation of the best top2vec model in Section 4.5.1 is based on human eye interpretation of denseness and connectiveness of the coloring groups, i.e., topic groups. A numerical evaluation of the visualization is an open research topic. Additionally in the same section, due to resource constrain, three to at most five levels of settings were considered when fine-tuning the top2vec model among the three hyperparameters. Subsequently, Section 4.5.2 also use human interpretation of the semantic quality of the keywords produced by top2vec. Combining the above limitation on evaluation of the top2vec model performance using the Tweeter dataset, one may consider a five-fold cross validation and grid search the most optimal model base on ROC curves performance metric.

Next limitation is on the built-in method within top2vec. We encountered a crucial issue when a user enters an out of vocabulary word in an inquiry, i.e., a word that does not appear in the training dataset, top2vec semantic search function returns an error. Even if a human enters a word that can be found in the training dataset, this same error can still occur. The reason is that the default requirement for the minimum count of a word appearing in the training dataset is 50 times. If a word only appears 49 times, it still won't make the cut and is error bound. This minimum count of a word is a hyperparameter in doc2vec algorithm which can be tuned according to operational

requirements. This work considered different values for the minimum count parameter and found that higher value tends to work well.

As noted in previous work [14], the conspiracy misinformation class is imbalanced for having 17% of the data being relabeled as conspiracy while the other classes achieve at least 36% and some are as high as 64%. Due to the imbalanced nature for the conspiracy class, the resulting keywords produced by the fine-tuned top2vec model are all predicted to be not probable as conspiracy words. All 50 keywords from the conspiracy misinformation class have lower than 0.5 probability being assigned as conspiracy. Yet, the partial ROC for the conspiracy class outperformed all others. This once again proved performance metrics such as ROC or partial ROC is more robust to imbalanced data than is the conditional probability used in this work. But, there are other remedies to address imbalanced data issue. One might consider up-sampling technique which increases the samples in the underrepresented class or minority class. The opposite of up-sampling is down-sampling which removes samples in the overrepresented classes or the majority classes provided that the amount of samples in these classes are abundant and remain sufficient in quantity after reduction. The second solution, Chawla et al. proposed a Synthetic Minority Over-Sampling Technique which combined method of oversampling the minority class and undersampling the majority class [74]. The third solution could be through changing the loss function via weights. When multiplying the loss with the term in which a minority class occurs with a constant value greater than 1, the classifier is forced or encouraged to pay more attention to the minority class. The forth and also the last solution to resolve imbalanced data is another

reweight technique. Cui et al. proposed a reweighting scheme that use the effective number of samples for each class to rebalance the loss which results in a class-balanced loss [75].

The fourth challenge of this work is acquiring a gold standard dataset. This work is based on the dataset curated and labeled by Sharma et al. yet the dataset labels were not labeled by human, rather, via factchecking sources [13]. Therefore, each Tweet with one or more misinformation label was solely dependent on the domain of the URLs shared in the Tweet, and content of the Tweet was not considered in identifying misinformation Tweets. Labeling a post on social media or news article based on domain of the news source is prevalent in recent research works [76][77][78][79]. Micallef et al. recognized that a very small percentage, 10%, of Tweets include external links and hence, developed a novel COVID-19 related dataset including 4,800 Tweets annotated by human such that each Tweet is labeled as one of the three categories: misinformation, counter-misinformation, and irrelevant [80]. However, Micallef et al. dataset was only concerned about COVID-19 misinformation on fake cures and 5G conspiracy theories. Clearly, a gold standard dataset that examines the content of a post is lacking. Furthermore, the standard for COVID-19 misinformation classification is a controversial topic since recommendations and assessments may change over time due to new and updated scientific evidence, according to the United States Surgeon General, Dr. Vivek Murthy [4]. Using the example in this work, the origin of COVID-19 in a laboratory was considered a conspiracy in the beginning of the outbreak in 2020. However, in May of 2021, the U.S. president ordered intelligence community to investigate the origins of

COVID-19 including the theory of the virus potentially coming from a Chinese laboratory [81]. What was once firmly held to be a pure conspiracy theory is now under investigation with inconclusive official conclusions made public [82]. This makes the subject to be exceedingly nebulous, and only serves to add to the level of uncertainty within the general public. Conspiracy theories as such are challenging to categorize into a binary of most likely true versus most likely untrue because of their shifting perception among both the public and the public officials investigating them. This generates an exponentially difficult scenario for scientific researchers to firmly and confidently debunk these statements accurately as time can shift perception of them from most likely untrue to controversial to possibly true as the investigation is ongoing. A solution to this challenge in categorizing the data is to increase transparency in the information made available to the research community and the general public as the president advocates for a “full and transparent accounting” of the pandemic [82].

## **V. Conclusions and Recommendations**

### **5.1 Chapter Overview**

This last chapter of the thesis concludes the results of the research in both Chapter III and Chapter IV, states the importance of the research, and lastly provides recommendations for future research.

### **5.2 Conclusions of Research**

#### ***5.2.1 Conclusion for Sensor Fusion for Context Analysis.***

Classification of textual context (misinformation narrative) on social media data relating COVID-19 is an effective tool to combat misinformation on social media platforms. In Chapter III, we took advantage of the large-scale Twitter data and developed two classification methods to classify sentimental context and misinformation narrative. Specifically, Tweets were classified as either having a positive sentiment or negative sentiment. At the same time, each Tweet was categorized as one of the six categories: unreliable, conspiracy, clickbait, political/biased, unreliable and political/biased, and clickbait and political/biased. The results demonstrate that fusion-based models can improve classification performance. The six Boolean fusion rules used in this work are AND, OR, majority vote, naïve Bayes sensor dominance, logistic regression sensor dominance, and support vector classifier sensor dominance. For all analyses, fusion-based model outperformed the three classical machine learning, multinomial naïve Bayes, logistic regression, and support vector classifier in predicting misinformation by true positive rate performance metric. Though no fusion-based model

scored highest based on accuracy performance metric, several fusion-based accuracy scores were not significantly different than that of the best performing machine learning algorithm. Based on these performance metrics, we selected logistic regression algorithm as sentimental context classification method, support vector classifier as individual misinformation narrative classification method, and OR fusion-based algorithm as combined misinformation narrative classification method. Applying the selected classification methods to COVID-19 Tweets that were created during the early outbreak of the pandemic and the fifth month into the pandemic, we found that majority of the public held positive sentiment toward all six types of misinformation news on Twitter social media platform. It should be noted that positive sentiment includes expression of approval, hope, excitement, and even somewhat neutral in addition to sentiments such as happy or joyous. We also noticed that the over 70% of the public expressed positively towards all misinformation news at the fifth month into the pandemic. Vast majority (>70%) of the public Tweeted most positively toward political or biased misinformation news during the early outbreak of COVID-19, but the percentage of the positive sentiment toward the same misinformation news remained almost unchanged at the latter month. For all misinformation category except political/biased, general public expressed more positively later in the summer month than earlier during the outbreak.

### ***5.2.2 Conclusion for A Human Interactive Process for Recommended***

#### ***Keywords.***

There has not been any publication on a user oriented/interactive process that allows users to search topic of interests relating COVID-19 misinformation on social

media. Thus, Chapter IV presented a novel method for a human-in-the-loop interactive framework integrating natural language processing, machine learning, and distributed representations of topics to analytically recommend keywords that are similarly and accurately related to user's topic of interest. In particular, the interactive framework digests an inquiry in a form of text from a user and systematically produces a set of keywords with the highest true positive rate with no greater than 0.2 false positive rate threshold in model performance metrics for each misinformation class, see Figure 36, Figure 37, Figure 38, and Figure 39. The system is designed for any texted based datasets ideally and is applied to an early COVID-19 Twitter dataset as proof-of-concept research. Figure 24 displays the human-in-the-loop interactive framework workflow for this paper. An inquiry "COVID start lab" was a topic of interest of a user, in particular, interest of the origin of COVID-19 specifically coming from a laboratory. Keep in mind that the model was trained on a dataset containing Tweets during late March and mid-April in the early outbreak of the COVID-19 pandemic. Therefore, the origin of COVID-19 started in a laboratory was considered a conspiracy during that timeframe. The preliminary results show promising traits of the system. Out of the four types of misinformation classes, unreliable, conspiracy, clickbait, and political/biased, both unreliable (Figure 37) and conspiracy (Figure 36) classes performed better than the other two based on the partial ROC curves. This also indicates the semantic meaning of the user inquiry tends to have a similar conspiracy or unreliable misinformation rather than being in the clickbait or political/biased category. As a final product of the system, two sets of keywords were recommended for user's information. The interactive framework



recommended 21 words that are conspiracy related and most semantically similar to the user inquiry (Figure 40) as well as 22 words recognized to be unreliable and most semantically similar to the user inquiry (Figure 41). Both Figure 40 and Figure 41 further display the probability of each word being classified as a conspiracy and unreliable, respectively, along with each word count and word similarity score. Additionally, performance metrics, true positive rate and false positive rate, for selecting these two sets of keywords are displayed on the top left corner in Figure 40 and Figure 41 accordingly.

### **5.3 Significance of Research**

In a joint statement by the DoD before the House Armed Services Committee Subcommittee on Intelligence and Special Operations, the DoD stated that disinformation and misinformation is a critical threat to force protection and recognized that disinformation and misinformation is one of today's greatest challenges not just to the DoD, but also to the U.S. [8]. In January 2020, the nine Combatant Commanders memorandum which known as the "36-star memo" requested increasing intelligence support for "messaging and countering disinformation operations as part of great power competition." The Office of the Under Secretary of Defense for Intelligence and Security and the joint DoD-Director of National Intelligence responded with efforts to support Operations in the Information Environment. The DoD continues to support follow-on initiatives and lines of efforts with focus in Open-Source intelligence. In light of DoD's increasing demand on Open-Source intelligence in combating COVID-19 disinformation and misinformation, this thesis work addressed COVID-19 related questions that may contribute to any on-going efforts that have been put forth by the DoD.

## 5.4 Recommendations for Future Research

For future studies, one may consider a context analysis on the recommended keywords since there are about 40% of the words overlapped. It is also natural to consider a cost analysis for selecting the distinct keywords instead of selecting the whole recommended set. Lastly, one may also investigate the improvement on a fully automated framework such that domain knowledge from a human is not required.

There are a few limitations and recommendations to this work. First, the evaluation of the best top2vec model in Section 4.5.1 is based on human eye interpretation of denseness and connectiveness of the coloring groups, i.e., topic groups. A numerical evaluation of the visualization is an open research topic. Additionally in the same section, due to resource constrain, three to at most five levels of settings were considered when fine-tuning the top2vec model among the three hyperparameters. Subsequently, Section 4.5.2 also use human interpretation of the semantic quality of the keywords produced by top2vec. Combining the above limitation on evaluation of the top2vec model performance using the Tweeter dataset, one may consider a five-fold cross validation and grid search the most optimal model base on ROC curves performance metric.

Next limitation is on the built-in method within top2vec. We encountered a crucial issue when a user enters an out of vocabulary word in an inquiry, i.e., a word that does not appear in the training dataset, top2vec semantic search function returns an error. Even if a human enters a word that can be found in the training dataset, this same error can still occur. The reason is that the default requirement for the minimum count of a

word appearing in the training dataset is 50 times. If a word only appears 49 times, it still won't make the cut and is error bound. This minimum count of a word is a hyperparameter in doc2vec algorithm which can be tuned according to operational requirements. This work considered different values for the minimum count parameter and found that higher value tends to work well.

As noted in previous work [14], the conspiracy misinformation class is imbalanced for having 17% of the data being relabeled as conspiracy while the other classes achieve at least 36% and some are as high as 64%. Due to the imbalanced nature for the conspiracy class, the resulting keywords produced by the fine-tuned top2vec model are all predicted to be not probable as conspiracy words. All 50 keywords from the conspiracy misinformation class have lower than 0.5 probability being assigned as conspiracy. Yet, the partial ROC for the conspiracy class outperformed all others. This once again proved performance metrics such as ROC or partial ROC is more robust to imbalanced data than is the conditional probability used in this work. But, there are other remedies to address imbalanced data issue. One might consider up-sampling technique which increases the samples in the underrepresented class or minority class. The opposite of up-sampling is down-sampling which removes samples in the overrepresented classes or the majority classes provided that the amount of samples in these classes are abundant and remain sufficient in quantity after reduction. The second solution, Chawla et al. proposed a Synthetic Minority Over-Sampling Technique which combined method of oversampling the minority class and undersampling the majority class [74]. The third solution could be through changing the loss function via weights.

When multiplying the loss with the term in which a minority class occurs with a constant value greater than 1, the classifier is forced or encouraged to pay more attention to the minority class. The forth and also the last solution to resolve imbalanced data is another reweight technique. Cui et al. proposed a reweighting scheme that use the effective number of samples for each class to rebalance the loss which results in a class-balanced loss [75].

The fourth challenge of this work is acquiring a gold standard dataset. This work is based on the dataset curated and labeled by Sharma et al. yet the dataset labels were not labeled by human, rather, via factchecking sources [13]. Therefore, each Tweet with one or more misinformation label was solely dependent on the domain of the URLs shared in the Tweet, and content of the Tweet was not considered in identifying misinformation Tweets. Labeling a post on social media or news article based on domain of the news source is prevalent in recent research works [76][77][78][79]. Micallef et al. recognized that a very small percentage, 10%, of Tweets include external links and hence, developed a novel COVID-19 related dataset including 4,800 Tweets annotated by human such that each Tweet is labeled as one of the three categories: misinformation, counter-misinformation, and irrelevant [80]. However, Micallef et al. dataset was only concerned about COVID-19 misinformation on fake cures and 5G conspiracy theories. Clearly, a gold standard dataset that examines the content of a post is lacking. Furthermore, the standard for COVID-19 misinformation classification is a controversial topic since recommendations and assessments may change over time due to new and updated scientific evidence, according to the United States Surgeon General, Dr. Vivek Murthy

[4]. Using the example in this work, the origin of COVID-19 in a laboratory was considered a conspiracy in the beginning of the outbreak in 2020. However, in May of 2021, the U.S. president ordered intelligence community to investigate the origins of COVID-19 including the theory of the virus potentially coming from a Chinese laboratory [81]. What was once firmly held to be a pure conspiracy theory is now under investigation with inconclusive official conclusions made public [82]. This makes the subject to be exceedingly nebulous, and only serves to add to the level of uncertainty within the general public. Conspiracy theories as such are challenging to categorize into a binary of most likely true versus most likely untrue because of their shifting perception among both the public and the public officials investigating them. This generates an exponentially difficult scenario for scientific researchers to firmly and confidently debunk these statements accurately as time can shift perception of them from most likely untrue to controversial to possibly true as the investigation is ongoing. A solution to this challenge in categorizing the data is to increase transparency in the information made available to the research community and the general public as the president advocates for a “full and transparent accounting” of the pandemic [82].

## Bibliography

- [1] Talkwalke, “COVID-19 Global Report,” Accessed: Jan. 06, 2022. [Online]. Available: <https://www.talkwalker.com/resource/covid-19-dashboard-march-22nd.pdf>.
- [2] V. Suarez-Lledo and J. Alvarez-Galvez, “Prevalence of Health Misinformation on Social Media: Systematic Review,” *J Med Internet Res* 2021;23(1)e17187 <https://www.jmir.org/2021/1/e17187>, vol. 23, no. 1, p. e17187, Jan. 2021, doi: 10.2196/17187.
- [3] M. Gottlieb and S. Dyer, “Information and Disinformation: Social Media in the COVID-19 Crisis,” *Acad. Emerg. Med.*, vol. 27, no. 7, pp. 640–641, Jul. 2020, doi: 10.1111/ACEM.14036.
- [4] V. H. Murthy, “Confronting Health Misinformation: The U.S. Surgeon General’s Advisory on Building a Healthy Information Environment,” 2021, Accessed: Jan. 06, 2022. [Online]. Available: <https://www.hhs.gov/sites/default/files/surgeon-general-misinformation-advisory.pdf>.
- [5] World Health Organization, “Novel Coronavirus(2019-nCoV) Situation Report-13,” Accessed: Jan. 06, 2022. [Online]. Available: <https://www.who.int/docs/default-source/coronaviruse/situation-reports/20200202-sitrep-13-ncov-v3.pdf>.
- [6] E. K. Vraga and L. Bode, “Defining Misinformation and Understanding its Bounded Nature: Using Expertise and Evidence for Describing Misinformation,” <https://doi.org/10.1080/10584609.2020.1716500>, vol. 37, no. 1, pp. 136–144, Jan.

- 2020, doi: 10.1080/10584609.2020.1716500.
- [7] DHS, “Homeland Threat Assessment October 2020,” 2020. Accessed: Jan. 06, 2022. [Online]. Available: [https://www.dhs.gov/sites/default/files/publications/2020\\_10\\_06\\_homeland-threat-assessment.pdf](https://www.dhs.gov/sites/default/files/publications/2020_10_06_homeland-threat-assessment.pdf).
- [8] C. Maier, N. Tipton, and J. Sullivan, “DISINFORMATION IN THE GRAY ZONE: OPPORTUNITIES, LIMITATIONS, CHALLENGES,” *HOUSE ARMED SERVICES COMMITTEE SUBCOMMITTEE ON INTELLIGENCE AND SPECIAL OPERATIONS*, Mar. 16, 2021. <https://docs.house.gov/meetings/AS/AS26/20210316/111323/HHRG-117-AS26-Wstate-MaierC-20210316.pdf> (accessed Jan. 06, 2022).
- [9] Office of the Director of National Intelligence, “Annual Threat Assessment of the US Intelligence Community,” Apr. 09, 2021. <https://www.dni.gov/files/ODNI/documents/assessments/ATA-2021-Unclassified-Report.pdf> (accessed Jan. 06, 2022).
- [10] L. Bandeira, N. Aleksejeva, T. Knight, and J. Le Roux, “WEAPONIZED: HOW RUMORS ABOUT COVID-19’S ORIGINS LED TO A NARRATIVE ARMS RACE,” Washington, DC, Feb. 2021. Accessed: Jan. 06, 2022. [Online]. Available: <https://www.atlanticcouncil.org/wp-content/uploads/2021/02/Weaponized-How-rumors-about-COVID-19s-origins-led-to-a-narrative-arms-race.pdf>.
- [11] WHO *et al.*, “Managing the COVID-19 infodemic: Promoting healthy behaviours

- and mitigating the harm from misinformation and disinformation,” *WHO*, Sep. 23, 2020. <https://www.who.int/news/item/23-09-2020-managing-the-covid-19-infodemic-promoting-healthy-behaviours-and-mitigating-the-harm-from-misinformation-and-disinformation> (accessed Jan. 06, 2022).
- [12] T. K. Sell *et al.*, “National Priorities to Combat Misinformation and Disinformation for COVID-19 and Future Public Health Threats: A Call for a National Strategy,” Mar. 2021, Accessed: Jan. 11, 2022. [Online]. Available: [https://www.centerforhealthsecurity.org/our-work/pubs\\_archive/pubs-pdfs/2021/210322-misinformation.pdf](https://www.centerforhealthsecurity.org/our-work/pubs_archive/pubs-pdfs/2021/210322-misinformation.pdf).
- [13] K. Sharma, S. Seo, C. Meng, S. Rambhatla, and Y. Liu, “COVID-19 ON SOCIAL MEDIA: ANALYZING MISINFORMATION IN TWITTER CONVERSATIONS A PREPRINT,” 2020, Accessed: Sep. 01, 2021. [Online]. Available: <https://usc-melady.github.io/COVID-19-Tweet-Analysis>.
- [14] G. Y. Smith, C. M. Schubert Kabban, K. M. Hopkinson, M. E. Oxley, G. E. Noel, and H. Cheng, “Sensor Fusion for Context Analysis in Social Media COVID-19 Data,” in *National Aerospace & Electronics Conference*, Feb. 2022, pp. 415–422, doi: 10.1109/NAECON49338.2021.9696396.
- [15] J. F. Allen, “NATURAL LANGUAGE PROCESSING,” *Encycl. Comput. Sci.*, pp. 1218–1222, Jan. 2003, doi: 10.5555/1074100.
- [16] J. Hirschberg and C. D. Manning, “Advances in natural language processing,” *Science* (80-. ), vol. 349, no. 6245, pp. 261–266, Jul. 2015, doi: 10.1126/SCIENCE.AAA8685/ASSET/D33AB763-A443-444C-B766-



A6B69883BFD7/ASSETS/GRAPHIC/349\_261\_F5.JPEG.

- [17] B. Shetty, “Natural Language Processing(NLP) for Machine Learning | by Badreesh Shetty | Towards Data Science,” *Towards Data Science*, Nov. 24, 2018.  
<https://towardsdatascience.com/natural-language-processing-nlp-for-machine-learning-d44498845d5b> (accessed Feb. 23, 2022).
- [18] NLTK Project, “Natural Language Toolkit — NLTK 3.2.5 documentation.”  
<https://nltk.readthedocs.io/en/latest/index.html> (accessed Feb. 28, 2022).
- [19] “Porter Stemming Algorithm.” <https://tartarus.org/martin/PorterStemmer/>  
(accessed Sep. 01, 2021).
- [20] “nltk.stem package — NLTK 3.6.2 documentation.”  
<https://www.nltk.org/api/nltk.stem.html?highlight=wordnetlemmatizer#nltk.stem.wordnet.WordNetLemmatizer> (accessed Sep. 01, 2021).
- [21] J. Daniel and J. H. Martin, “Naive Bayes and Sentiment Classification,” in *Speech and Language Processing*, 2021.
- [22] J. Daniel and J. H. Martin, “Logistic Regression,” in *Speech and Language Processing*, 3rd ed., 2021.
- [23] D. Jurafsky and J. Martin, “Speech and Language Processing,” 2021.  
<https://web.stanford.edu/~jurafsky/slp3/> (accessed Feb. 22, 2022).
- [24] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning with Applications in R*. New York, NY: Springer, 2013.
- [25] “sklearn.naive\_bayes.MultinomialNB — scikit-learn 0.24.2 documentation.”  
<https://scikit->

[learn.org/stable/modules/generated/sklearn.naive\\_bayes.MultinomialNB.html](http://learn.org/stable/modules/generated/sklearn.naive_bayes.MultinomialNB.html)

(accessed Sep. 01, 2021).

- [26] F. Pedregosa *et al.*, “Scikit-learn: Machine Learning in Python Gaël Varoquaux Bertrand Thirion Vincent Dubourg Alexandre Passos PEDREGOSA, VAROQUAUX, GRAMFORT ET AL. Matthieu Perrot,” *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011, Accessed: Feb. 17, 2022. [Online]. Available: <http://scikit-learn.sourceforge.net>.
- [27] “sklearn.linear\_model.LogisticRegression — scikit-learn 0.24.2 documentation.” [https://scikit-learn.org/stable/modules/generated/sklearn.linear\\_model.LogisticRegression.html](https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html) (accessed Sep. 01, 2021).
- [28] “sklearn.svm.LinearSVC — scikit-learn 0.24.2 documentation.” <https://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVC.html> (accessed Sep. 01, 2021).
- [29] D. L. Hall and J. Llinas, “An introduction to multisensor data fusion,” *Proc. IEEE*, vol. 85, no. 1, pp. 6–23, 1997, doi: 10.1109/5.554205.
- [30] J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas, “On Combining Classifiers,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 3, pp. 226–239, Mar. 1998, doi: 10.1109/34.667881.
- [31] A. K. Jain, R. P. W. Duin, and J. Mao, “Statistical pattern recognition: A review,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 1, pp. 4–37, Jan. 2000, doi: 10.1109/34.824819.

- [32] C. Schubert, “Quantifying Correlation and Its Effects on System Performance in Classifier Fusion,” Air Force Institution of Technology, Wright Patterson Air Force Base, 2005.
- [33] G. Rogova, “Combining the results of several neural network classifiers,” *Neural Networks*, vol. 7, no. 5, pp. 777–781, Jan. 1994, doi: 10.1016/0893-6080(94)90099-X.
- [34] D. Angelov, “Top2Vec: Distributed Representations of Topics,” Aug. 19, 2020. <https://arxiv.org/abs/2008.09470v1> (accessed Feb. 03, 2022).
- [35] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, “Indexing by Latent Semantic Analysis,” *J. Assoc. Inf. Sci. Technol.*, vol. 41, pp. 391–407, 1990, Accessed: Feb. 03, 2022. [Online]. Available: <https://www.stat.cmu.edu/~cshalizi/350/2008/readings/Deerwester-et-al.pdf>.
- [36] T. Hofmann, “Probabilistic latent semantic indexing,” *Proc. 22nd Annu. Int. ACM SIGIR Conf. Res. Dev. Inf. Retrieval, SIGIR 1999*, pp. 50–57, Aug. 1999, doi: 10.1145/312624.312649.
- [37] D. M. Blei, A. Y. Ng, and M. I. Jordan, “Latent Dirichlet Allocation,” *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, 2003, Accessed: Feb. 03, 2022. [Online]. Available: <https://jmlr.org/papers/volume3/blei03a/blei03a.pdf>.
- [38] L. G. Serrano, “Latent Dirichlet Allocation (Part 1 of 2),” *YouTube*, Mar. 19, 2020. <https://www.youtube.com/watch?v=T05t-SqKArY> (accessed Feb. 26, 2022).
- [39] Q. Le and T. Mikolov, “Distributed Representations of Sentences and Documents,” *Proc. 31st Int. Conf. Mach. Learn.*, vol. 32, pp. 11–1188–11–1196,

- 2014, Accessed: Mar. 05, 2022. [Online]. Available:  
[https://cs.stanford.edu/~quocle/paragraph\\_vector.pdf](https://cs.stanford.edu/~quocle/paragraph_vector.pdf).
- [40] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Distributed Representations of Words and Phrases and their Compositionality,” *NIPS’13 Proc. 26th Int. Conf. Neural Inf. Process. Syst.*, vol. 2, pp. 3111–3119, Dec. 2013, Accessed: Mar. 05, 2022. [Online]. Available:  
<https://proceedings.neurips.cc/paper/2013/file/9aa42b31882ec039965f3c4923ce901b-Paper.pdf>.
- [41] L. McInnes, J. Healy, and J. Melville, “UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction,” Feb. 2018, doi:  
 10.48550/arxiv.1802.03426.
- [42] J. Laufs and Z. Waseem, “Policing in pandemics: A systematic review and best practices for police response to COVID-19,” *Int. J. Disaster Risk Reduct.*, vol. 51, p. 101812, Dec. 2020, doi: 10.1016/J.IJDRR.2020.101812.
- [43] D. J. Jones, “The Potential Impacts of Pandemic Policing on Police Legitimacy: Planning Past the COVID-19 Crisis,” *Polic. A J. Policy Pract.*, vol. 14, no. 3, pp. 579–586, Sep. 2020, doi: 10.1093/POLICE/PAAA026.
- [44] K. Papazoglou, D. M. Blumberg, M. D. Schlosser, and P. I. Collins, “Policing during COVID-19: Another day, another crisis,” *J. Community Saf. Well-Being*, vol. 5, no. 2, pp. 39–41, Jul. 2020, doi: 10.35502/JCSWB.130.
- [45] S. C. Guntuku *et al.*, “Tracking Mental Health and Symptom Mentions on Twitter During COVID-19,” *J. Gen. Intern. Med.*, vol. 35, no. 9, pp. 2798–2800, Sep.

2020, doi: 10.1007/S11606-020-05988-8.

- [46] W. Cullen, G. Gulati, and B. D. Kelly, “Mental health in the COVID-19 pandemic,” *QJM An Int. J. Med.*, vol. 113, no. 5, pp. 311–312, May 2020, doi: 10.1093/QJMED/HCAA110.
- [47] B. Pfefferbaum and C. S. North, “Mental Health and the Covid-19 Pandemic,” *N. Engl. J. Med.*, vol. 383, no. 6, pp. 510–512, Aug. 2020, doi: 10.1056/NEJMP2008017/SUPPL\_FILE/NEJMP2008017\_DISCLOSURES.PDF.
- [48] N. Micallef, B. He, S. Kumar, M. Ahamad, and N. Memon, “The Role of the Crowd in Countering Misinformation: A Case Study of the COVID-19 Infodemic,” in *Proceedings - 2020 IEEE International Conference on Big Data, Big Data 2020*, Dec. 2020, pp. 748–757, doi: 10.1109/BIGDATA50022.2020.9377956.
- [49] Z. Kou, L. Shang, Y. Zhang, C. Youn, and D. Wang, “FakeSens: A Social Sensing Approach to COVID-19 Misinformation Detection on Social Media,” in *2021 17th International Conference on Distributed Computing in Sensor Systems (DCOSS)*, Nov. 2021, pp. 140–147, doi: 10.1109/DCOSS52077.2021.00035.
- [50] M. A. Weinzierl and S. M. Harabagiu, “Automatic detection of COVID-19 vaccine misinformation with graph link prediction,” *J. Biomed. Inform.*, vol. 124, p. 103955, Dec. 2021, doi: 10.1016/J.JBI.2021.103955.
- [51] M. S. Al-Rakhami and A. M. Al-Amri, “Lies Kill, Facts Save: Detecting COVID-19 Misinformation in Twitter,” *IEEE Access*, vol. 8, pp. 155961–155970, 2020, doi: 10.1109/ACCESS.2020.3019600.

- [52] L. Singh, L. Bode, C. Budak, K. Kawintiranon, C. Padden, and E. Vraga, "Understanding high- and low-quality URL Sharing on COVID-19 Twitter streams," *J. Comput. Soc. Sci.*, vol. 3, no. 2, pp. 343–366, Nov. 2020, doi: 10.1007/S42001-020-00093-6/TABLES/6.
- [53] L. Prandi and G. Primiero, "Effects of misinformation diffusion during a pandemic," *Appl. Netw. Sci.*, vol. 5, no. 1, pp. 1–20, Dec. 2020, doi: 10.1007/S41109-020-00327-6/FIGURES/14.
- [54] M. E. Basiri, S. Nemati, M. Abdar, S. Asadi, and U. R. Acharrya, "A novel fusion-based deep learning model for sentiment analysis of COVID-19 tweets," *Knowledge-Based Syst.*, vol. 228, p. 107242, Sep. 2021, doi: 10.1016/J.KNOSYS.2021.107242.
- [55] D. Ashok Kumar and A. Chinnalagu, "Sentiment and emotion in social media covid-19 conversations: SAB-LSTM approach," in *Proceedings of the 2020 9th International Conference on System Modeling and Advancement in Research Trends, SMART 2020*, Dec. 2020, pp. 60–68, doi: 10.1109/SMART50582.2020.9337098.
- [56] Q. Liu *et al.*, "Health Communication Through News Media During the Early Stage of the COVID-19 Outbreak in China: Digital Topic Modeling Approach," *J Med Internet Res* 2020;22(4)e19118 <https://www.jmir.org/2020/4/e19118>, vol. 22, no. 4, p. e19118, Apr. 2020, doi: 10.2196/19118.
- [57] F. Kaveh-Yazdy and S. Zarifzadeh, "Track Iran's national COVID-19 response committee's major concerns using two-stage unsupervised topic modeling," *Int. J.*

*Med. Inform.*, vol. 145, p. 104309, Jan. 2021, doi:

10.1016/J.IJMEDINF.2020.104309.

- [58] A. Go, R. Bhayani, and L. Huang, "Twitter Sentiment Classification using Distant Supervision," Accessed: Jan. 26, 2022. [Online]. Available: <https://www-cs.stanford.edu/people/alecmgo/papers/TwitterDistantSupervision09.pdf>.
- [59] World Health Organization, "Novel Coronavirus(2019-nCoV) Situation Report-13." Accessed: Sep. 06, 2021. [Online]. Available: <https://www.who.int/docs/default-source/coronaviruse/situation-reports/20200202-sitrep-13-ncov-v3.pdf>.
- [60] J. Posetti and K. Bontcheva, "Deciphering COVID-19 disinformation Policy brief 1."
- [61] T. Hossain, R. L. Logan IV, A. Ugarte, Y. Matsubara, S. Young, and S. Singh, "COVIDLies: Detecting COVID-19 Misinformation on Social Media," 2020, doi: 10.18653/v1/2020.nlpccovid19-2.11.
- [62] S. Boon-Itt and Y. Skunkan, "Public perception of the COVID-19 pandemic on twitter: Sentiment analysis and topic modeling study," *JMIR Public Heal. Surveill.*, vol. 6, no. 4, p. e21978, Oct. 2020, doi: 10.2196/21978.
- [63] Hyeju, E. Rempel, D. Roth, G. Carenini, and N. Z. Janjua, "Tracking COVID-19 Discourse on Twitter in North America: Infodemiology Study Using Topic Modeling and Aspect-Based Sentiment Analysis," *J Med Internet Res* 2021;23(2)e25431 <https://www.jmir.org/2021/2/e25431>, vol. 23, no. 2, p. e25431, Feb. 2021, doi: 10.2196/25431.

- [64] F. Martinez-Plumed *et al.*, “CRISP-DM Twenty Years Later: From Data Mining Processes to Data Science Trajectories,” *IEEE Trans. Knowl. Data Eng.*, pp. 1–1, Dec. 2019, doi: 10.1109/tkde.2019.2962680.
- [65] “Sentiment140 dataset with 1.6 million tweets | Kaggle.”  
<https://www.kaggle.com/kazanova/sentiment140> (accessed Sep. 01, 2021).
- [66] “US COVID Tweets | Kaggle.” <https://www.kaggle.com/yazanshannak/us-covid-tweets?select=US+COVID-19+Tweets.csv> (accessed Sep. 01, 2021).
- [67] “COVID19 Tweets | Kaggle.” <https://www.kaggle.com/gpreda/covid19-tweets> (accessed Sep. 01, 2021).
- [68] “Media Bias/Fact Check - Search and Learn the Bias of News Media.”  
<https://mediabiasfactcheck.com/> (accessed Sep. 01, 2021).
- [69] “COVID-19 Misinformation Resources - NewsGuard.”  
<https://www.newsguardtech.com/covid-19-resources/> (accessed Sep. 01, 2021).
- [70] M. Zimdars, “False, Misleading, Clickbait-y, and/or Satirical ‘News’ Sources.”  
<https://21stcenturywire.com/wp-content/uploads/2017/02/2017-DR-ZIMDARS-False-Misleading-Clickbait-y-and-Satirical-“News”-Sources-Google-Docs.pdf>  
 (accessed Sep. 01, 2021).
- [71] “sklearn.feature\_extraction.text.CountVectorizer — scikit-learn 0.24.2 documentation.” [https://scikit-learn.org/stable/modules/generated/sklearn.feature\\_extraction.text.CountVectorizer.html](https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html) (accessed Sep. 01, 2021).
- [72] “sklearn.feature\_extraction.text.TfidfVectorizer — scikit-learn 0.24.2



- documentation.” [https://scikit-learn.org/stable/modules/generated/sklearn.feature\\_extraction.text.TfidfVectorizer.html](https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html) (accessed Sep. 01, 2021).
- [73] J. Roozenbeek *et al.*, “Susceptibility to misinformation about COVID-19 around the world,” *R. Soc. Open Sci.*, vol. 7, no. 10, Oct. 2020, doi: 10.1098/RSOS.201199.
- [74] N. V Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “SMOTE: Synthetic Minority Over-sampling Technique,” *J. Artif. Intell. Res.*, vol. 16, pp. 321–357, Jun. 2002, Accessed: Mar. 06, 2022. [Online]. Available: <https://www3.nd.edu/~dial/publications/chawla2002smote.pdf>.
- [75] Y. Cui, M. Jia, T. Y. Lin, Y. Song, and S. Belongie, “Class-balanced loss based on effective number of samples,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Jun. 2019, vol. 2019-June, pp. 9260–9269, doi: 10.1109/CVPR.2019.00949.
- [76] A. Guess, B. Nyhan, and J. Reifler, “Selective exposure to misinformation: Evidence from the consumption of fake news during the 2016 US presidential campaign,” 2018. Accessed: Feb. 18, 2022. [Online]. Available: <http://www.askeforce.org/web/Fundamentalists/Guess-Selective-Exposure-to-Misinformation-Evidence-Presidential-Campaign-2018.pdf>.
- [77] C. Shao, G. L. Ciampaglia, O. Varol, K. C. Yang, A. Flammini, and F. Menczer, “The spread of low-credibility content by social bots,” *Nat. Commun.* 2018 91, vol. 9, no. 1, pp. 1–9, Nov. 2018, doi: 10.1038/s41467-018-06930-7.

- [78] L. Bozarth and C. Budak, "Toward a Better Performance Evaluation Framework for Fake News Classification," in *Proceedings of the International AAAI Conference on Web and Social Media*, May 2020, vol. 14, pp. 60–71, Accessed: Feb. 18, 2022. [Online]. Available: <https://ojs.aaai.org/index.php/ICWSM/article/view/7279>.
- [79] L. Bozarth and C. Budak, "Market Forces: Quantifying the Role of Top Credible Ad Servers in the Fake News Ecosystem," in *Proceedings of the International AAAI Conference on Web and Social Media*, May 2021, vol. 15, pp. 83–94, Accessed: Feb. 18, 2022. [Online]. Available: <https://ojs.aaai.org/index.php/ICWSM/article/view/18043>.
- [80] N. Micallef, B. He, S. Kumar, M. Ahamad, and N. Memon, "The Role of the Crowd in Countering Misinformation: A Case Study of the COVID-19 Infodemic," in *2020 IEEE International Conference on Big Data (Big Data)*, Dec. 2020, pp. 748–757, doi: 10.1109/BIGDATA50022.2020.9377956.
- [81] "Covid: Biden orders investigation into virus origin as lab leak theory debated - BBC News." <https://www.bbc.com/news/world-us-canada-57260009> (accessed Feb. 18, 2022).
- [82] "Statement by President Joe Biden on the Investigation into the Origins of COVID-19 | The White House." <https://www.whitehouse.gov/briefing-room/statements-releases/2021/08/27/statement-by-president-joe-biden-on-the-investigation-into-the-origins-of-covid-19/> (accessed Feb. 19, 2022).

<b>REPORT DOCUMENTATION PAGE</b>					<i>Form Approved OMB No. 0704-0188</i>	
<small>The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.</small>						
<b>PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.</b>						
<b>1. REPORT DATE (DD-MM-YYYY)</b>		<b>2. REPORT TYPE</b>			<b>3. DATES COVERED (From - To)</b>	
<b>4. TITLE AND SUBTITLE</b>				<b>5a. CONTRACT NUMBER</b>		
				<b>5b. GRANT NUMBER</b>		
				<b>5c. PROGRAM ELEMENT NUMBER</b>		
<b>6. AUTHOR(S)</b>				<b>5d. PROJECT NUMBER</b>		
				<b>5e. TASK NUMBER</b>		
				<b>5f. WORK UNIT NUMBER</b>		
<b>7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)</b>					<b>8. PERFORMING ORGANIZATION REPORT NUMBER</b>	
<b>9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)</b>					<b>10. SPONSOR/MONITOR'S ACRONYM(S)</b>	
					<b>11. SPONSOR/MONITOR'S REPORT NUMBER(S)</b>	
<b>12. DISTRIBUTION/AVAILABILITY STATEMENT</b>						
<b>13. SUPPLEMENTARY NOTES</b>						
<b>14. ABSTRACT</b>						
<b>15. SUBJECT TERMS</b>						
<b>16. SECURITY CLASSIFICATION OF:</b>			<b>17. LIMITATION OF ABSTRACT</b>	<b>18. NUMBER OF PAGES</b>	<b>19a. NAME OF RESPONSIBLE PERSON</b>	
a. REPORT	b. ABSTRACT	c. THIS PAGE			<b>19b. TELEPHONE NUMBER (Include area code)</b>	

## INSTRUCTIONS FOR COMPLETING SF 298

**1. REPORT DATE.** Full publication date, including day, month, if available. Must cite at least the year and be Year 2000 compliant, e.g. 30-06-1998; xx-06-1998; xx-xx-1998.

**2. REPORT TYPE.** State the type of report, such as final, technical, interim, memorandum, master's thesis, progress, quarterly, research, special, group study, etc.

**3. DATES COVERED.** Indicate the time during which the work was performed and the report was written, e.g., Jun 1997 - Jun 1998; 1-10 Jun 1996; May - Nov 1998; Nov 1998.

**4. TITLE.** Enter title and subtitle with volume number and part number, if applicable. On classified documents, enter the title classification in parentheses.

**5a. CONTRACT NUMBER.** Enter all contract numbers as they appear in the report, e.g. F33615-86-C-5169.

**5b. GRANT NUMBER.** Enter all grant numbers as they appear in the report, e.g. AFOSR-82-1234.

**5c. PROGRAM ELEMENT NUMBER.** Enter all program element numbers as they appear in the report, e.g. 61101A.

**5d. PROJECT NUMBER.** Enter all project numbers as they appear in the report, e.g. 1F665702D1257; ILIR.

**5e. TASK NUMBER.** Enter all task numbers as they appear in the report, e.g. 05; RF0330201; T4112.

**5f. WORK UNIT NUMBER.** Enter all work unit numbers as they appear in the report, e.g. 001; AFAPL30480105.

**6. AUTHOR(S).** Enter name(s) of person(s) responsible for writing the report, performing the research, or credited with the content of the report. The form of entry is the last name, first name, middle initial, and additional qualifiers separated by commas, e.g. Smith, Richard, J, Jr.

**7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES).** Self-explanatory.

**8. PERFORMING ORGANIZATION REPORT NUMBER.** Enter all unique alphanumeric report numbers assigned by the performing organization, e.g. BRL-1234; AFWL-TR-85-4017-Vol-21-PT-2.

**9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES).** Enter the name and address of the organization(s) financially responsible for and monitoring the work.

**10. SPONSOR/MONITOR'S ACRONYM(S).** Enter, if available, e.g. BRL, ARDEC, NADC.

**11. SPONSOR/MONITOR'S REPORT NUMBER(S).** Enter report number as assigned by the sponsoring/monitoring agency, if available, e.g. BRL-TR-829; -215.

**12. DISTRIBUTION/AVAILABILITY STATEMENT.** Use agency-mandated availability statements to indicate the public availability or distribution limitations of the report. If additional limitations/ restrictions or special markings are indicated, follow agency authorization procedures, e.g. RD/FRD, PROPIN, ITAR, etc. Include copyright information.

**13. SUPPLEMENTARY NOTES.** Enter information not included elsewhere such as: prepared in cooperation with; translation of; report supersedes; old edition number, etc.

**14. ABSTRACT.** A brief (approximately 200 words) factual summary of the most significant information.

**15. SUBJECT TERMS.** Key words or phrases identifying major concepts in the report.

**16. SECURITY CLASSIFICATION.** Enter security classification in accordance with security classification regulations, e.g. U, C, S, etc. If this form contains classified information, stamp classification level on the top and bottom of this page.

**17. LIMITATION OF ABSTRACT.** This block must be completed to assign a distribution limitation to the abstract. Enter UU (Unclassified Unlimited) or SAR (Same as Report). An entry in this block is necessary if the abstract is to be limited.