Towards Checking Veracity of Medical Claims

by

Abhishek Dhankar

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

Department of Computing Science
University of Alberta

Abstract

Medical Fake News is a pervasive part of the information that people consume on the internet. It may lead people to take actions which may put the lives of their family and community in danger - such actions include vaccine hesitancy, administering unverified and harmful treatments, etc. First step towards countering such fake news is to detect it. In this report we explore various approaches to automatically detect and determine the veracity of textual claims, especially but not limited to medical claims, found online in social media posts and articles. In this report we present (1) An automated veracity checker for online articles pertaining to NeuroDevelopmental Disorders (NDDs) (2) Our work on detection of fake news in social media posts related to COVID-19 (3) Our approach to the shared task of Multi-Modal Fake News detection at the De-FACTIFY Workshop collocated with AAAI'22, where we secured the 4th position on the leaderboard.

Preface

The research presented in this thesis forms a part of a research collaboration led by Prof. Osmar R. Zaïane and Dr. Francois Bolduc. In Chapter 3, I was responsible for data collection, writing code, conceptualizing experiments, and analyzing the results. A. Bui, J. Costello and M. Kaur helped in collection of data. Dr. F. Bolduc helped in recruiting experts and Neuroscience students to annotate data. Prof. O. Zaïane helped in conceptualization of the experiments, overall focus of research and helped in writing manuscript. Some parts of this thesis are a part of published works with collaborators. Chapter 4 of this thesis has been published as A. Dhankar, H. Samuel, F. Hassan, N. Farruque, F. Bolduc, & O.R. Zaïane, "Analysis of COVID-19 Misinformation in Social Media using Transfer Learning," *2021 IEEE 33rd International Conference on Tools with Artificial Intelligence (ICTAI)* (pp. 880-885), IEEE. I was responsible for writing the code, running the experiments, analyzing the results, and writing the manuscript. H. Samuel led the research group and contributed to writing the manuscript. F.Hassan helped in providing a healthcare context in the manuscript. N. Farruque helped in formulating experiments since they made use of his previous work. Prof. O.R. Zaïane helped in conceptualization of experiments, writing and editing the manuscript, and overall focus of the paper. Chapter 5 of this thesis has been published as A. Dhankar, O.R. Zaïane, & F. Bolduc, "UofA-Truth at Factify 2022: A simple approach to multi-modal fact-checking," *Proc. of De-Factify Workshop on Multimodal Fact Checking and Hate Speech Detection, co-located with AAAI 2022*. I was responsible for carrying out experiments, analyzing results and writing the manuscript. O. Zaïanea and F. Bolduc helped in writing the manuscript.

Acknowledgements

First and foremost, I would like to thank my thesis advisors Professor Osmar Zaiane and Dr. Francois Bolduc. This thesis would not have been possible without their unyielding support.

Professor Zaiane's support through the pandemic was essential in the completion of this these. He was extremely generous with his time and always available to provide advise on the Machine Learning and Natural Language Processing aspects of this thesis. His immense reservoir of patience and optimism buoyed my spirits during some of the worst times of my research. Members of his research group immensely helped in completion of significant portions of this thesis. Dr. Hamman Samuel provided code and clarifications on his implementation of MedFact, which formed the basis for this thesis. He also provided leadership in the paper on COVID-19 misinformation detection in Social Media posts, which formed a part of this thesis in Chapter 4. Mr. Nawshad Farruque provided valuable advise and helped in validating my implementation of his paper used in this thesis. Mr. Fahim Hassan helped in providing a healthcare policy perspective on misinformation detection. All the aforementioned research group members helped in editing and adding to our paper on COVID-19 misinformation detection.

Dr. Bolduc provided guidance for the medical aspects of the project. He assembled the group of experts and neuroscience students who participated in the annotation of the dataset. I would like to deeply thank members of Dr. Bolduc's research group - Ms. An Bui for helping of the collection of the NDD dataset, Mr. Cory Rosenfelt for providing advise on medical aspects and moral support, Ms. Kerri Whitlock for

handling complicated the ethics approval process and communication with dataset annotators, James Benoit for providing guidance on medical aspects of the project, Ms. Manpreet Kaur for providing NDD related keywords and collecting the YouTube Dataset, Mr. Ashwani Singla for providing feedback on model performance.

I would also like to thank Mr. Jeremy Costello for aiding in the collection of YouTube Dataset.

Last, but not the least, I would like to thank my parents Anurag and Kalpana Dhankar, who stood by me through thick & thin, and have provided me their unconditional love and support throughout my life.

This work was partially funded by a Collaborative Health Research Project grant from the Canadian Institutes of Health Research (CIHR) and the Natural Sciences and Engineering Research Council of Canada (NSERC). Osmar Zaiane, a Canada CIFAR AI Chair, is also funded by the Canadian Institute for Advanced Research (CIFAR).

Contents

List of Tables

List of Figures

Abbreviations

ADHD Attention-Deficit/Hyperactivity Disorder.

ASD Autism Spectrum Disorder.

ATE-FC Augmented Twitter Embeddings for False Claims.

ATL Augmentation Transfer Learning.

CDC Centers for Disease Control and Prevention.

CP Cerebral Palsy.

CSE Context-Specific Embedding.

CT-BERT COVID-Twitter-BERT.

CTL Concatenation Transfer Learning.

CV Cross Validation.

FCE False Claim Embeddings.

FNC-1 Fake News Challenge - 1.

GTE General Twitter Embedding.

HBOT Hyperbaric Oxygen Therapy.

IMHO In My Honest Opinion.

LSTM Long Short Term Memory.

MLP Multi-Layer Perceptron.

MMR Measles, Mumps & Rubella.

NCBI National Center for Biotechnology Information.

NDD Neurodevelopmental Disorder.

NER Named Entity Recognition.

NLI Natural Language Inference.

NLM National Library of Medicine.

NYT New York Times.

PCA Principal Component Analysis.

RNN Recurrent Neural Network.

RST Rhetorical Structure Theory.

RTE Recognizing Textual Entailment.

SAF Social Article Fusion.

SVD Singular Value Decomposition.

SVM Support Vector Machine.

TF Term Frequency.

TF-IDF Term Frequency - Inverse Document Frequency.

YAKE Yet Another Keyword Extractor.

Chapter 1

Introduction

Fake news, in the general vernacular, has become a very nebulous term, including everything from verifiable false claims, to simply contrary political opinions. However, for the sake of scientific rigour, and for creating an automated approach to identify and eliminate "fake news", the term and its subcategories need to be defined. A definition will help in clarifying the subset of "fake news" that an automated approach can identify. For instance, the type of "fake news" identified could depend on the availability of a database of ground truth statements. In cases where such a database is available a statement with unknown veracity can be compared against one or more of the statements in the ground truth database to determine whether it is "fake" or not. In such a case "fake news" is any claim that does not agree with a predetermined set of true claims. In the absence of such a database, automated approaches may rely on linguistic cues prevalent among "fake news" sources to identify the same.

Firstly, we define "news" as a piece of text of arbitrary length. "Fake news" may be divided into the following categories:

- News Satire: Ridicule or criticize through the use of exaggeration. The news piece itself is true, however, the presentation is exaggerated. For example The Daily News.

- News Parody: Humorous news, which is completely, or partially fabricated, however, the audience has complete knowledge of the fabrication. For example,

the Onion.

- News Fabrication: News which contains verifiably false claims because the underlying facts have been intentionally fabricated.

- News Mistakes: News where a few claims are verifiably false due to mistakes which were committed while researching for the news. These mistakes could include sourcing the information from non-peer reviewed or even peer reviewed studies which later turn out to be false, or studies published in non-reputable or predatory journals.

Among the categories listed above, News Mistakes, Parodies, and Satire are not created to intentionally mislead. However, News Fabrication is created to intentionally mislead. All the categories except for satire contain false claims.

1.1 Motivation

1.1.1 Neurodevelopmental Disorder (NDD)

NDD afflicted children are typically disadvantaged when it comes to schooling, admissions to colleges, and employment opportunities. Since it is only possible to attenuate the negative affects of NDD in most cases, and not outright cure the disorders, medical professionals have come up with a number of interventions to improve the life outcomes of such individuals.

According to statistics 5% of Canadian children, aged 5 to 14 years, have disabilities, and 74% of these have Neurodevelopmental Disorder(NDD). This population is set to grow in the future. NDD is a collection of developmental disorders including Autism Spectrum Disorder (ASD), Cerebral Palsy (CP), Attention-Deficit/Hyperactivity Disorder (ADHD), etc.

1.1.2 Motivation for CAMI Chatbot

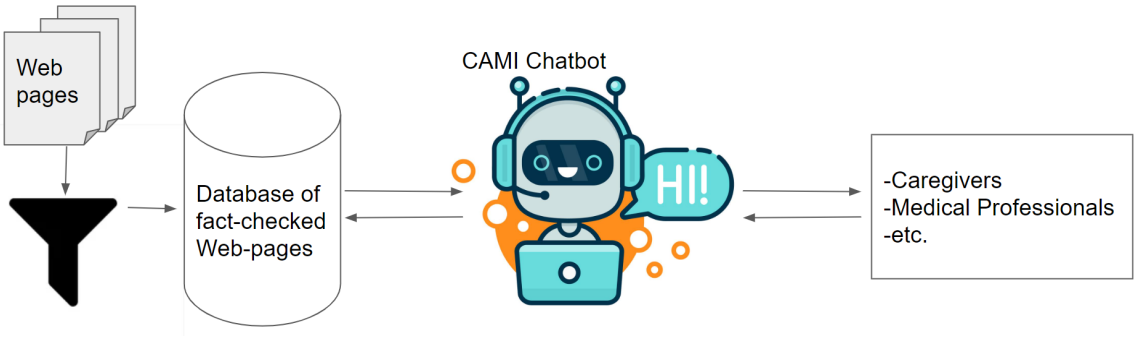


Figure 1.1: CAMI Chatbot & the Need for an Article Filter

A very small percentage, 5%, of children are diagnosed with NDD, and parents of children with a recent NDD diagnosis usually do not have more than a surface level knowledge of the disorders, their symptoms, causes, and treatments. Even fewer parents know of the multitude of resources including specialized education, different types of interventions available, income support, tax credits, etc., provided by federal or provincial governments. Interventions may not always work quickly to alleviate the symptoms of NDD, which is the best that can be done at the moment for most NDDs. This forces parents to search for interventions that promise to provide quicker or immediate relief. Usually, these types of interventions either do not have any evidence supporting their efficacy, or have inconclusive evidence, or have evidence showing that the intervention harms the patients. A clinician may not always have the information at hand or the time to advise parents on interventions or direct them to resources to aid them. In order to help such parents, clinicians, and other medical professionals find reliable information about all the aforementioned topics, a chatbot, called CAMI, shown in Figure 1.1, is being built to converse with users and automatically suggest resources in the form of webpages depending on the specific requirements of each user.

1.1.3 Motivation for Automated Veracity Checker

As shown in Figure 1.1 CAMI needs a database of webpages which is large enough to answer any query by users, but also should not contain false claims, i.e., the webpages need to be fact-checked before being included in the database. Currently, the webpages are added to the database after a manual fact-check by medical experts, but as the size of the database grows, it will be difficult to manually verify the veracity of information provided on each webpage. Furthermore, even webpages already admitted into the database need to be fact-checked regularly since information on these webpages can become outdated with respect to the latest medical literature or they may be modified by their authors to include inaccurate information. An automated approach is specially required in the case of NDDs, since there is a lot of misinformation on the internet related to these disorders. Some of the fake news related to disorders like ASD also contribute to vaccine hesitancy, which given the prevailing pandemic could prove dangerous for children with such a disorder if their parents are exposed to anti-vaccine information. This is where our contributions in fact-checking articles can help in creating a filter (shown as the filter sign in Figure 1.1) which ensures that only webpages containing accurate information are shared with the users of CAMI Chatbot.

1.2 Thesis Objectives

The overall objective of the thesis is to develop applications which can help in determining the veracity of only the medical claims made within webpages. These applications are to help filter out webpages containing any false claims in them.

The categories of "fake news" included the aspect of intention in their definitions. The objective of this thesis can be divided into two, one approach takes intention into consideration and the other does not. In the following objectives, "webpage" means the piece of text whose veracity is to be determined.

- Develop models which can identify and fact-check NDD related claims made in webpages .
- Explore the effect of different combinations of embeddings on model performance for detecting COVID-19 misinformation in social media posts.
- We demonstrate the effectiveness of using multimodal sources of information to determine the veracity of social media posts.

1.3 Thesis Outline

In Chapter 2 outlines the related literature. We outline the definition of fake news in greater detail, and discuss previous approaches to detecting misinformation, specifically in the domains of Politics and Medicine.

In Chapter 3 details the methodology of building a pipeline to fact-check NDD related articles. This tool for compares claims in unverified webpages to the claims made in relevant medical literature. The veracity of the individual claims will depend on whether they agree with the ground truth, i.e., the corresponding claims made in relevant medical literature. In addition, we also introduce a new NDD focused dataset which is used to test the aforementioned fact-checking tool.

In Chapter 4 we detail the effects of different combinations of embeddings on model performance for detecting COVID-19 misinformation, including a new method of creating context specific word embedding outlined in [1], where it has been successfully applied to the task of identifying tweets exhibiting depression. We test these embeddings on the CONSTRAINT_2021 dataset [2]. We demonstrate that the concatenation of general and context specific misinformation improves model performance over using the constituent embeddings individually.

In Chapter 5 we detail the results of our participation in the FACTIFY shared task at De-Factify@AAAI2022. The shared task challenged its participants to come up with the best model to detect misinformation using multi-modal sources of infor-

mation, i.e., images and text. Our submission to this task produced an F1-weighted score of 74.807%, which was the fourth best out of all the submissions.

Chapter 6 presents the conclusions of the findings of this dissertation and details the future works that may be undertaken on the basis of our work thus far.

Chapter 2

Related Works

The COVID-19 pandemic has revealed that misinformation spreads very quickly through social media. However, the infodemic associated with the pandemic merely revealed what had already been discovered by previous research about the spread of fake news. In [3], the authors discovered that fake news spreads faster than true news. They first identified news articles fact-checked by multiple fact-checking organizations, and classified them as true, false and mixed (partially true). They then tracked the spread of these articles across Twitter by looking at the tweets and re-tweets containing the links to aforementioned articles and formulating a tree structure. The root of such a tree was the first independent tweet containing links to aforementioned articles and its immediate children were the retweets of the root, and so on for rest of the roots. The maximum depth, the breadth of the tree at each level and the total number of nodes in the tree, all represent ways of measuring spread of news through Twitter. The authors found that fake news trees are deeper (greater max depth) and greater size (more total number of unique users reached) than true news. The authors further classified each article as relating to different topics such as politics, health, fiction, etc. The aforementioned conclusions hold true for all the topics, but were most pronounced for politics.

Waszak *et al.* in [4] did a similar study exclusively focusing on fake news related to health. They used the Buzzsumo Application to collect ten articles each for eight

keywords, namely cancer, neoplasm, heart attack, stroke, hypertension, diabetes, vaccinations, HIV and AIDS. The ten articles were selected after being ordered according to total engagement across multiple social media platforms. The selected articles were then classified into one of the following classes: fabricated news, manipulated news, advertisement news, irrelevant news, sufficient news. An article is defined as "fake news" if it is classified as any of the the following: fabricated, manipulated, advertisement news. The study found that 40% of the articles were classified as fake, and these were shared a total of 425,000 times across social media platforms such as Twitter, facebook, etc.

Given the speed with which fake news spreads and the number of people it misleads, it is important to counteract misinformation by either detecting and removing it early or counteracting it with accurate information which debunks said misinformation. The latter is also called Fact-Checking.

2.1 Manual Approach to Fact-Checking

Manual approaches generally consist of a Human annotator manually going through news articles, social media posts, etc. individually. Generally, fact-checks are carried out after the misinformation has already spread substantially, thus blunting any positive effects of the fact-check. An experienced journalist may suffice as an annotator for general political news, but fact-checking novel misinformation in specialized fields like medicine requires highly trained professionals like doctors to act as part-time annotators. Thus, fact-checking in these specialized fields may take longer due to the extreme time constraints that specialized professionals operate under.

2.2 Automated Approaches to Fact-Checking

Given that the manual fact-checking process inherently has low throughput, automatic approaches have been developed which either complement the work done by

manual fact-checkers or seek to completely replace them.

The work by Shu *et al.* in [5] falls under the category of complementing manual fact-checking. In an nutshell, they try to find fake news by matching unverified claims to manually fact-checked news articles. This study uses datasets which have been collected in previous studies to create and test an end to end fake news detection tool called FakeNewsTracker. They collected fact-checks from Politifact and Buzzfeed to build their dataset. Furthermore, they collected tweets and re-tweets related to the articles mentioned in the aforementioned fact-checks. The article content and social media engagement is jointly used to determine the veracity of said article.

The authors consider two types of information -

- Social Context **OR** Twitter User Information & Engagement: $\mathbf{x}_{ui}$ represents the user (u_i) information component which encodes the types of news the user engages with on Twitter. Here, a twitter user is said to "engage" with a news or article if they tweeted or retweeted a tweet with a link to the article, or they replied to a tweet or retweet linking to said article. The users' engagement with articles on twitter can thus be represented by $E \in \mathbb{R}^{m \times n}$, where m is the number of twitter users in the dataset and n is the number of articles, and each entry in E, i.e., E_{ij}, has a binary value representing whether the Twitter user u_i ever engaged with news piece j, or not. Every row in E is now a vector representation of the types of articles that a user engages with on twitter. However, in our case, the matrix E may contain thousands of columns (one column for each article in the dataset), and might be very sparse because each user may engage with a very small subset of all articles in the dataset. Thus, Singular Value Decomposition (SVD) [6] is used to reduce the dimensionality of E_i so that user engagement can be represented using a smaller and less sparse vector. Using SVD, the matrix E can be decomposed into matrices U, Σ and V as follows:

$$E = U\Sigma V \tag{2.1}$$

, where $U \in \mathbb{R}^{m \times k}$, $\Sigma \in \mathbb{R}^{k \times k}$ and $U \in \mathbb{R}^{k \times n}$. The i^{th} row of U, i.e., U_i, is of size $k(< n)$, and is a vector representation of the kind of articles or news that user i engages with on twitter - this is also called User Information of user i. Furthermore, the content of a user's tweets and retweets about a news piece is called Engagement Information and it contains information about the user's views about said news piece. This information can be encoded into a vector by using Doc2Vec[7]. The concatenation of U_i or User Information and the Engagement Embedding represents the Social Context corresponding to a single user's tweet, retweet or reply related to an article. Since these tweets, retweets or replies contain timestamps, there is a temporal aspect to the engagement of twitter users with the Article. When arranged by their timestamp, the set of vectors formed by the concatenation of Engagement Information and User Information, represent the Social Context of the article. If there are s tweets, retweets and replies related to an article, then its Social Context is represented by $X = x_1, x_2, x_3, ..., x_s$, where x_i is the concatenation of the User Information and Engagement Information of the tweets, retweets and replies, arranged such that the timestamp of $x_i < x_j$, when $i < j$.

- Article Content: Simply put, this is the vector representation of the article in question. Let $A = a_1, a_2, a_3, ..., a_m$ represent the article containing m words, each represented by an embedding vector a_i. The vector set A represents the Article Content.

Both Social Context and Article Content are fed to different neural networks during training.

Let A^j be an article such that $A^j = {a_1}^j, {a_2}^j, {a_3}^j, ..., {a_m}^j$, where ${a_i}^j$ is the embedding of the i^th word in the article. These word embeddings are sequentially fed to an auto-encoder. The encoder part of this auto-encoder is composed of LSTM layers, which takes Article A^j as input and compresses said vector into vector representation ${v_j}^1$,

in vector space Z. The decoder then takes the compressed vector representation of A^j and tries to auto-regressively regenerate the article A^j. The vector ${v_j}^1$ is the final output of the auto-encoder, which is further used in classification.

The Social Context vectors ($X = x_1, x_2, x_3, ..., x_s$) for Article A^j are sequentially put through an RNN with an LSTM unit. The order is determined by the time stamp on the tweet. The final output of the RNN is vector ${v_j}^2$, which represents the Social Context.

Both vectors ${v_j}^1$ (Article Content) and ${v_j}^2$ (Social Context) are concatenated and then fed through a shallow neural network to make the final prediction on the veracity of the Article. The entire network, including the Article Contentn (auto-encoder) and Social Context (LSTM layers), and shallow neural networks are trained together.

Some automated approaches to fake news detection have tried to mimic the process in which human fact-checkers operate. For instance, Atanasova *et al.* in [8] focused on detecting check-worthy sentences/claims before predicting the sentences' veracity. To learn how human fact-checkers determine check-worthiness of sentences, they used CW-USPD-2016 dataset introduced by [9] for political fact-checking. The dataset consists of a set of sentences which are annotated as check-worthy or not check-worthy. The sentences belong to 4 political debates which happened in the United States. Since it was a moderated debate, the participants, including the moderator, spoke in turns. Atanasova *et al.* termed each of these turns, a segment. A lot of reputable news organizations fact-checked these debates sentence by sentence. Of these, fact-checks by 9 reputable fact-checking organizations for each of the 4 debates were collected. Obviously, all the organizations did not fact-check the same set of sentences. Therefore, a sentence was annotated to be check-worthy if at least one of the organizations had fact-checked the whole or any segment of said sentence.

Atanasova *et al.* in [8] use upwards of a $1,000$ features to build their classifier. They selected features to reflect the content of the sentence in considerations as well as the context in which the sentence was uttered. Some of their different classes of

features were as follows:

- Position (3 features): The position of the target sentence. Two binary features, one each for whether it was the first sentence or the last sentence in the segment. The third feature is the reciprocal rank of the sentence in the segment.

- Embeddings (303 features): [8] used 300 vector length pre-trainend embeddings introduced by Mikolov *et al.* in [10] to encode sentences by taking the average of the embedding vectors of the constituent words. They also modelled the context by calculating the cosine measure of the vector representation of the target sentence with those of the previous, current and the following segment.

Other features included whether one or more Named Entities were present in the target sentence, whether the target sentence contained words which signalled negation or disagreement, such as, "didn't", "can't", etc.

The vector representation for each sentence was calculated using the aforementioned features. These vectors were used to train a fully connected neural network of two hidden layers.

Atanasova *et al.* found in their experiments that context modelling, as in the case of cosine measurements of the embedding features, along with features from the content of the target sentence were both key in producing state-of-the-art results.

Dai *et al.* in [11] present a dataset for misinformation detection in the medical domain. They scraped the HealthNewsReview.org, a website which determines the quality of news pieces related to medical news about latest products, research advances, etc.

The news pieces are broadly divided into two categories, namely, Health Story and Health Release. Health Story contains all the news pieces which have been released by media organizations like the health sections of NYT and Reuters about recent developments in the health and pharma world. Health Release contains all the

announcements of the latest research and studies coming out of universities, R&D of companies, etc.

Each news piece is reviewed by at least two human annotators. These annotators had expertise in relevant fields of journalism, medicine, health services research, etc. They rated each article on a score of 1 to 10 on each of the 10 criteria predetermined by the website. 8 of these criteria are common for news pieces contained in Health Release and Health Story categories, while the remaining 2 criteria are different for the 2 categories. Thereafter, the scores are averaged across the 10 criteria for each article, and the resulting score is scaled to a rage of 5. The score thus obtained is the final rating of the news piece. The criteria on which news pieces are judged are related to a range of issues, some of them are detailed below:

- Does it compare the new approach with existing alternatives?
- Does it seem to grasp the quality of the evidence?
- Does it commit disease-mongering?

In addition to the news pieces, the authors Dai *et al.*, also collect tweets linking said pieces, and the user profiles of the users engaging in discussion about the news pieces. Furthermore, the images in the articles are also collected, making the dataset multimodal.

As mentioned before, the final score of the articles ranges from 0 to 5. For the purposes of their experiments, the authors considered all news pieces scoring less than 3 to be fake news, and true news otherwise. The resulting dataset is balanced. Dai *et al.* treated the task as a binary classification problem, and accordingly carried out experiments using various approaches, including the aforementioned Social Article Fusion detailed by Shu *et al.* in [5]. In their experiments, SAF turned out to be the best model.

In this chapter we have presented various ways in of automated fake news detection. Broadly, these various fake news detection methodologies follow a similar

path, starting with detection of potential claims, comparing them with a database containing claims of known veracity, aggregating those comparisons into a statistic which gives an idea of the veracity of the unverified claims. As will be explained in the next chapter, our primary approach to medical claims verification follows the aforementioned method, and extends this approach to determine the overall veracity of online articles which may contain one or more unverified medical claim(s).

Chapter 3

MedFact for Neurodevelopmental Disorders

3.1 Introduction

Almost every disorder and disease is accompanied by a corresponding rise in fake news about it. The ease of access to social media has only made it easier to spread such fake news. Fake news is almost always detrimental to the health of the populace, as patients or their caretakers try every remedy on the internet, whether medically safe or not, to provide respite from the affliction. Often, fake news can discourage patients from seeking out treatment from legitimate sources, or completely disregard prevention strategies advocated by public health organizations like the CDC. This is especially true in cases where modern medicine is unable to cure disorders or diseases. An umbrella term for one such set of disorders is Neurodevelopmental Disorders or NDDs. The real world effect of fake news can be seen in the fake news "infodemic" which has accompanied the recent COVID pandemic. This "infodemic" has promoted vaccine hesitancy, unapproved and potentially harmful treatments like consumption of cow urine, disinfectants, etc. In such an environment, an automated process of fact-checking can help in countering the rapid spread of fake news online. Furthermore, an annotated test set, related to the field of NDDs, is required to test the aforementioned automated approaches to fact-checking.

3.2 Related Works

While there are a lot of datasets related to medical misinformation detection, most of them have limited sizes, numbering in a few hundreds. This is specially the case in annotating medicine related online articles. One of the major reasons is that unlike social media posts, articles contain more lines on average than said social media posts. That likely means greater time required to read and annotate those online articles. This problem is further complicated by the fact that medical experts are required to annotate these articles. These medical experts have very limited time, specially in midst of a once in a century pandemic.

Alsyouf *et al.* in [12] collected 50 articles which are related to Genitourinary Malignancies, and annotate them with the help of two physicians. Out of those 50 articles, 35 were classified as accurate.

Dai *et al.* in [11] used the healthnewsreview.org website, which were already contained articles annotated by the subject matter experts, to build their dataset of 2296 articles. healthnewsreview.org had been annotating articles from news outlets like CNN, FOX, NYT,etc., for greater than 10 years. Furthermore, their primary concern was not only to determine the veracity of information in those articles, but to assess the quality of said articles along a set of criteria, some of which are as follows:

- Whether the cost of treatment were discussed adequately
- Whether the novelty of the proposed treatment was adequately explained
- Whether there were any undeclared conflicts of interest

One of the reasons why they were not very worried about the veracity of information in the articles they annotated was because they collected articles from reputed sources like NYT, WaPo, and news releases by universities about their latest health related research. These news sources are very unlikely to have any outright fake news. This dataset would therefore be more useful to train models which determine the quality of

articles which have already been classified as containing true news by an automated or manual annotator.

Iglesias-Puzas *et al.* in [13] collected and annotated 385 online articles related to dermatology, which included topics such as "acne", "alopecia/hair loss", etc. These articles were annotated by two dermatologists, while a third dermatologist resolved any disagreements in annotations. These articles were classified into three classes, namely precise - if medical literature supports the information in the article, confused - if there is limited evidence for the claims in the article, imprecise - if there is no evidence or the medical literature contradicts claims made in the article.

Our fact-checking models introduced in this chapter make use of embedding models. Embedding models are trained to project human language (sentences, words, characters, etc.) into vector spaces that can then be used to build various machine learning models. The earliest attempts at creating embedding models sought to simply convert words or characters into one hot vectors (length of vector is equal to total vocabulary), and train models on top of such vectors. Another approach was to use n-gram models where probability distribution (a language model) of a series of words or characters is modelled from a large corpus [14]. These approaches, however, did not encode any concept of word similarity or semantics and thus their representative power was very limited. Furthermore, they were very sparse which hindered learning in downstream tasks. Mikolov, Sutskever, Chen, Corrado, and Dean then introduced a new way of creating language models through Continuous Bag of Words (CBOW) and Continuous Skip-Gram (CSG) algorithms [15]. CBOW, for instance, creates an embedding by training the model to predict the terms that appear in the context of a word in the training corpus. This helps model the context of each word, and two words with similar contexts tend to have similar semantics. The limitation of this representation is that it provides a static representation of each word in the vocabulary, but words' meanings can change depending on their contexts. For example, consider the two sentences:

- The **key** to success is hard work
- I can't find the **key** to this lock

The word **key** has different meanings in the two sentences, which depends on the other words that appear in the sentence. Thus, Word2Vec type vector representations which offer static vectors for each word regardless of its current context will find it difficult to model words in contexts that did not appear sufficiently large number of times in their training corpus. Hence, it is important that a word's vector representation dynamically changes with its context. The latest State-of-the-Art embedding models incorporate the context of a word into their embedding. These models are usually based on the Transformer architecture. The Transformer architecture was introduced by Vaswani, Shazeer, Parmar, Uszkoreit, Jones, Gomez, Kaiser, and Polosukhin in [16], and consists of an encoder which encodes text input into a vector representation and the decoder takes said vector representation and carries out down stream tasks like machine translation. BERT, and its variants, that make use of the encoder part of Transformers, have emerged to be the State-of-the-Art on various tasks in Natural Language Understanding [17]. Some of these BERT based models are publicly available on Hugging Face [1]. We have used some of these BERT based models as part of our model pipeline.

3.3 Methodology

In sub-section 3.3.1, we will explain the Methodology of the annotation of an NDD specific dataset, i.e., true and fake articles related NDD. Thereafter, in sub-section 3.3.2, we will explain the architectures of the models whose performance was validated and compared using the aforementioned dataset.

[1] https://huggingface.co/

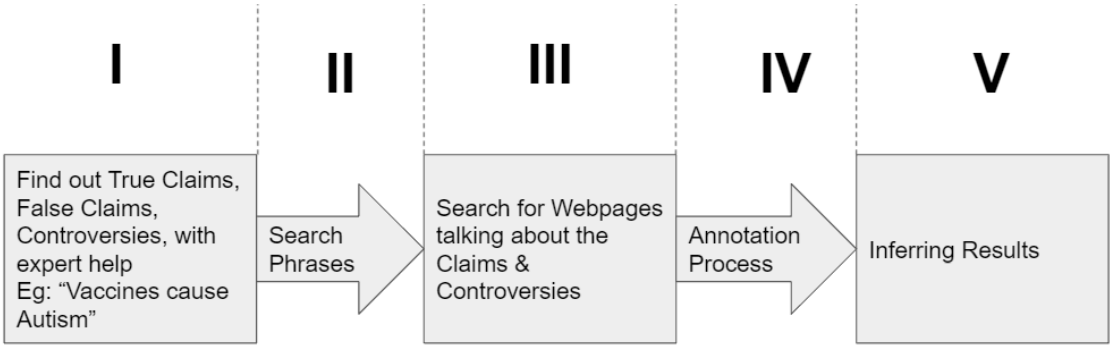


Figure 3.1: Annotation Procedure Overview

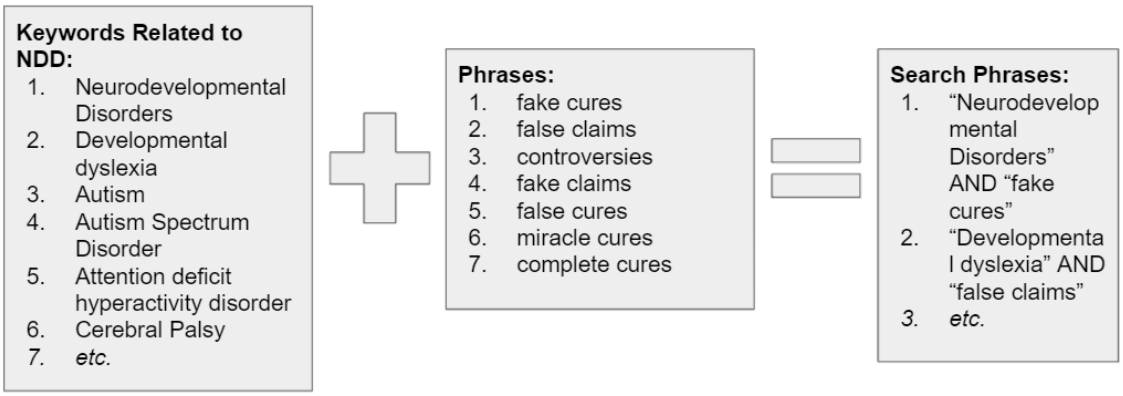


Figure 3.2: Stage I of Annotation Process

3.3.1 Dataset Annotation Procedure

This section contains the description of the annotation procedure of our Dataset related to NDDs. Figure 3.1 provides an overview of the five stages of the annotation procedure.

Stage 1: Find out True Claims, False Claims, Controversies, with expert help

The first stage of the annotation process, illustrated in figure 3.2 of the annotation process is to find out True, False, and Controversial claims from the medical literature and verify the same with experts in the field of neuro-science and/or NDD. We did this by creating search phrases which could be used to query academic papers debunking fake news, or discussing controversial claims. Claims were extracted from

these papers, and then annotated as true or false by subject matter experts.

Firstly, we identified the medical terms of the disorders which are considered to be under the umbrella term NDD. This was done by using the MeSh glossary provided by the NCBI [18]. This glossary assists in indexing publications for the Pubmed search engine [19], which searches for medical literature in the Medline database [20]. This indexing procedure requires that every major topic like NDD have a series of pre-identified keywords which define the major avenues of inquiry in that field. For instance, the Autism Spectrum Disorder (ASD) is a type of disorder under NDD and is a major field of study, in that, many publications deal with ASD. The accurate indexing of such studies requires that the ASD be considered a sub-topic or sub-heading under the term NDD. However, this means that all the terms under NDD are not necessarily describing a disorder. This is where experts associated with the project, specifically Dr. Francois Bolduc, identified the names of the disorders associated with the NDD from the myriad of keywords or phrases describing various avenues of study under NDD. Some of these keywords or phrases identified during the procedure were: "Neurodevelopmental Disorders", "Developmental dyslexia", "Autism Spectrum Disorder", "Attention deficit hyperactivity disorder", "Cerebral Palsy", etc.

Secondly, general phrases like "fake cures", "false claims", "controversies", "miracle cures", etc., were created. These phrases were to be appended to aforementioned medical terms to create a search query. For instance, a sample search query could have looked like "Neurodevelopmental Disorders" AND "fake cures". This search phrase requires a search engine to return all documents which contain both phrases in the quotation marks.

These search phrases were then used to query search engines like PubMed for academic documents which debunked and/or discussed fake news related to NDDs. The resulting papers were manually read, and relevant phrases stating the false claim being debunked or discussed in the study were extracted. However, these extracted phrases had to be checked by subject matter experts for to determine the veracity

of the extracted claims. This was necessary for two major reasons. Firstly, this extraction was not done by subject matter experts, but by the author of this report, and thus expert input was required to verify if the phrase had been correctly extracted. Secondarily, the medical claims considered to be false or questionable at the time of the publication of the study might have been reconsidered by the medical community in the light of new evidence. For instance, gene therapy was not proven to cure or reduce the symptoms of NDD, however, recent evidence and advancements in gene therapy have started to change that perception. Thirdly, the study itself could have erred in identifying false claims, even by the standards of the evidence available during publication. For instance, in 2010, Lancet retracted a highly controversial and inaccurate paper linking measles, mumps and rubella (MMR) vaccine to Autism, more than 10 years after its original publication in 1998 [21]. This publication has been responsible for a large part of the modern anti-vax movement. For all the aforementioned reasons, experts helped in the curation of the claims extracted from PubMed.

Some examples of the resultant phrases or claims are: "Treatments that shouldn't be used to treat dyslexia: eye exercises/vision therapy, Irlen lenses/filters", "HBOT is controversial cure for cerebral palsy".

Expert annotations were collected via a survey which was structured as follows. Firstly, the claims extracted from PubMed are presented. After each claim, a set of options are presented based on the Likert scale [22]. The options are as follows: "Disagree", "Neutral", "Agree", "I Don't Know". The annotators were supposed to select the options depending on their stance with respect to the corresponding claim, i.e., whether they disagreed, agreed, felt neutral about the claim, or didn't know enough about the claim to have a position on it.

The final option for each claim was selected on the basis of majority voting. Obviously, the veracity of the claims could be derived from the responses of the experts; the claim would be classified as false or true if the majority of the experts disagree

or agree with the claim respectively.

Stage 2: Search Phrases

In this phase we convert the claims, from the previous section, into search phrases. We simply do that by breaking up the claims into their constituent medical words and combining them with the AND operator. For instance, the claim "Vaccines cause Autism" can be converted into the search phrase " "vaccines" AND "autism" ". When used in a search engine, this search phrase would allow for the retrieval of only those entries which contain the words "vaccines" and "autism", not necessarily in the same sentence.

Phase 3: Search for Websites talking about the Claims & Controversies

The search phrases, derived from claims in the previous phase, can now be used in search engines like Twitter and Google to retrieve relevant webpages. While Google directly provides these webpages, Twitter can also contain links to webpages. These webpages were then manually checked to see if they pertained to the claim for which they were extracted. This was done to reduce the number of webpages to be processed in the following annotation procedure.

Phase 4: Stance Annotation

At this point, we have a series of claims and their corresponding webpages. The stance annotation process involves annotating whether the whole or part of a webpage agrees or disagrees with the corresponding annotation. For the benefit of the annotators, the relevant keywords common between the claim and the webpage had already been highlighted.

Phase 5: Inferred Results

The final classification of the webpages was inferred on the basis of whether they agreed or disagreed with respect to the claim. If a webpage agreed with a false claim,

it was classified as a "Fake Article", else as a "True Article" if it disagreed. Similarly, a webpage that agreed with a true claim was classified as "True Article", else as a "Fake Article" if it disagreed with the true claim.

The Dataset finally contained 76 Fake Articles (articles that contain at least one false claim), and 40 Real Articles (articles that contain at least one true claim) - for a total of 116 annotated articles.

3.3.2 General Model Pipeline

All our model architectures will follow the same general architecture. In this section we will introduce said General Model Architecture by explaining each one of the modules, depicted as rectangular boxes, in Figure 3.3.

Sentence Selection Process

This is the first module in the pipeline which takes an Unverified Article as an input. The article into its constituent m sentences $s_1, s_2, s_3, ..., s_m$, and fed to this module. The Sentence Selection module then selects n sentences, $s_1, s_2, s_3, ..., s_n$, a subset of m article sentences. Following are two ways in which this sentence subset is selected:

- Extractive Summarization based Sentence Selection: The simplest way to select a subset of sentences is carry out Extractive Summarization. Extractive summarization derives the summary by extracting relevant sentences from the input it is provided, as opposed to Abstractive Summarization where the summary is created through auto-regressive generation and may contain phrases and sentences not present in the original input. Thus, Extractive Summarization was the obvious choice since the abstractive approach may introduce new information into the summary that was not present in the unverified article, which would introduce inaccuracy in the fact-checking procedure.

  The Summarization is carried out using the Bert Extractive Summarizer library [23]. Only the first three sentences of the summarizer's output are used

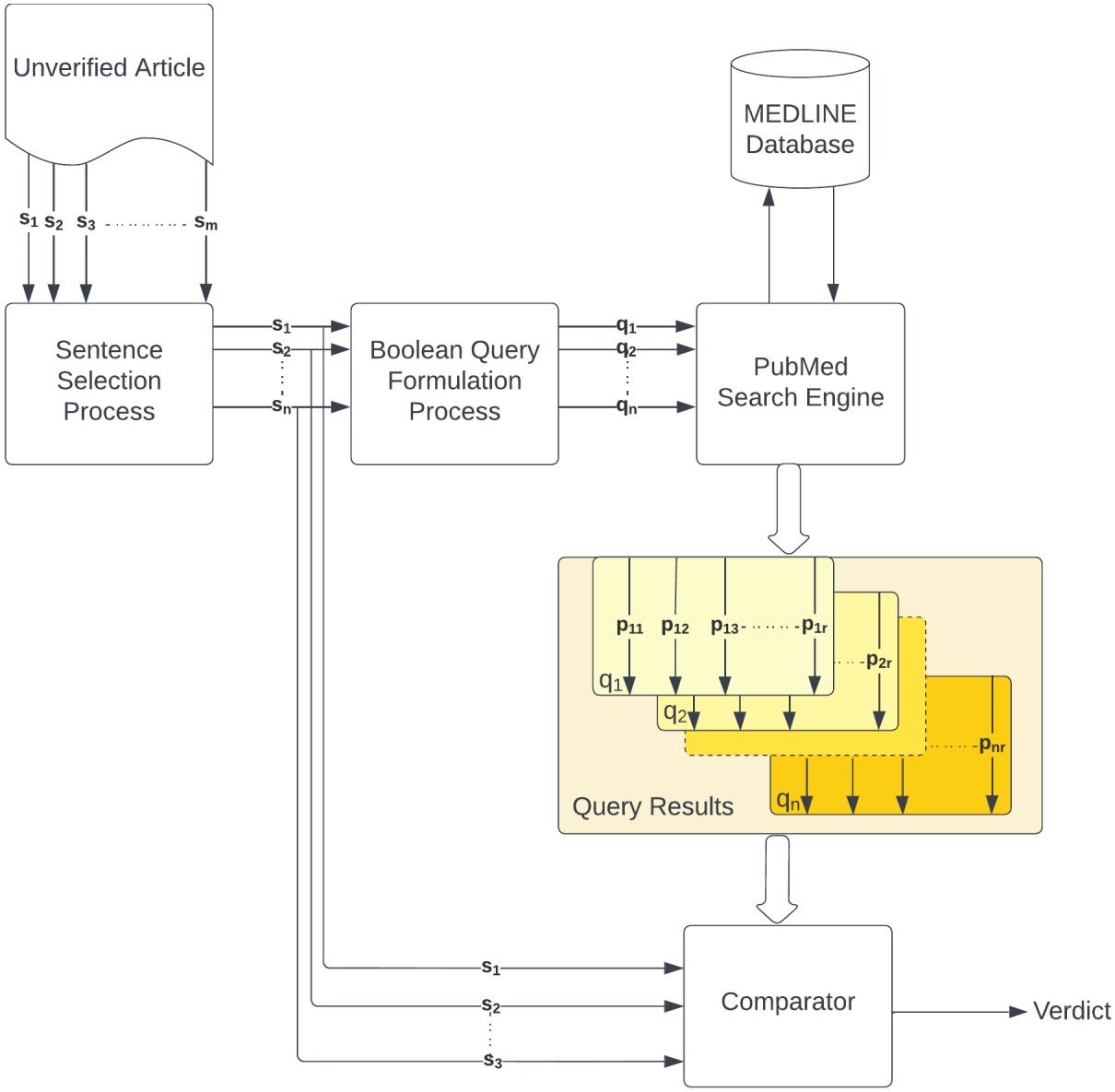


Figure 3.3: General Model Pipeline

for further processing.

- Stanza Based Sentence Selection: The Stanza library [24] provides models trained on different Named Entity recognition tasks in the biomedical domain. We utilize two of their pretrained models:

  - I2B2 model: This model is trained to detect and group named entities into categories like symptoms, treatments, problems, etc. in clinical reports.
  - Disease model: the other model is trained to detect disease names, for instance, Autism, Cerebral Palsy, Malaria, etc.

  The I2B2 and Disease models are used in conjunction with each other to determine the relevant named entities in each sentence of the unverified article. The I2B2 model is first used to retrieve all the named entities that belong to one of the following groups symptoms, treatments and problems in one sentence. Thereafter, the Disease model is used to detect disease named entities. All the diseases thus detected are removed from the problems group previously detection by the I2B2 model in the same sentence. Thus, finally we have at most four distinct groups of named entities for each sentence, namely, symptoms, treatments, problems and diseases. Only those sentences that contain named entities from at least two of the aforementioned entity groups, are considered for further NLP operations in the pipeline.

Boolean Query Formulation Process

In order to get search results from the MEDLINE databast [20], we need to provide the PubMed search engine with a Boolean query. A Boolean query consists of a series of keywords joined with Conjunctions and/or Disjunctions. This module takes one or more sentence(s), $s_1, s_2, s_3, ..., s_n$ as input and outputs one query for each sentence, $q_1, q_2, q_3, ..., q_n$, as shown on in Figure 3.3. There are two ways in which we accomplish this task:

- YAKE! based Boolean Query Formulation (YAKE! Query): YAKE! [25, 26] is an unsupervised keyword extractor, which we used to get the relevant keywords from the unverified article. YAKE! returns keywords in a decreasing order of relevance, but it was noticed that useful keywords were appearing lower down the relevance list while Named Entities like dates and person names were rising to the top. To remedy this, the named entities from the entire text of the article were detected using SpaCy's [27] default Named Entity Recognition (NER) model, and these entities were then removed from the list of relevant keywords returned from YAKE!.

  Since, our use case dealt with the detection of misinformation related to Neurodevelopmental Disorders (NDDs), we created a set of keywords/keyphrases describing the various disorders that comprise NDDs. A few of them are: Autism, Dyslexia, Dyspraxia, Attention Deficit Hyperactivity Disorder, etc. Thereafter, we removed any keywords in the intersection of the processed YAKE! output and the NDD keywords list from the former.

  We then combined the first three keyphrases with a disjunction, i.e., an "OR" operator. Thereafter, another part of the query was created by joining the NDD keywords, found in the YAKE! list earlier, with a disjunction as well. Finally, the two disjunctions thus formulated are combined with a conjunction, i.e., an "AND" operator.

- Stanza based Boolean Query Formulation (Stanza Query): This Query Formulation is a continuation of the Stanza based Sentence Selection explained in 3.3.2. After retrieving sentences and their corresponding named entities belonging to predefined categories (namely symptoms, treatments and problems) as detailed in the aforementioned section, the query to search the PubMed database is formulated by joining the named entities in each entity category with a disjunction, or an "OR" keyword, and then joining the resultant queries

for each group with a conjunction, or an "AND" keyword.

As opposed to YAKE! Query, the Stanza Query is specific to each sentence since the latter depends on the words that are found in sentence for which the query is being formulated. On the other hand, YAKE! Query is formulated by taking the entire article into consideration. This difference will become clearer when we explain our final models.

PubMed Search Engine & Query Results

The queries formulated in section 3.3.2 are then passed on to the PubMed search engine using the Python API Bio-Entrez [28]. PubMed in turn retrieves relevant results from MEDLINE [20], a database of biomedical literature maintained by the National Library of Medicine (NLM) [29]. Thus, for each query q_i in $q_1, q_2, q_3, ..., q_n$ we take the top r (in out case, 20) abstracts out of all the abstracts that PubMed returns, $p_{i1}, p_{i2}, p_{i3}, ..., p_{ir}$.

Comparator

Taking n selected sentences from Section 3.3.2, and the last 3 sentences from the corresponding r medical abstracts for each sentence from Section 3.3.2 as input, the comparator compares the sentences with their respective query results to determine whether the latter supports the former or not. The last three sentences of an abstract usually contain the conclusion of the medical study, thus it is the most important portion of the abstract for the purposes of fact-checking. If it is determined that the medical literature supports the conclusion in the corresponding sentence then it contains a true claim, else it contains a false claim. Using these sentence wise verdicts, the comparator carries out a simple majority voting between sentences determined to contain true or false claims, to determine the article level verdict, i.e., whether the article is true or false. There are two ways in which we determine the sentence level verdicts:

- Sentiment Matching Comparator: The sentiment exhibited by the last three sentences of each medical article is determined through a pre-trained transformer based sentiment analyzer available on HuggingFace[2]. This sentiment analysis model is used to determine the sentiment of the sentence in the unverified medical article. The sentiment of the medical article is determined to be negative if any of its last 3 sentences are predicted as having negative sentiment by the sentiment classification model, otherwise the sentiment is classified as positive. If the sentiment of the sentence in the unverified article and the corresponding medical article matches, then the medical article supports the claim made in the former, otherwise it does not. The final veracity of the sentence is determined by a majority voting between all medical abstracts with matching and non-matching sentiments. Thereafter, the overall unverified article veracity is determined by another majority vote by the veracity of the individual sentences determined before.

- Inference Comparator: Inference models are trained to predict whether two sentences agree, disagree or are neutral to each other. We used a Sentence Transformer based inference model [3]. The last three sentences of each of the medical article retrieved, are compared to the first three sentences produced by a transformer based summarizer through the aforementioned inference model.

3.3.3 Model Pipelines

In this section we will explain the four model pipelines we derived by using the various modules explained in section 3.3.2.

- Yake! Query+Inference: This model is a combination of the following modules: Extractive Summarization based Sentence Selection, YAKE! Query, Inference Comparator, which were explained in Sections 3.3.2, 3.3.2 and 3.3.2 respectively.

[2]sentiment-roberta-large-english: https://huggingface.co/siebert/sentiment-roberta-large-english
[3]https://huggingface.co/cross-encoder/nli-deberta-v3-base

The YAKE! based query formulator determines the query on the basis of the entire article. The last three sentences of the medical abstracts retrieved using the aforementioned query are compared to the first three sentences of the Extractive Summarizer's output using the Inference Comparator. Note that the comparison is not between three pairs of sentences, but between two pieces of text containing three lines each. Thereafter, the article level veracity is determined by majority voting between the number of sentences determined to be false or true.

- YAKE! Query+Sentiment Matching: This pipeline is the same as "Yake! Query+Inference", except that the last module in the pipeline is the Sentiment Matching Comparator instead of the Inference Comparator.

- Stanza Query+Sentiment: This model is a combination of the following modules: Stanza Based Sentence Selection, Stanza Query, Sentiment Matching Comparator, which were explained in Sections 3.3.2, 3.3.2 and 3.3.2 respectively. Sentences selected by "Stanza Based Sentence Selection" are passed on to the Stanza Query, which then formulates a query for each sentence. Medical abstracts retrieved for each query are then compared to the corresponding selected sentence in the unverified article by using the Sentiment Comparator.

- Stanza Query+Inference:This pipeline is the same as "Stanza Query+Sentiment", except that the last module in the pipeline is the Inference Comparator instead of the Sentiment Matching Comparator.

3.4 Experiments

We conducted two sets of experiments to test our models. Firstly, we tested the models, described in Section 3.3.3, on our NDD-focused dataset. Precision, Recall, and F1-Score metrics were collected to compare the models, and the results are presented

and discussed in Section 3.5.

Secondly, we tested Stanza Query+Sentiment model on an unannotated set of transcripts of YouTube [4] videos. These videos were mostly related to the topic of sleep problems that accompany cognitive disorders. Furthermore, since YouTube videos do not come with punctuation, we added them using the rpunct library [5] before running our models on them. The objective of this experiment is to dive deep into the model and show some examples of the sentences the model selected from the transcripts for fact-checking, the corresponding medical literature retrieved, and the verdicts delivered. The aim of running our model on YouTube transcripts is to demonstrate the following:

- Our model is applicable to Videos if their transcripts are available
- Our model is able to identify medical claims in text and retrieve relevant abstracts

We chose the Stanza Query+Sentiment Matching Model and not YAKE! Query+Sentiment Matching Model because most of the YouTube videos we are testing on, are not limited to Neurodevelopmental Disorder topics. As mentioned before, they are a combination of Sleep issues and cognitive disorders, and as explained in Section 3.3.2, models that included YAKE! based Query Formulation in their pipeline required a set of predefined Neurodevelopmental Disorder terms (eg, Autism, ADHD, etc), which occur in the dataset. Such a set of disorder terms is not available for the YouTube dataset and thus precludes the use of YAKE! Query+Sentiment Matching Model.

3.5 Results & Discussion

In this section we will present and discuss the results of running our models on the NDD and Youtube datasets. While all models/pipelines were tested on the

[4] https://www.youtube.com/
[5] https://pypi.org/project/rpunct/

NDD Dataset, only the Stanza Query+Sentiment Matching Model was tested on the Youtube Dataset.

3.5.1 NDD Dataset Test Results

	Fake Article			Real Article		
Model	Precision	Recall	F1-Score	Precision	Recall	F1-Score
YAKE! Query+Inference	0.706	0.632	0.667	0.417	0.500	0.455
YAKE! Query+Sentiment	**0.816**	0.816	0.816	0.650	**0.650**	**0.650**
Stanza Query+Sentiment	0.805	**0.868**	**0.835**	**0.706**	0.600	0.649
Stanza Query+Inference	0.667	0.684	0.675	0.368	0.350	0.359

Table 3.1: Precision and Recall for Fake and Real Articles

The precision, recall and F1-score related to Fake and Real articles' detection respectively, are displayed in Table 3.1. Overall Stanza Query+Sentiment performs the best in terms of Fake Articles detection F1-Score and has 0.649 Real Article detection score as opposed to 0.650 of YAKE! Query+Sentiment.

Sentiment Matching vs Inference

Using Sentiment as a tool for comparison yields better results than Inference. This is true, regardless of the query formulation method. For instance, Yake Query+Sentiment produces better results across the board than Yake Query+Inference. Similarly, Stanza Query+Sentiment has better performance than Stanza Query+Inference for all metrics considered. In fact, both Yake Query+Sentiment and Stanza Query+Sentiment perform better than Yake Query+Inference and Stanza Query+Sentiment.

While using models trained on sentence pair entailment datasets seems to be a logical choice for our usecase, they may not always be able to detect entailment accurately. This may be due to various reasons. In our experiments we take the last three sentences of medical abstracts which may not contain sentences that clearly

entail or contradict the unverified article sentence. The medical literature may use synonyms to refer to named entities which could throw off the entailment detection model.

On the other hand, using the Sentiment matching approach as a proxy for entailment performed better since this approach does not rely too heavily on whether the unverified article sentence and the medical literature are using the same synonyms or are clearly entailing or contradicting each other. As long as the articles retrieved are relevant, the sentiment of the conclusion gives an indication of whether the authors of the medical report feel positively or negatively about the relation between the entities that make up the query.

YAKE! Query vs Stanza Query

Given Sentiment based sentence comparator, Stanza Based Query Formulation (Stanza Query) performs better than Yake Based Query Formulation (Yake Query) in terms of F1-Score for Fake Article detection, and vice-versa for Real Article Detection.

Given Inference based sentence comparator, Stanza Based Query Formulation again (Stanza Query) performs better than Yake Based Query Formulation (Yake Query) in terms of F1-Score for Fake Article detection, and vice-versa for Real Article Detection.

All other things equal, Stanza Query performs worse than YAKE! Query in terms of F1-Score for Real Articles. This has to do with the nature of the dataset that contains real articles most of which debunk the claims made in fake articles (which are also in the dataset). Therefore, these real articles tend to have false claims stated verbatim in their text. Since Stanza Query based model goes through the article line by line, as opposed to YAKE! based Query which only considers the summary, the former gets misled by the false claims in otherwise real articles and erroneously labels those articles as false.

3.5.2 YouTube Dataset Test Results

Table 3.2 presents some of the snippets from transcripts of the YouTube videos, and their corresponding relevant medical literature retrieved by Stanza Query+Sentiment Model. The transcripts were from YouTube (YT) videos related to Sleep Disorders and Autism.

The sentences in bold-face in the Transcript Snippet column of Table 3.2 were selected by Stanza Query+Sentiment Model for further processing in the model pipeline. The Title and Abstract Snippet column contain the title and abstract snippet, respectively, retrieved from MEDLINE by PubMed.

The first snippet in Table 3.2 is related to sleep, and talks about how sleep is important for survival and memory consolidation. The retrieved article titled "Functions of Sleep" makes a similar point.

The second snippet asserts application of a medical device called CPAP improves oxygen levels (in sleeping patients). Again, the retrieved literature is relevant to the transcript snippet. The paper titled "Comparison of positional therapy versus continuous positive airway pressure in patients with positional obstructive sleep apnea: a meta-analysis of randomized trials", discusses how CPAP is superior to positional therapy in "increasing the oxygen saturation in patients with positional OSA". Obstructive Sleep Apnea or OSA is a type of Sleep Apnea [30].

	Relevant Medical Literature Retrieved	
Transcript Snippet	Title	Abstract Snippet
...tends to consolidate it, making more room for information that we did. **we do need to store for survival, so memory consolidation is another useful and potentially helpful mechanism that helps explain the process of enda need for sleep.** ok, another very important concept that patients often may ask me...	Functions of Sleep	Our article will specifically focus on role of sleep in neuronal development, synaptic plasticity, memory consolidation or mental health in general. Its role in immune system functioning will also be mentioned. Moreover, we will also consider more general functions of sleep, such as well-being of the organisms or securing survival of the individual. In conclusion, we will highlight possible main function of sleep.
...dropping to a very dangerous level. **and with the application of cpap, you improve the oxygen level.** and you also improve the patient's...	Comparison of positional therapy versus continuous positive airway pressure in patients with positional obstructive sleep apnea: a meta-analysis of randomized trials	Positional therapy showed higher AHI (mean difference, MD: 4.28, 95% CI: 0.72-7.83) and lower oxygen saturation level (MD: -1.04, 95% CI: -1.63 to -0.46) than CPAP. It showed no distinct advantage over CPAP in terms of arousal index, sleep efficiency, and total sleep time, but CPAP reduced sleep time in the supine position. Conclusion: CPAP is superior to positional therapy in reducing the severity of sleep apnea and increasing the oxygen saturation level in patients with positional OSA.

	Relevant Medical Literature Retrieved	
Transcript Snippet	Title	Abstract Snippet
...for some people, it happens hundreds of times throughout the night. **if it's not treated, sleep apnea can lead to high blood pressure, stroke, or memory loss.** some people have sleep apnea and they don't even know it...	Influence of Obstructive Sleep Apnea Severity on Muscle Sympathetic Nerve Activity and Blood Pressure: a Systematic Review and Meta-Analysis.	These data are clinically important for understanding cardiovascular disease risk in patients with OSA.
...and then we wake up again around 9:00 a.m. thanks to another neurotransmitter called histamine. **and that's when melatonin levels begin to drop, preparing us to start the wake period just around the morning time, and this is helpful because it helps you appreciate what is the anticipated level of sleepiness and wakefulness during the 24-hour period as you try to know what what is really normal baseline as opposed to what's pathologic.** what you can appreciate is that certain periods of the 24-hour cycle are likely to put people...	What keeps us awake? The role of clocks and hourglasses, light, and melatonin	This in turn can lead hourglass processes, as indexed by accumulated homeostatic sleep need over time, to strongly oppose the clock. To add to the complexity of our sleep and wakefulness behavior, light levels as well as exogenous melatonin can impinge on the clock, by means of their so-called zeitgeber (synchronizer) role or by acutely promoting sleep or wakefulness. Here we attempt to bring a holistic view on how light, melatonin, and the brain circuitry underlying circadian and homeostatic processes can modulate sleep and in particular alertness, by actively promoting awakening/arousal and sleep at certain times during the 24-h day.

Transcript Snippet	Relevant Medical Literature Retrieved	
	Title	Abstract Snippet
	Melatonin: role in gating nocturnal rise in sleep propensity	Based on these findings and on the precise coupling between the endogenous nocturnal increase in melatonin secretion and the opening of the sleep gate, it is suggested that melatonin participates in the regulation of the sleep-wake cycle by inhibiting the central nervous system wakefulness generating system. This inhibition allows a smooth transition from wakefulness to sleep. Clinical findings on decreased levels of nocturnal melatonin in chronic insomniacs, and on the efficacy of exogenous melatonin in improving sleep in melatonin-deficient insomniacs, are congruent with this hypothesis.
...during their first three years of the child's development. **although autism is congenital, signs of the disease can be difficult to identify and diagnose.** during infancy are foiled. who is the second one...	Family history of autoimmune diseases is associated with an increased risk of autism in children: A systematic review and meta-analysis	The results varied in some subgroups. Conclusion: An overall increased risk of autism in children with family history of ADs was identified. More mechanistic studies are needed to further explain the association between family history of ADs and increased risk of autism in children.
	Identification of Chromosomal Regions Linked to Autism-Spectrum Disorders: A Meta-Analysis of Genome-Wide Linkage Scans	Finally, region 8p21.1-8q13.2 reached significant linkage peak in all our meta-analyses. When we combined all available genome scans (15), the same results were produced. Conclusions: This meta-analysis suggests that these regions should be further investigated for autism susceptibility genes, with the caveat that autism spectrum disorders have different linkage signals across genome scans, possibly because of the high genetic heterogeneity of the disease.

Table 3.2: Some Samples of Relevant Literature Retrieved by Stanza Query+Sentiment Model

3.6 Conclusion

In this chapter we introduced a dataset and multiple post hoc explainable models, and compared the models' performance on said dataset. We discussed the methodology of the collection and annotation of the Dataset. We showed that Sentiment matching improves model performance over sentence pair inference models, and that the combination of Stanza Query & Sentiment matching leads to the best results. We analyzed the reasons for Sentiment matching approach's advantage over the inference models and the substantial gap between precision and recall results of models based on Stanza Query Formulation. Finally, we showed that Stanza Query+Sentiment Model can be run on YouTube videos' transcripts and retrieve relevant medical literature for fact-checking purposes.

Chapter 4

Analysis of COVID-19 Misinformation in Social Media using Transfer Learning

4.1 Introduction

The COVID-19 pandemic has played out as an infodemic, with misinformation, disinformation and rumours rapidly spreading on various facets of the disease such as origin, causes, symptoms, prevention, and treatments [31]. This has significantly hampered the global public health response. Social media is a popular way of communication, but uncertainties during the pandemic have caused proliferation of harmful health misinformation posts via platforms such as Reddit, Twitter, and Facebook, among others [32]. There are a number of topics fueling COVID-related misinformation, ranging from conspiracy theories, misreporting of morbidity and mortality, disease spread mechanisms, prevention methods, treatments and drugs, recovery experiences, and political controversies [33].

Although misinformation spreads both online and offline, the propagation and contagion of misinformation are more pronounced in social media platforms [34, 35]. Therefore, a critical understanding of the methods to detect misinformation in various social media platforms is a precursor to the design and implementation of effective health promotion policies [36]. One of the earlier attempts to detect health misinfor-

mation used Twitter data; Castillo et al. extracted multiple features from trending topic posts to classify the messages based on credibility [37]. Since then, there have been several interdisciplinary studies using social media (mostly using Twitter data) to understand the spread of misinformation [38–43], ranging from experiments on attitudes towards fake news [44], to public health policy frameworks [45], and conceptual theories in information and knowledge management [36]. In the field of Natural Language Processing (NLP), researchers have worked on building datasets related to misinformation, including representing with GloVe embeddings to find relevance between posts and misinformation [46]. Others have collected fact-checked articles covering a broad range of topics, including political and medical discussions [47, 48].

In this chapter different supervised classification models were explored, namely Support Vector Machine (SVM) and Multi-Layer Perceptron (MLP), trained on different embeddings. Various combinations of embeddings were used to determine whether these produced significant improvements in comparison to their constituent parts. This has practical implications during a pandemic when fact-checking activities are usually manual, and therefore, time-consuming, labour intensive and expensive. Insights from this study can help with the development of automated systems which reduce the workload of manual fact-checking to clarify and debunk different types of misinformation.

4.2 Related Works

The challenges of generic fake news detection from an NLP perspective can be categorized into four areas: fact-checking, rumor detection, stance detection, and sentiment analysis [49]. To facilitate the formulation of fake news as a supervised classification or regression task, various types of datasets have been used in literature, ranging from labelled short claims, e.g. PolitiFact and Snopes, to entire-article datasets where the whole article is either true or false. Labelled datasets for fake news detection in social networking services are limited. Various methods for general-purpose fake news

detection have been utilized in literature, including machine learning models with and without neural networks, rhetorical approaches with Rhetorical Structure Theory (RST) to define the semantic role of text units and the overall coherence of a story, as well as Recognizing Textual Entailment (RTE) to recognize relationships between sentences.

On the area of health misinformation, there is considerable work in literature, mainly covering vaccinations and infectious diseases. The findings from the related papers show notable prevalence of misinformation within social media posts [34]. Diving further into the specific topic of COVID-19 misinformation, one of the challenges has been inaccurate news coming from reputable sources on developing stories, such as efficacy of anti-inflammatory drugs [50]. At the same time, medical professionals have utilized social media more than ever before for sharing professional opinions and democratizing access to scientific data [50–52].

On the subject of COVID-19 misinformation, recent studies have also attempted to tackle this research challenge. Meng et al. fuse general embedding-based RoBERTa and COVID domain-specific embedding COVID-Twitter-BERT (CT-BERT) [53] using a simple MLP. The authors carried out the experiments on the aforementioned dataset to demonstrate that the combination of general and context specific embeddings marginally improves the performance of a classification model. However, they did not demonstrate that these improvements were statistically significant. Wani et al. [54] compare the performance of models based on general GloVe embeddings and domain specific fastText embeddings, which were trained on an non-annotated dataset of 179k COVID-related tweets posted by Gabriel Preda on Kaggle [55]. The word embeddings were not combined in any way by the authors. However, the context-specific fastText embedding did produce better results than the general GloVe embeddings. Here again no testing was done to determine whether those gains were statistically significant. The same COVID-related Kaggle dataset from [55] has been used for creating context-specific word embeddings, as explained later in the chapter.

From the sociological perspective, studies have shown that social media users share posts with misinformation mainly due to inattention to detail rather than any malicious intent [56]. Essentially, what people share on social media is not always what they believe. In the light of this, misinformation detection research can highlight trends that need public health interventions and repeated messaging [56]. Another factor to take into account is the sense of desperation that could be making people susceptible to misinformation. Research has shown that parents of ailing children are more likely to fall for online misinformation owing to desperation in finding treatments for chronic diseases like cancer [57]. In light of these factors, proper interventions on trending misinformation posts can help users to think more carefully about the accuracy of information they consume. This chapter shines a light on ways of improving misinformation detection on social media, thereby aiding effective public health responses. Additionally, once misinformation is identified, health professionals can also be enabled to engage with patients in social media to counter trending misinformation topics. As an example, pediatric infectious disease specialists have been proposed as a solution to social media misinformation about COVID-19 related to children and parents [58]. Ultimately, users consuming or spreading misinformation are usually not malicious, and once misinformation is detected, subject matter experts can be used to counter the same.

4.3 Methodology

This section is laid out as follows - firstly, the details of the dataset and preprocessing done on the same are discussed, then the configurations for word embeddings are explained, thereafter the transfer learning approaches utilized are described, and finally the experiments conducted, to find out if the different embeddings significantly improve the weighted F1 score, are detailed.

4.3.1 Dataset and Preprocessing

The dataset used in the experiments was released at the CONSTRAINT 2021 workshop colocated with the 35th AAAI Conference on Artificial Intelligence [59]. Henceforth, the dataset shall be referred to as the CONSTRAINT_2021 dataset. This dataset contains 6420 and 2140 social media posts in the train and test sets respectively. These posts have been labelled as "real" and "fake". The "real" posts were collected from official and verified Twitter handles, including government accounts, medical institutes, etc. [2]. The "fake" posts were collected from fact-checking websites such as Politifact, NewsChecker, and Boomlive. Fact-checked "fake" posts were collected regardless of the social media platforms that they were posted on, in contrast to the "real" social media posts which are entirely from Twitter. The combined CONSTRAINT_2021 dataset contains a total of 8,560 posts, of which 4,080 (47.7%) are "fake" posts, and 4,480 (52.3%) are "real" posts. All punctuation and standard stopwords were removed from the tweets. Thereafter, the tweet level representation was calculated and concatenations done as explained in the proceeding sections.

4.3.2 Twitter Embeddings

Word embeddings are vector representations of words where words with similar meaning share similar vector spaces. There are different ways of creating vector these representation. Word2vec embeddings [60] were used for all the following experiments. Specifically, two word embeddings were used, namely General Twitter Embedding (GTE) and the Context-Specific Embedding (CSE), which were further used to derive all the other Twitter embeddings for the experiments.

General Twitter Embedding (GTE)

For the general/universal embedding, a General Twitter Embedding (GTE) introduced in [61] was used. This embedding was trained on a corpus of 400 million tweets, and has a vocabulary size of 3 million words.

Context-Specific Embedding (CSE)

A corpus of tweets related to COVID-19 posted on Kaggle [55] was chosen to create embeddings specific to the COVID-19 context. This corpus contained 179,108 tweets spanning between 29th February, 2020 and 24th July, 2020. These tweets were used to create a Word2vec embedding of vector size 200. This embedding has a vocabulary size of 22,012 words, which is substantially lesser than the vocabulary of size of the GTE.

Tweet Level Vector Representations

This representation is created by taking the average across the word vector representations of all the words left in the tweet after pre-processing, provided said words are also in the vocabulary of the Word Embedding. This results in a single vector representation for each tweet, whose dimension will be equal to those of the individual words in the word embedding from which they were created.

4.3.3 Transfer Learning

Word embeddings that are learned on a small corpus [55], corresponding to a particular task, are then used to transfer the knowledge from the larger embedding space represented by a general corpus, thereby improving the representational power of the general embedding for a specific task. This transfer learning is creating representations of words in the context of the COVID-19 pandemic discourse on social media. Such an embeddding can take advantage of the large vocabulary size of a general embedding and the representational accuracy of the context specific word embedding. This transfer learninf is carried out in two distinct ways. Firstly, by using the method of transfer learning explained in [1], which will be referred to as Augmentation Transfer Learning (ATL). Secondly, by using the concatenation of general and context specific embeddings to create new embeddings. This process will be called Concatenation Transfer Learning (CTL).

Augmentation Transfer Learning (ATL)

Firstly, the common words between the vocabularies of GTE and CSE are determined. The word embeddings of these common words are then used to train a simple neural network with ReLU activation function, which takes GTE word embeddings as input and the CSE word embeddings as the target output. This neural network, trained on common vocabulary word embeddings, can now be used to find the context-specific word embeddings of all the corresponding word embeddings in the GTE vocabulary, thereby creating a third representation called Augmented Twitter Embedding (ATE), which has the same vocabulary as GTE, but a vector size of CSE. This process by which the ATE is created is termed the Augmentation Transfer Leaning (ATL). The resulting ATL embedding does not have any duplicate words in its vocabulary.

Concatenation Transfer Learning (CTL)

The General Twitter Embedding (GTE), Context Specific Embedding (CSE), and Augmented Twitter Embedding (ATE) are used to create concatenated embeddings. To create the concatenated tweet level vector representations, the tweet level vector representations explained in sub-section 4.3.2, are concatenated with each other. Two tweet level embeddings are created via this process of concatenation, one by concatenating GTE and ATE, called GTE+ATE, and the other by concatenating GTE and CSE, called GTE+CSE.

Overall, 5 different types of word embeddings were used to derive tweet level embeddings: (1) GTE, an off-the-shelf general word embedding for tweets, (2) CSE, a context specific word embedding for COVID-19 related tweets, (3) ATE, a context specific word embedding, but with a larger vocabulary than CSE, (4) GTE+CSE, a concatenation of the tweet level embeddings derived from GTE and CSE word embeddings, and (5) GTE+ATE, a concatenation of the tweet level embeddings derived from GTE and ATE word embeddings.

4.3.4 Experiments

Our objective with these experiments were two-fold: firstly, to determine if there was an improvement in the performance (determined by the weighted F1 score) of classification models when trained using embeddings created through Augmented Transfer Learning (namely ATE) and/or Concatenated Transfer Learning (namely GTE+CSE & GTE+ATE) vis-a-vis the embeddings from which the aforementioned embeddings were derived (namely GTE, CSE, and ATE). And secondly, to determine if the improvements thus produced were statistically significant.

In order to meet both these requirements, train and test sets of the CONSTRAINT_-2021 dataset are combined, and 5x2 Cross Validation (CV) tests are carried out. The 5x2 CV involves carrying out 2-fold cross validation on the combined dataset across 5 iterations, with the dataset getting shuffled at every iteration. In each of the 5 iterations all the models are trained on one half of the dataset, and tested on the other half. In the same iteration, the training and testing halves are then swapped, and the models are trained on the erstwhile testing half and tested on the erstwhile training half. This produces a total of 10 test results for each model. These 10 test results can then be used to test for whether a pair of models are statistically significantly different or not. This significance testing method, called the Combined 5x2 CV f-test, is done via the series of formulae specified in [62]: Let there be two models, namely A & B which need to be compared for statistical significance.

$$p^{(1)} = p_A^{(1)} - p_B^{(1)} \tag{4.1}$$

In Equation 4.1, $p_A^{(1)}$ is the vector of 5 weighted F1 scores which model A produced over the test set in the first split of the 2-CV, in each of the five iterations. Similarly for $p_B^{(1)}$. $p^{(1)}$ is the element-wise subtraction between $p_A^{(1)}$ & $p_B^{(1)}$.

$$p^{(2)} = p_A^{(2)} - p_B^{(2)} \tag{4.2}$$

In Equation 4.2, $p^{(2)}$ is similar to $p^{(1)}$, except the weighted F1 scores involved in this calculation were calculated in the second split of the 2-CV, in each of the 5 iterations.

$$\bar{p} = \frac{p^{(1)} + p^{(2)}}{2} \tag{4.3}$$

In Equation 4.3, $\bar{p}$ is the element-wise mean of the element-wise differences $p^{(1)}$ & $p^{(2)}$.

$$s^2 = \left(p^{(1)} - \bar{p}\right)^2 + \left(p^{(2)} - \bar{p}\right)^2 \tag{4.4}$$

In Equation 4.4, s^2 is the element-wise variance of the element-wise differences, $p^{(1)}$ & $p^{(2)}$. Finally, the f-statistic is calculated as follows:

$$f = \frac{\sum_{i=1}^{5}\sum_{j=1}^{2}\left(p_i^j\right)^2}{2\sum_{i=1}^{5} s_i^2} \tag{4.5}$$

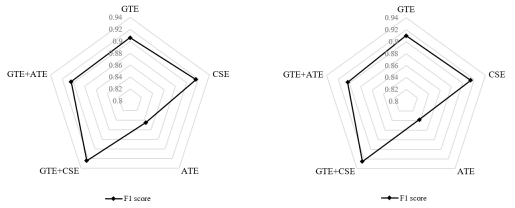
The f-statistic is distributed with 10 and 5 degrees of freedom, for the numerator and denominator respectively. Those degrees of freedom along with the value of the f-statistic are used to determine the p-value for a pair of models. The two models in consideration are determined to be significantly different if the corresponding p-value < 0.05, i.e., the null hypothesis, that the two models are similar, is rejected. This significance testing is carried out for every possible pairing of the models, and the conclusions are drawn accordingly.

4.3.5 Model Parameters

Two classification models were used for the experiments, an SVM model, and an MLP model. SVM was a more traditional model, and the MLP was a stand in for Deep Learning models. The parameters for the SVM model are {`kernel:rbf, C:10, gamma:scale, random_state:8`}. $C = 10$ was configured based on hyperparameter tuning. The MLP contains two hidden layers of sizes 512 & 128 for the first and second layers respectively, the rest having the default values. The MLP was implemented

using the Keras library, and the random state was set to 8 for reproducibility. The parameters are kept constant, regardless of the embedding being experimented on, because our objective is to determine whether changing the embedding alone can produce a significant change in the performance measure. If true, that change in the performance measure can be directly attributed to the change in embedding. Significance testing further confirms whether the change was statistically significant or just a fluke of random sampling. Each of the models will be subsequently referred to by the name of the embedding used to train the model. For instance, the SVM model trained on GTE will be simply referred to as GTE.

4.4 Results and Discussion



(a) SVM Configurations (b) MLP Configurations

Figure 4.1: 5x2-fold CV Results for SVM and MLP

Figure 4.1 shows radar charts with the average of the 10 test results from the 5x2 CV experiment for different embeddings, for SVM and MLP models respectively. Tables 4.1b & 4.1a provide the relevant significance testing results to determine whether the differences in weighted F1-score performance metrics in Figure 4.1 are statistically significant.

Following is the analysis of the models trained on embeddings created through TL, as detailed in Section 4.3.3. The most relevant and interesting results are discussed to appraise the overarching goal of research towards COVID-19 misinformation detection.

Model 1	**Model 2**	***f*-statistic**	**p-value**	**Significant?**
GTE	ATE	81.785	0.000066	Yes
CSE	ATE	890.184	$< .00001$	Yes
GTE	GTE+ATE	1.345	0.391441	No
ATE	GTE+ATE	64.005	0.000121	Yes
GTE	CSE	3.105	0.111458	No
GTE	GTE+CSE	16.561	0.00317	Yes
CSE	GTE+CSE	3.953	0.071234	No
GTE+ATE	GTE+CSE	18.385	0.00248	Yes

(a) SVM Results

Model 1	**Model 2**	***f*-statistic**	**p-value**	**Significant?**
GTE	ATE	114.177	0.000029	Yes
CSE	ATE	61.369	0.000135	Yes
GTE	GTE+ATE	1.099	0.488222	No
ATE	GTE+ATE	11.006	0.008146	Yes
GTE	CSE	2.055	0.220775	No
GTE	GTE+CSE	7.826	0.017418	Yes
CSE	GTE+CSE	8.107	0.016123	Yes
GTE+CSE	GTE+ATE	2.593	0.152293	No

(b) MLP Results

Table 4.1: Results of 5x2-Fold CV for 2-Tailed Significance Testing with SVM and MLP Variants

4.4.1 ATE

Figure 4.1 shows that ATE based models perform worse that every other model for MLP & SVM, including models based on embeddings used to create ATE, namely GTE and CSE. Furthermore, Tables 4.1a & 4.1b shows that the performance difference is significant.

4.4.2 GTE+ATE

For SVM, Figure 4.1 shows that GTE+ATE performs as well as GTE, but it performs far better than ATE. Furthermore, Table 4.1a shows that the improvement over ATE is significant, but there is no significant difference between performances of GTE and GTE+ATE. It can be surmised that the improvement in the performance of GTE+ATE over ATE could be largely attributed to GTE. For MLP, GTE+ATE also performs far better than ATE, and as well as GTE. GTE+ATE's weighted F1-score is only 1% lesser than GTE's. While Table 4.1b shows that GTE+ATE's performance is significantly better than ATE's, there is no significant difference between GTE's and GTE+ATE's performances. It is possible that the MLP was not able to take advantage of the concatenation of GTE and ATE. Again, improvement in GTE+ATE's performance can be attributed to GTE.

4.4.3 GTE+CSE

It can be clearly seen from Figure 4.1 that models based on GTE+CSE provide the best performance for both SVM and MLP across all the performance metrics. However, for SVM, Table 4.1a shows that while GTE+CSE performs significantly better than GTE, it does not show significant improvement over CSE. It can be surmised that for SVM, GTE+CSE's performance improvement is due to CSE. But, GTE+CSE has a vocabulary of size 3M, far larger than CSE's 22k, thereby lending GTE+CSE more generalizability because it can represent more words. Hence, GTE+CSE stands a better chance at outperforming CSE on datasets whose vocabulary may be very dif-

ferent from CSE. For MLP, Table 4.1b clearly shows that GTE+CSE's performance is significantly better than that of both GTE and CSE individually. While Figure 4.1 shows that on average GTE+CSE is better than GTE+ATE, but Tables 4.1b and 4.1a show that GTE+CSE performs significantly better than GTE+ATE for MLP, but the same performance is not observed for SVM. Ultimately, concatenation of general and context-specific embeddings significantly improves performance as shown by our analysis of the F1 score and the statistical significance tests. Such tests have not been carried out by comparable studies. This appraisal shows promise on using CTL for future research on health misinformation detection.

4.5 Conclusion

This chapter tackled the issue of health misinformation detection, and explored combinations of different supervised classification models trained on different general and domain-specific embeddings. Furthermore, analysis of the Cross Validation results was presented to determine whether the differences in weighted F1-score performance metrics were statistically significant. Ultimately, the concatenation approach of general and context-specific embeddings showed statistically significant improvement in performance. For future work, the generalizability of Concatenation Transfer Learning with other datasets could be explored. Experiments using BERT and RoBERTA could be conducted in future works to further evaluate the findings in this chapter, given that transformers have been heavily utilized in recent research on similar NLP tasks.

Chapter 5

UofA-Truth at Factify 2022 : Transformer And Transfer Learning Based Multi-Modal Fact-Checking

5.1 Introduction

Humankind has dealt with misinformation since time immemorial [63]. However, never in human history have people had access to the amount of information that they have today. The Internet is the primary reason for the easy access to this information. It has given people the ability to access information from all over the world and from innumerable sources. However, this deluge of information has brought with it the problem of misinformation/disinformation/fake news. Never before have we had more efficacious means to disseminate deceptive fallacies, falsehood that is unfortunately believed and is wrongfully, and sometimes dangerously impacting people.

While there are many definitions of Fake News, for the purposes of this paper Fake News can be defined as a news piece, social media post, etc., which contains claim(s) that can be refuted by information put out by "reputable organizations". Such organizations may include, but are not limited to, government bodies, news outlets which score high on Media Bias/Fact Check's Factual Reporting scale [64] or professional fact-checking organizations which are verified signatories of the International Fact-

Checking Network (IFCN) code of conduct [65] [66]. This definition of fake news off-loads the responsibility of determining what exactly fake news is, on to expert fact-checkers or domain experts, and allows Artificial Intelligence (AI) to deal with the more manageable problem of determining whether claim(s) made in a news piece is entailed, not entailed or refuted by a corresponding news piece from a reputable source.

Fake news can cause real world harm as is being seen during the COVID-19 pandemic: misinformation has led to vaccine hesitancy, which is directly tied to increased chances of mortality due to COVID-19 [67]. Fact-checking or determining whether a news piece contains fake claims is the first step in countering such fake news. Furthermore, it is not only important to detect and counteract fake news, but to do so in a timely manner. Given the large amount of information generated on social media sites every-day and the time constraints that online fact-checking operates under, it is imperative that automated methods of misinformation detection are developed to aid in the manual fact-checking of fake news.

In general, the information generated and distributed on the Internet is multi-modal, i.e., consisting of text, images, audio-visual, etc. Often times information is conveyed via a combination of two or more modes, for instance, memes, pieces of information rapidly spread among users, are often a combination of text and image/short video, where text is overlaid on the image or short video (also called a gif). Thus, an automated method should be able to take advantage of all the modes of information available to fact-check a claim.

The shared task FACTIFY, in conjunction with the AAAI conference, attempts to aid in the development of automated multi-modal fact-checking by introducing a dataset which consists of multi-modal claims and corresponding supplementary information or documents, using which said claims need to be fact-checked [70]. Each data-point in the dataset contains a "claim"/Un-Verified text that consists of a short sentence or phrase, and a "claim"/Un-Verified image associated image, which may

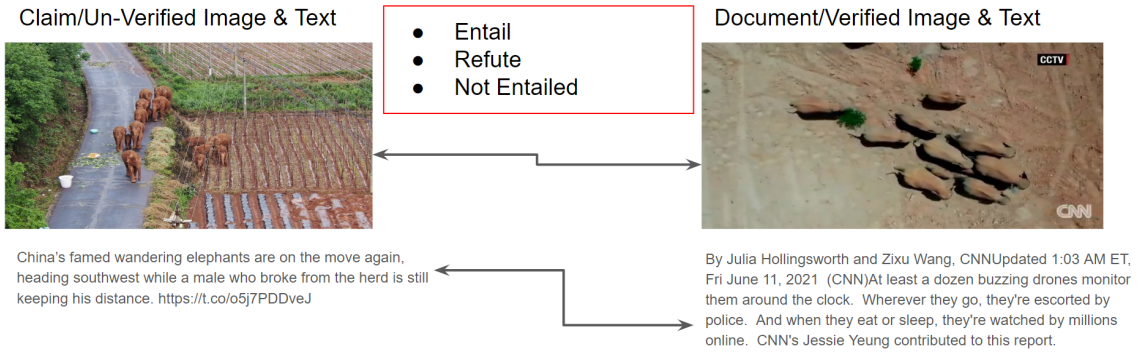


Figure 5.1: A single data-point in the Factify Shared Task Dataset. The Claim and Document Image & Text pairs were retrieved from [68] and [69] respectively.

or may not have overlaid text. An Example of the Un-Verified Image and Text pair to be fact-checked is provided on the left side of Figure 5.1. Similarly, the corresponding supplementary information or "document" similarly consists of a text and an image component as shown in the right side of Figure 5.1. As shown in the red bordered box in Figure 5.1 the relation between the claim/un-verified text and document/verified text is labeled as "Entail", "Refute" or "Not-Entailed" depending on whether the document text supports, does not support, or is unrelated to the document image, respectively. The relation between claim and document image pair is similarly labelled. Thus, the task is to create a model that can determine whether the claim text and image are individually entailed, not entailed, or refuted by the corresponding document text and image pair. Since there are two labels for each datapoint - one for claim and document image pair and the other for claim and document text pair, the task organizers have defined five possible labels for each datapoint, depending on the labels of the relation of claim and document image and text pairs. They are as follows:

- Support_Multimodal: Both claim text and image are entailed by document text and image respectively
- Support_text: Claim text is entailed, but the claim image is not entailed by their document counterparts

- Insufficient_Multimodal: Claim text is not entailed, but claim image is entailed by their document counterparts
- Insufficient_Text: Neither claim text not claim image is entailed by their document counterparts
- Refute: Both claim text and image are refuted by their document counterparts

Our team "UofA-Truth" participated in the shared task and secured the 4^{th} position with a weighted F1-score of 74.807%, just ≈ 2 F1 points behind the top submission. In this chapter we shall describe our simple yet effective automated fact-checking model.

5.2 Related Works

The dataset used to train and test our model was released under the shared task FACTIFY, which is a part of the workshop De-Factify at the AAAI 2022 conference [70]. The dataset consists of a total of $50,000$ claim and document pairs, which are divided into train, validation and test sets of sizes $35,000(70\%)$, $7,500(15\%)$ and $7,500(15\%)$ respectively.

The entailment aspect of the shared task is similar to "Stance Detection", which can be defined as the classification of the stance of the producer of a news piece with respect to an unverified claim [71]. In the context of the shared task, the unverified claim is the claim text and image pair, and the news piece is the document text and image pair.

Stance Detection is an important part of Fake News detection and was notably used in the Fake News Challenge - 1 (FNC-1) [72]. This challenge was similar to the FACTIFY shared task, except FNC-1 only dealt with text entailment or stance detection, unlike FACTIFY which deals with multi-modal entailment. FNC-1 introduced a dataset which consisted of a headline and a body of text, which may be from the

same article or different articles. Depending on the stance of the body of text with respect to the headline, the text-headline pairs were to be classified into any of the following classes:

- Agrees: The body of text agrees with the claim(s) made in the headline
- Disagrees: The body of text disagrees with the claim(s) made in the headline
- Discusses: The body of text and headline are referring to the same subject, but the body does not take any stance or position on the claim(s) made in the headline
- Unrelated: The body of text is not related to the claim(s) being made in the headline

FACTIFY's not-entail class can be considered similar to a combination of Unrelated and Discusses classes of FNC-1, while entails and refutes classes can be considered similar to FNC-1's Agrees and Disagrees classes respectively.

This similarity between the two tasks led us to draw inspiration from the UCL Machine Reading team's submission to the FNC-1's challenge, which performed 3^{rd} best among the 50 submissions to the challenge [73]. In their submission the UCL team, Riedel, Augenstein, Spithourakis, and Riedel, describe their approach as a "simple but tough-to-beat baseline" for stance detection. As explained above, there are two inputs for this task - a headline and a body of text. Riedel, Augenstein, Spithourakis, and Riedel calculated the Term Frequency (TF) vectors and Term Frequency - Inverse Document Frequency (TF-IDF) for both the headline and the body of text on the basis of the $5,000$ most frequent words. The TF vectors of the two inputs are concatenated with result of the cosine similarity of between the TF-IDF vectors of the headline and body, as shown in Figure 5.2. The resultant vector of length $10,001$ is then fed as input into a shallow Multi Layer Perceptron (MLP) network, which has a softmax output of length four, one for each class in the FNC-1 task.

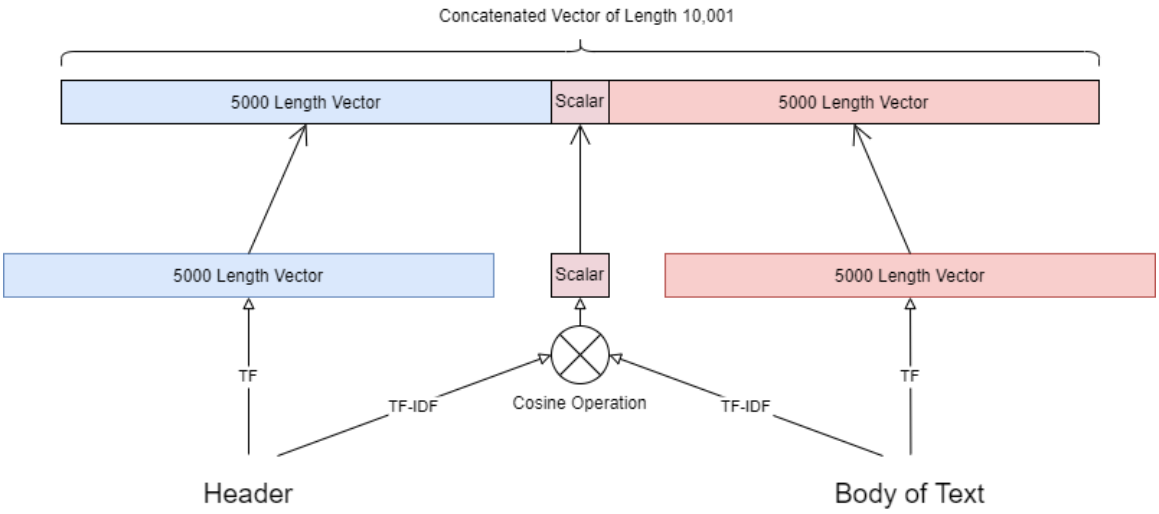


Figure 5.2: Concatenated Vector Representation. Adapted from "A simple but tough-to-beat baseline for the Fake News Challenge stance detection task", by Riedel, Augenstein, Spithourakis, and Riedel

Given the similarity of the tasks being solved in FNC-1 and FACTIFY, we adopted the manner of concatenation of the cosine similarity and vector representations of the header and body as explained in [73]. Instead of using TF for vector representations and TF-IDF for cosine similarity calculation, we used Sentence-BERT in lieu of both TF and TF-IDF vectors to determine entailment between the claim text and document text [74]. Sentence BERT is a BERT based model which has been specifically fine tuned for Natural Language Inference (NLI) task and has proven to be better at capturing features of a sentence that are relevant to the inference task. To determine entailment between claim and document images, we used a pre-trained instantiation of the Xception architecture [75] available in Keras [76], which had been trained on JFT-300M dataset [77].

The FNC-1 challenge had two other submissions which performed better than [73], however, we concluded that those were more complicated architectures and might hamper the scalability and time complexity of our model. For instance, Pan, Sibley, and Baird, who submitted the winning model [78], had an ensemble model which consisted of a deep learning model and a tree based ensemble model as implemented

Table 5.1: FACTIFY Task Labels & Corresponding Text and Image Entailment Labels

FACTIFY Task Label	Text Entailment Label	Image Entailment Label
Support_Multimodal	$\mathcal{T}$_0	$\mathcal{I}$_0
Support_text	$\mathcal{T}$_0	$\mathcal{I}$_1
Insufficient_Multimodal	$\mathcal{T}$_1	$\mathcal{I}$_0
Insufficient_Text	$\mathcal{T}$_1	$\mathcal{I}$_1
Refute	$\mathcal{T}$_2	$\mathcal{I}$_2

in Xgboost [79]. The outputs of the two models were weighted equally to produce the final predictions. Hanselowski, PVS, Schiller, and Caspelherr also implemented an ensemble model which consisted of five Neural Network models. The final prediction was made through majority voting. Despite the increased complexity, Team UCL Machine Reading's model performance was within $\approx$ 1 point of the top two submissions.

5.3 Methodology

The FACTIFY shared task's classes (Support_Multimodal, Support_text, Insufficient_-Multimodal, Insufficient_Text, Refute) are composed of a combination of text and image entailment classes. For instance, if text entailment, non-entailment and refutation are represented by $\mathcal{T}$_0, $\mathcal{T}$_1, $\mathcal{T}$_2, and image entailment, non-entailment and refutation are represented by $\mathcal{I}$_0, $\mathcal{I}$_1, $\mathcal{I}$_2 respectively; then the shared task's classes can be reformulated as as a combination of text and image entailment labels as shown in Table 5.1. It is important to note here that all combinations of text entailment labels and image entailment labels are not present in Table 5.1. For instance, combinations such as $\mathcal{T}$_0 & $\mathcal{I}$_2 do not exist. The lacking combinations are treated when consolidating the labels after classification. This is explained in Section 5.3.4.

It can be clearly seen that the shared task can now be broken down into two sub-

Table 5.2: Text Entailment Task Labels in Terms of Original FACTIFY Task Labels

Text Entailment Label	FACTIFY Labels
$\mathcal{T}$_0	Support_Multimodal & Support_text
$\mathcal{T}$_1	Insufficient_Multimodal & Insufficient_Text
$\mathcal{T}$_2	Refute

Table 5.3: Image Entailment Task Labels in Terms of Original FACTIFY Task Labels

Image Entailment Label	FACTIFY Labels
$\mathcal{I}$_0	Support_Multimodal & Insufficient_Multimodal
$\mathcal{I}$_1	Support_text & Insufficient_Text
$\mathcal{I}$_2	Refute

tasks; namely, text entailment and image entailment, where text entailment consists of classes $\mathcal{T}$_0, $\mathcal{T}$_1 and $\mathcal{T}$_2, and image entailment consists of classes $\mathcal{I}$_0, $\mathcal{I}$_1, $\mathcal{I}$_2. These new classes are the combination of the original class labels as shown in Table 5.2 and Table 5.3 for the text entailment and image entailment tasks respectively. Once the dataset is rearranged according to the sub-task labels, we end up with one dataset for each sub-task.

We now define Text Entailment as a task of predicting the document text's stance towards the claim text, and Image Entailment as a task of prediction the document image's stance towards the claim image.

5.3.1 Preprocessing

Image preprocessing is done by resizing all the images to (256, 256, 3) size with bilinear interpolation as implemented in image_dataset_from_directory in Keras [76]. Thereafter, all the pixel values are scaled to a a range of 0 to 1.

Text preprocessing involves removing urls from all claim and document texts with the help of the Preprocessor library [81].

5.3.2 Vector Representations

The preprocessed inputs (text and images) need to be converted into vector representations so that they can be presented as input for a classifier.

The preprocessed images are converted into vectors of size 2048 each, by using the pre-trained Xception model in Keras [76]. This can be achieved by setting include_top attribute to False and pooling attribute to 'avg'. Setting include_top to False removes the fully connected layer at the end of the model and exposes the output of the second to last layer. Setting pooling to 'avg' ensures that a global pooling average is applied to the 3D output of the second last layer of Xception, to convert it into a 1D output. Since the Xception model has been trained on a massive dataset for a general image classification task, it can be reasonably assumed that the output of the second to last layer captures information which may be useful for downstream tasks such as image entailment.

The preprocessed texts are converted into a vectors of length 384 each by using the pre-trained Sentence-BERT model [74].

The cosine similarity of the vector representations of claim and corresponding document images is calculated and concatenated in the manner shown in Figure 5.3. This creates a concatenated representation for each claim and corresponding document image of size 4097. Similarly, the concatenated representation of claim and corresponding document text of size 769 is created through the same procedure of cosine similarity calculation and subsequent concatenation as shown in Figure 5.4.

5.3.3 Classifiers

The vector representations are now used for training the classifiers for the image and text entailment tasks. Different classifiers are used for the image and text entailment tasks.

As shown in Figure 5.3 the image entailment classifier consists of a single fully connected hidden MLP layer of 5000 units, ReLU activation with a dropout probability

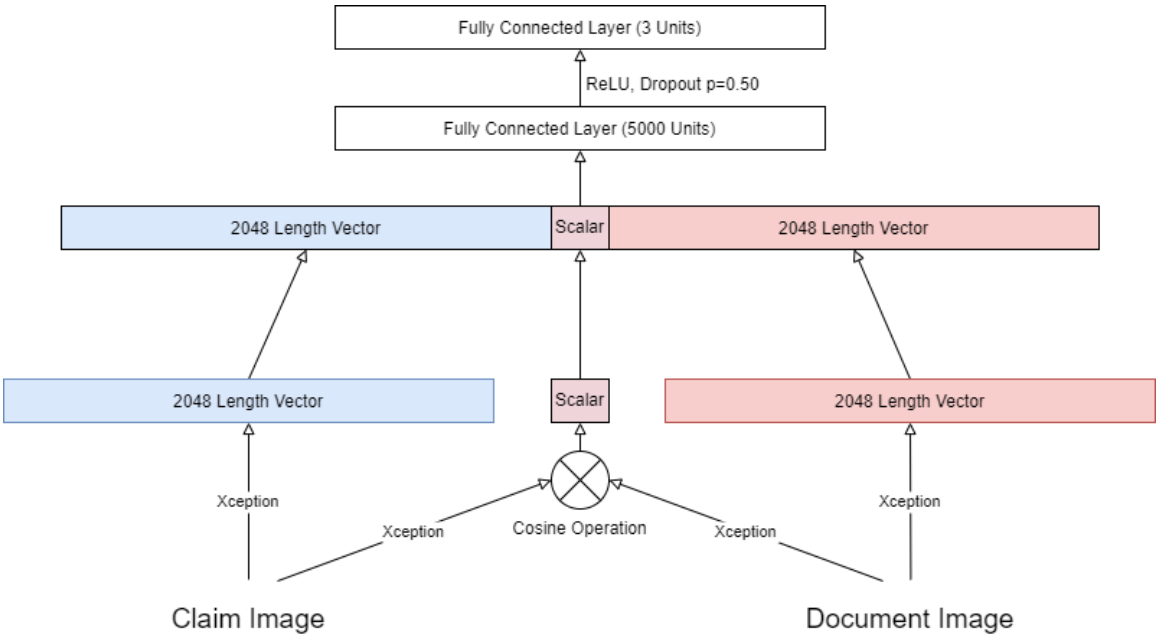


Figure 5.3: Image Entailment Classifier Architecture

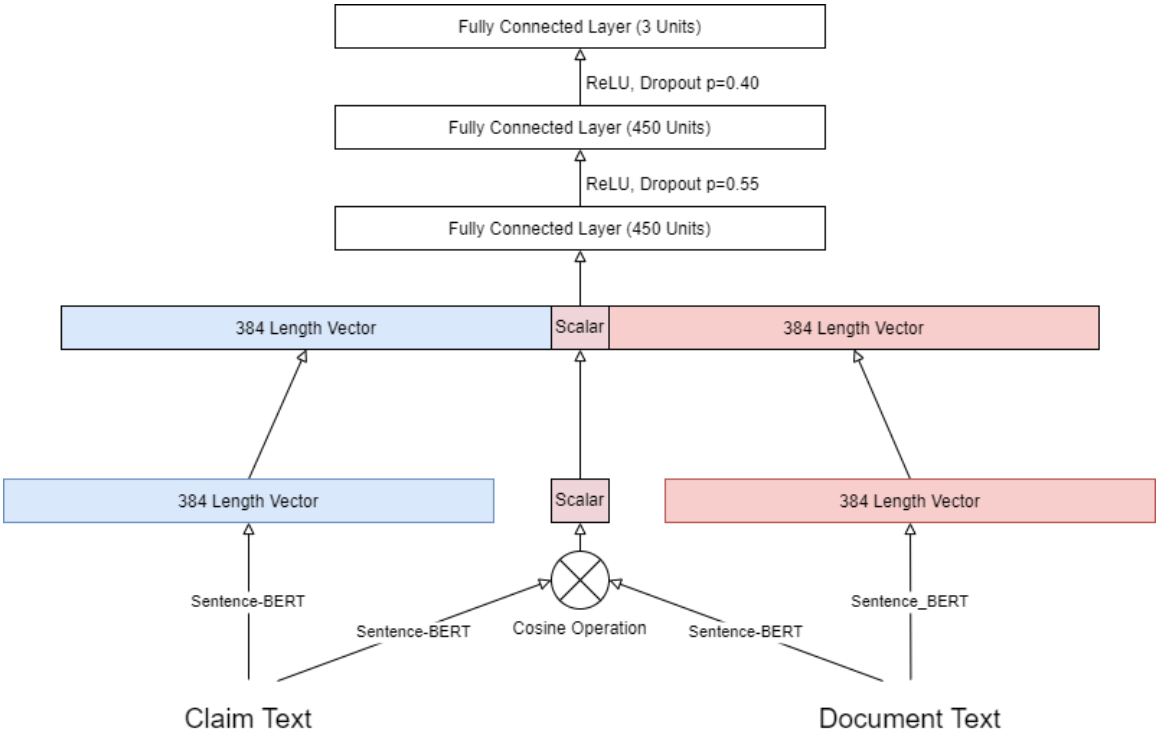


Figure 5.4: Text Entailment Classifier Architecture

of 0.5. The output of this layer then feeds into a fully connected output layer of 3 units, one for each class label (entailment, non-entailment and refute), and a sigmoid activation function.

On the other hand, as shown in Figure 5.4 the text entailment classifier consists of two fully connected layers of 450 units each, ReLU activation functions, $l2$ activity regularizers, and a dropout probability of 0.55 for the first layer and 0.4 for the second layer. The output of the two hidden layers then feeds into the fully connected output layer 3 units with sigmoid activation.

The Cross Entropy loss is calculated after performing the softmax operation on the outputs of both the classifiers.

5.3.4 Label Consolidation

The output of the image classifier classifies every pair of claims and document image into one of the three labels $\mathcal{I}\_0$, $\mathcal{I}\_1$ and $\mathcal{I}\_2$. Similarly, for claim and document text pairs.

The pairs of image and text entailment labels belonging to the same data-point are combined and then converted into the original FACTIFY task labels (namely, Support_Multimodal, Support_text, Insufficient_Multimodal, Insufficient_Text, Refute) according to Table 5.1.

However, it is possible that the combination procedure may produce pairs of entailment labels which do not have any corresponding FACTIFY task label. For instance, ($\mathcal{T}\_0$, $\mathcal{I}\_2$), ($\mathcal{T}\_1$, $\mathcal{I}\_2$), ($\mathcal{T}\_2$, $\mathcal{I}\_0$), ($\mathcal{T}\_2$, $\mathcal{I}\_1$), are four such invalid pairs of labels. We thus have to change such label pairs into valid label pairs. We do so by using a heuristic as described in Table 5.4(A). If one of claim text or claim image is entailed, i.e., $\mathcal{T}\_0$ or $\mathcal{I}\_0$, it is unlikely that the other claim mode will be refuted by the document, hence, the latter's label needs to be changed to not-entailed, i.e., $\mathcal{T}\_1$ or $\mathcal{I}\_1$. If however, one of the claim text or image is refuted by the corresponding document then it is unlikely that the other claim mode will have uncertain entailment, hence

the latter's label should be converted to refuted as well, i.e., $\mathcal{T}\_2$ or $\mathcal{I}\_2$. Thereafter, we can calculate the final weighted F1 accuracy on the Test set.

Table 5.4: Heuristics for invalid label conversion

Invalid Label Pair	Valid Label Pair
($\mathcal{T}\_0$, $\mathcal{I}\_2$)	($\mathcal{T}\_0$, $\mathcal{I}\_1$)
($\mathcal{T}\_1$, $\mathcal{I}\_2$)	($\mathcal{T}\_2$, $\mathcal{I}\_2$)
($\mathcal{T}\_2$, $\mathcal{I}\_0$)	($\mathcal{T}\_2$, $\mathcal{I}\_2$)
($\mathcal{T}\_2$, $\mathcal{I}\_1$)	($\mathcal{T}\_1$, $\mathcal{I}\_0$)

A: Invalid to Valid Label Pair Conversion

Invalid Label Pair	Valid Label Pair
($\mathcal{T}\_0$, $\mathcal{I}\_2$)	($\mathcal{T}\_0$, $\mathcal{I}\_0$)
($\mathcal{T}\_1$, $\mathcal{I}\_2$)	($\mathcal{T}\_1$, $\mathcal{I}\_1$)
($\mathcal{T}\_2$, $\mathcal{I}\_0$)	($\mathcal{T}\_2$, $\mathcal{I}\_2$)
($\mathcal{T}\_2$, $\mathcal{I}\_1$)	($\mathcal{T}\_2$, $\mathcal{I}\_2$)

B: New Invalid to Valid Label Pair Conversion

5.4 Results & Discussion

Our team, UofA-Truth, secured the 4^{th} position on the leaderboard, with an F1-score of 74.807% on the final evaluation. However, the confusion matrix, shown in Figure 5.5, reveals more fine grained details about our model's performance on the test set.

The model performed worst on the Insufficient_Multimodal category. It can be clearly seen from the matrix that a large number (304) of data-points with ground truth Insufficient_Multimodal were incorrectly classified as Support_Multimodal. Since the only difference between the two classes is text entailment, it is possible that the model is unable to differentiate between text entailment and non-entailment. This may be because the claim and document texts might have common words or might even talk about tangential or similar topics, but do not reach the threshold of text entailment.

Furthermore, the model did not perform well on the Support_text class. A significant number (202) of data-points belonging to Support_Text were mis-classified as Support_Multimodal. Again, given that the only difference between the two classes

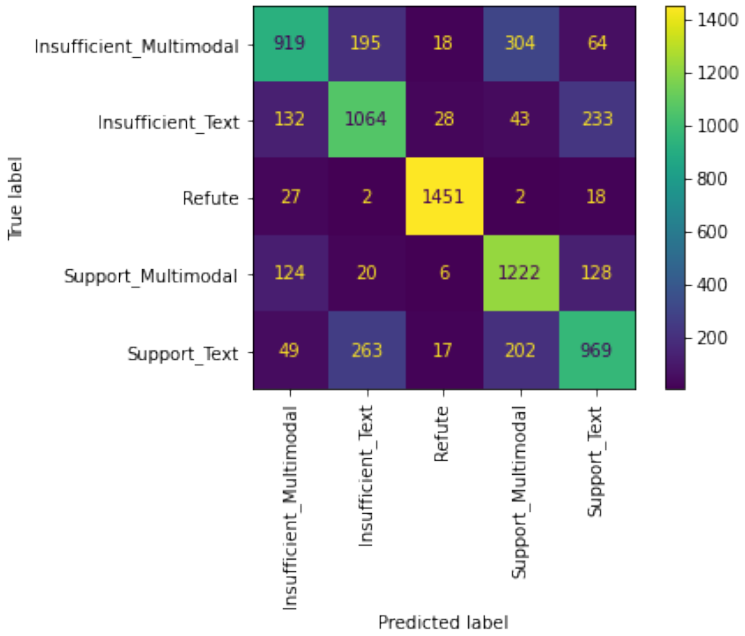


Figure 5.5: Confusion Matrix for Original Heuristic

is in image entailment, it follows that a model would find it hard to differentiate between the two. Similarly, for the Support_Text and Insufficient_Text.

The model performs best on the Refute class despite the fact that the said class had the fewest data-points in the training set. Very few of its data-points are misclassified as other classes, and vice-versa. This may be because a significant number of the data-points belonging to the Refute class have been take from fact-checking websites. This fact may set such data-points apart from other claim-document pairs. For instance, document images and corresponding claim images of the Refute class tend to be identical because fact-checking websites almost always provide a screenshot of the fake news/social media posts they debunk in their articles. They may even overlay images of news pieces they fact-check with a digital stamp, indicating their logo or whether the news piece was true or fake. They usually clearly state the gist of the fake news they debunk, at the beginning of every article, often times quoting said fake news verbatim. Such peculiarities may make data-points belonging to the Refute class easy to discern.

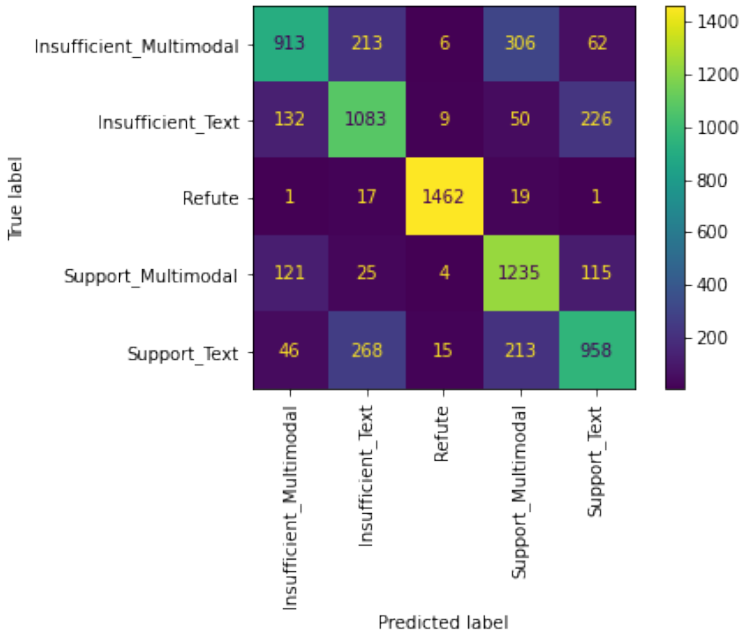


Figure 5.6: Confusion Matrix for Modified Heuristic

Heuristics mentioned in Section 5.3.4 can be changed to improve the weighted F1-score on the test set. It is possible that the image entailment model merely learns to determine similarity between claim and document image pairs. Thus, it may be better to have a heuristic which changes the invalid label pairs into valid label pairs by changing the image entailment label to be the same as the text entailment label. Therefore, after the competition results we changed the heuristic for invalid label pair to valid label pair conversion as per the new heuristics shown in Table 5.4(B). These modified heuristics improve the final F1-score from 74.807% to 75.183%. As can be seen by the confusion matrix in Figure 5.6, the new heuristic reduces the classification accuracies of the Insufficient_Multimodal and Support_Text classes, for the benefit of the other classes. Other than that, the overall dynamics remain the same as in Figure 5.5. It could be possible to continue adjusting these heuristics to obtain even better results but have not experimented further.

5.5 Conclusion

In this chapter, we introduced a simple, yet effective method of multi-modal fake news detection. We divided the main task into two sub-tasks; namely, text entailment and image entailment. Thereafter, we used pre-trained Xception network and Sentence-BERT to get vector representations of images and text respectively. We then used these vector representations for classifications tasks of image and text entailment by adapting the approach introduced by Riedel, Augenstein, Spithourakis, and Riedel in their submission to the FNC-1 task. Finally, we consolidated the prediction of the two sub-tasks of image and text entailment to get the final predictions. We used the model thus created to make predictions on the test set, and our team's submission achieved the 4^{th} position on the leader board with a 74.807% weighted F1-score.

Chapter 6

Conclusion and Future Works

In this Thesis, we presented various approaches to deal with fake news detection.

In Chapter 3 we introduced post-hoc explainable models for detection of articles containing false claims related to Neurodevelopmental Disorders. We also introduced an annotated dataset to test said models. The test results were presented and analyzed. It was found that matching the sentiment of the unverified article and the relevant medical literature abstract yielded better results than using sentence pair inference models. Furthermore, comparing the extractive summary of the unverified article, rather than every line that contains medical entities, to relevant medical literature yields better results. Significantly, the process of medical textual claims verification incorporated in our models is very similar to the way in which medical professionals search for evidence for various interventions, medicines and therapies in a clinical setting. This allows any medical practitioners, caregivers to "look into" the workings of our models in a way that purely neural network based models don't. Specifically, any human can check the sentences that our models select for fact-checking from the unverified article, the query formulated on the basis of said sentence selection, the query results from Pubmed, and finally the sentiment matching or sentence pair inference based determination of agreement or disagreement between the unverified article and the retrieved medical literature.

In Chapter 4 we explored variations and concatenations of different Word2Vec

embeddings to build models on top of and determine if the different embedding concatenations lead to statistically significant performance differences. This was in pursuit of detection of tweets containing false claims related to COVID-19. The results showed that concatenation of embeddings trained on context specific data and general language modelling data leads to statistically significant improvements in overall performance of the model over models based on one of the two embeddings.

In Chapter 5, we presented the result of our participation in the Multi-Modal Fact Checking competition in the De-Factify workshop at AAAI, 2022. Our team secured the 4^{th} position on the leader board. The task involved determining the veracity labels of claims represented by text and image pairs by using the corresponding text and image pair "evidence" provided by the organizers. My team's model separately determined the veracity of the text and image components by concatenating the text and image components' vector representations with those of their corresponding text and image "evidence" and their cosine similarities.

The contributions detailed in the three aforementioned chapters tackle different ways of automated fake news detection. This is important because manual fact-checking of large number of articles and social media posts that spread fake news is time and cost inefficient. For instance, using our approach in Chapter 5 text and images in articles can be used to fact-check against articles verified by professional fact-checkers. This will allow fact-checkers to manually debunk other unique false claims rather than fact-checking semantic variations of claims they have already fact-checked. Our contribution in Chapter 3 can be used as a tool to assist parents and caregivers to filter out quack medications and therapies from their search results, or by medical professionals and researchers to conduct systematic reviews.

For future work, the textual inference model in Chapter 5 can be used in place of the inference models used in Chapter 3 to determine the agreement and disagreement between unverified articles and medical literature. Furthermore, the work in Chapter 3 could be extended by incorporating image and videos available on webpages into

the fact-checking pipeline.

Further work can be conducted in converting the explainable models in Chapter 3 into end to end trainable models. As of now, the post-hoc explainable models are made up of a pipeline of pre-trained or unsupervised models that have been strung together to give the final veracity rating. An end-to-end trainable model would be one where all the constituent pre-trained models' parameters could be trained on a context specific dataset - the context being NDD related.

The models presented in Chapter 3 could be used together to create a useful tool for fact-checking. For instance, we have already pointed out that line-by-line fact-checking may mislead the classifier since true articles/webpages debunking false claims state those claims verbatim before proceeding to debunk them. Thus, a tool could be created which takes advantage of the Yake Query+Sentiment Matching to determine the overall veracity of the article, and then use the more granular, Stanza Query+Sentiment model to fact-check claims sentence-by-sentence.

To achieve an end-to-end trainable model, the dataset size needs to be expanded. Our NDD specific dataset stands at 116 articles which is too small to train on. However, engaging subject matter experts in the actual annotation is difficult due to the experts' time constraints. One solution could be to use experts to annotate a set of unique single sentence statements, as was done in Chapter 3, and using platforms like Amazon Mechanical Turk [1] to annotate articles related to expert annotated sentences. However, there may be concerns about the quality of annotations emerging from Mechanical Turk due to the specialized nature of medicine. This was one of the reasons why we chose to enlist the help of Neuroscience students instead on using Mechanical Turk. Heuristics could also be used to annotate datasets. For instance, a dataset of webpages/articles from reliable and unreliable websites about medical news can be collected. The reliability of these sites may be determined from Media

[1] https://www.mturk.com/

Bias Fact Check [2]. The retrieved webpages could be labelled as reliable or unreliable depending on the reliability rating of the website they were extracted from. This dataset could be used to train the aforementioned end-to-end trainable model.

There are other architectural changes which can fix some limitations in the model. One of them is to use pronoun resolution models to replace pronouns with the nouns that they refer to. This would aid in better detection of relevant keywords by both YAKE! and Stanza models. The other limitation is that the model does not take into consideration the reputation of the venue where the medical article or paper, retrieved from MEDLINE by PubMed, was published. The venue reputation can be gauged through the Impact Factor, and the same could be incorporated into the fact-checking models.

In our view, a minimum viable automated fact-checker would be one with high recall of fake news and high precision of Real news because it is more dangerous to mis-classify false claims or fake news as true news and suggest them to parents and caregivers through the CAMI chatbot than classify real news as fake and not suggest it to the users of CAMI. Furthermore, we want to make sure that the articles classified by the model to be true are actually true, thus requiring high precision for real news detection. Hence, we selected Stanza Query+Sentiment Matching model as the best model according to its performance on the test set. Stanza Query+Sentiment Matching has 0.868 recall for Fake Article detection and 0.706 recall for True Article detection - the highest out of the models we tested on the NDD specific dataset. However, as can be seen by the performance metrics, there is still much progress to be made in creating that minimum viable product.

On the other hand, the ideal automated fact-checker would be one which has high recall for true news, in addition to the performance characteristics of the aforementioned minimum viable model, i.e., high precision and recall for fake news and true news respectively. This is because we would ideally like to have as large a database as

[2]https://mediabiasfactcheck.com/

possible of authoritative, fact-checked news sources, in order to cater to all possible needs of the CAMI chatbot's users, i.e., caregivers, medical professionals, etc.

Finally, the multi-modal fact-checking task presented in the De-Factify workshop has not yet been solved since the highest score achieved on the dataset was 0.77(rounded up) weighted F1-score. There is clearly scope for improvement. For instance, our model presented in Chapter 5 freezes the weights of the Xception and the Sentence Transformer models, and only uses the vectors produced by these networks for further learning. Thus, the performance of the model could probably be improved by fine-tuning the aforementioned pre-trained model on the textual and image inference. Furthermore, the textual and image inference models could be combined into a single model which learns to predict the final 5 classes instead of breaking up the task into the textual and image inference constituents, and then combining the resultant labels to predict the final label from among the 5 classes.

As presented in the conclusion, there is a lot of scope for improvement in the tasks related to fact-checking and fake news detection. In this section we summarized the various contributions made in different chapter and the possible directions future work could take.

Bibliography

[1] N. Farruque, O. Zaiane, and R. Goebel, "Augmenting Semantic Representation of Depressive Language: From Forums to Microblogs," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, Springer, 2019, pp. 359–375.

[2] P. Patwa, S. Sharma, S. PYKL, V. Guptha, G. Kumari, M. S. Akhtar, A. Ekbal, A. Das, and T. Chakraborty, *Fighting an Infodemic: COVID-19 Fake News Dataset*, 2020. arXiv: 2011.03327 [cs.CL].

[3] S. Vosoughi, D. Roy, and S. Aral, "The spread of true and false news online," *Science*, vol. 359, no. 6380, pp. 1146–1151, 2018.

[4] P. M. Waszak, W. Kasprzycka-Waszak, and A. Kubanek, "The spread of medical fake news in social media – the pilot quantitative study," *Health Policy and Technology*, vol. 7, no. 2, pp. 115–118, 2018, ISSN: 2211-8837. DOI: https://doi.org/10.1016/j.hlpt.2018.03.002. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2211883718300881.

[5] K. Shu, D. Mahudeswaran, and H. Liu, "Fakenewstracker: A tool for fake news collection, detection, and visualization," *Computational and Mathematical Organization Theory*, vol. 25, no. 1, pp. 60–71, 2019.

[6] V. Klema and A. Laub, "The singular value decomposition: Its computation and some applications," *IEEE Transactions on Automatic Control*, vol. 25, no. 2, pp. 164–176, 1980. DOI: 10.1109/TAC.1980.1102314.

[7] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*, ser. ICML'14, Beijing, China: JMLR.org, 2014, II–1188–II–1196.

[8] P. Atanasova, P. Nakov, L. Màrquez, A. Barrón-Cedeño, G. Karadzhov, T. Mihaylova, M. Mohtarami, and J. Glass, "Automatic fact-checking using context and discourse information," *Journal of Data and Information Quality (JDIQ)*, vol. 11, no. 3, pp. 1–27, 2019.

[9] P. Gencheva, P. Nakov, L. Màrquez, A. Barrón-Cedeño, and I. Koychev, "A context-aware approach for detecting worth-checking claims in political debates," in *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, 2017, pp. 267–276.

[10] T. Mikolov, Q. V. Le, and I. Sutskever, "Exploiting similarities among languages for machine translation," *arXiv preprint arXiv:1309.4168*, 2013.

[11] E. Dai, Y. Sun, and S. Wang, "Ginger cannot cure cancer: Battling fake health news with a comprehensive data repository," in *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 14, 2020, pp. 853–862.

[12] M. Alsyouf, P. Stokes, D. Hur, A. Amasyali, H. Ruckle, and B. Hu, "Fake news in urology: Evaluating the accuracy of articles shared on social media in genitourinary malignancies," *BJU international*, vol. 124, no. 4, pp. 701–706, 2019.

[13] Á. Iglesias-Puzas, A. Conde-Taboada, B. Aranegui-Arteaga, and E. López-Bran, ""'fake news" in dermatology. results from an observational, cross-sectional study," *International Journal of Dermatology*, vol. 60, no. 3, pp. 358–362, 2021.

[14] T. Brants, A. C. Popat, P. Xu, F. J. Och, and J. Dean, "Large language models in machine translation," 2007.

[15] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," *Advances in neural information processing systems*, vol. 26, 2013.

[16] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[17] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[18] NCBI, *MeSH*, Retrieved on 2022-09-30 from https://www.ncbi.nlm.nih.gov/mesh/68065886, 2020.

[19] (1996). "Pubmed," [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/ (visited on 07/29/2022).

[20] (1971). "Medical literature analysis and retrieval system online," [Online]. Available: https://www.nlm.nih.gov/medline/medline_overview.html (visited on 07/29/2022).

[21] L. Eggertson, "Lancet retracts 12-year-old article linking autism to mmr vaccines," *Canadian Medical Association. Journal*, vol. 182, no. 4, E199, 2010.

[22] R. Likert, "A technique for the measurement of attitudes.," *Archives of psychology*, 1932.

[23] D. Miller, *Leveraging bert for extractive text summarization on lectures*, 2019. DOI: 10.48550/ARXIV.1906.04165. [Online]. Available: https://arxiv.org/abs/1906.04165.

[24] Y. Zhang, Y. Zhang, P. Qi, C. D. Manning, and C. P. Langlotz, "Biomedical and clinical english model packages for the stanza python nlp library," *Journal of the American Medical Informatics Association*, vol. 28, no. 9, pp. 1892–1899, 2021.

[25] R. Campos, V. Mangaravite, A. Pasquali, A. Jorge, C. Nunes, and A. Jatowt, "Yake! keyword extraction from single documents using multiple local features," *Information Sciences*, vol. 509, pp. 257–289, 2020, ISSN: 0020-0255. DOI: https://doi.org/10.1016/j.ins.2019.09.013. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0020025519308588.

[26] R. Campos, V. Mangaravite, A. Pasquali, A. M. Jorge, C. Nunes, and A. Jatowt, "A text feature based automatic keyword extraction method for single documents," in *Advances in Information Retrieval*, G. Pasi, B. Piwowarski, L. Azzopardi, and A. Hanbury, Eds., Cham: Springer International Publishing, 2018, pp. 684–691, ISBN: 978-3-319-76941-7.

[27] M. Honnibal, I. Montani, S. Van Landeghem, and A. Boyd, "spaCy: Industrial-strength Natural Language Processing in Python," 2020. DOI: 10.5281/zenodo.1212303.

[28] P. J. A. Cock, T. Antao, J. T. Chang, B. A. Chapman, C. J. Cox, A. Dalke, I. Friedberg, T. Hamelryck, F. Kauff, B. Wilczynski, and M. J. L. de Hoon, "Biopython: freely available Python tools for computational molecular biology and bioinformatics," *Bioinformatics*, vol. 25, no. 11, pp. 1422–1423, Mar. 2009, ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btp163. eprint: https://academic.oup.com/bioinformatics/article-pdf/25/11/1422/944180/btp163.pdf. [Online]. Available: https://doi.org/10.1093/bioinformatics/btp163.

[29] (1836). "National library of medicine," [Online]. Available: https://www.nlm.nih.gov/ (visited on 07/29/2022).

[30] J. H. Medicine, *The Dangers of Uncontrolled Sleep Apnea*, Retrieved on 2022-09-30 from https://www.hopkinsmedicine.org/health/wellness-and-prevention/the-dangers-of-uncontrolled-sleep-apnea, 2022.

[31] C. Pazzanese, *Battling the 'Pandemic of Misinformation'*, Retrieved on 2020-05-08 from https://news.harvard.edu/gazette/story/2020/05/social-media-used-to-spread-create-covid-19-falsehoods/, 2020.

[32] K.-C. Yang, C. Torres-Lugo, and F. Menczer, "Prevalence of Low-Credibility Information on Twitter During the COVID-19 Outbreak," *arXiv 2004.14484*, 2020.

[33] E. Ferrara, "What Types of COVID-19 Conspiracies are Populated by Twitter Bots?" *First Monday*, vol. 25, no. 6, May 2020.

[34] Y. Wang, M. McKee, A. Torbica, and D. Stuckler, "Systematic Literature Review on the Spread of Health-Related Misinformation on Social Media," *Social Science & Medicine*, vol. 240, p. 112 552, 2019.

[35] D. M. Lazer, M. A. Baum, Y. Benkler, A. J. Berinsky, K. M. Greenhill, F. Menczer, M. J. Metzger, B. Nyhan, G. Pennycook, D. Rothschild, *et al.*, "The Science of Fake News," *Science*, vol. 359, no. 6380, pp. 1094–1096, 2018.

[36] G. Eysenbach *et al.*, "How to Fight an Infodemic: The Four Pillars of Infodemic Management," *Journal of Medical Internet Research*, vol. 22, no. 6, e21820, 2020.

[37] C. Castillo, M. Mendoza, and B. Poblete, "Information Credibility on Twitter," in *Proceedings of the 20th International Conference on World Wide Web*, 2011, pp. 675–684.

[38] I. Hernández-García and T. Giménez-Júlvez, "Assessment of Health Information about COVID-19 Prevention on the Internet: Infodemiological Study," *JMIR Public Health and Surveillance*, vol. 6, no. 2, e18717, 2020.

[39] L. Singh, S. Bansal, L. Bode, C. Budak, G. Chi, K. Kawintiranon, C. Padden, R. Vanarsdall, E. Vraga, and Y. Wang, "A First Look at COVID-19 Information and Misinformation Sharing on Twitter," *arXiv preprint arXiv:2003.13907*, 2020.

[40] A. Bridgman, E. Merkley, P. J. Loewen, T. Owen, D. Ruths, L. Teichmann, and O. Zhilin, "The Causes and Consequences of COVID-19 Misperceptions: Understanding the Role of News and Social Media," *Harvard Kennedy School Misinformation Review*, vol. 1, no. 3, 2020.

[41] V. Molter and R. DiResta, "Pandemics & Propaganda: How Chinese State Media Creates and Propagates CCP Coronavirus Narratives," *Harvard Kennedy School Misinformation Review*, vol. 1, no. 3, 2020.

[42] S. B. Naeem, R. Bhatti, and A. Khan, "An Exploration of How Fake News is Taking Over Social Media and Putting Public Health at Risk," *Health Information & Libraries Journal*, 2020.

[43] M. O. Lwin, J. Lu, A. Sheldenkar, P. J. Schulz, W. Shin, R. Gupta, and Y. Yang, "Global Sentiments Surrounding the COVID-19 Pandemic on Twitter: Analysis of Twitter Trends," *JMIR Public Health and Surveillance*, vol. 6, no. 2, e19447, 2020.

[44] J. Agley and Y. Xiao, "Misinformation about COVID-19: Evidence for Differential Latent Profiles and a Strong Association with Trust in Science," *BMC Public Health*, vol. 21, no. 1, pp. 1–12, 2021.

[45] V. Tangcharoensathien, N. Calleja, T. Nguyen, T. Purnat, M. D'Agostino, S. Garcia-Saiso, M. Landry, A. Rashidian, C. Hamilton, A. AbdAllah, *et al.*, "Framework for Managing the COVID-19 Infodemic: Methods and Results of an Online, Crowdsourced WHO Technical Consultation," *Journal of Medical Internet Research*, vol. 22, no. 6, e19659, 2020.

[46] T. Hossain, R. L. Logan IV, A. Ugarte, Y. Matsubara, S. Singh, and S. Young, "Detecting COVID-19 Misinformation on Social Media," *ACL 2020 Workshop*, 2020.

[47] G. K. Shahi and D. Nandini, "FakeCovid–A Multilingual Cross-domain Fact Check News Dataset for COVID-19," *arXiv 2006.11343*, 2020.

[48] L. L. Wang, K. Lo, Y. Chandrasekhar, R. Reas, J. Yang, D. Eide, K. Funk, R. M. Kinney, Z. Liu, W. Merrill, P. Mooney, D. Murdick, D. Rishi, J. Sheehan, Z. Shen, B. Stilson, A. Wade, K. Wang, C. Wilhelm, B. Xie, D. Raymond, D. S. Weld, O. Etzioni, and S. Kohlmeier, "CORD-19: The Covid-19 Open Research Dataset," *arXiv 2004.05125*, 2020.

[49] R. Oshikawa, J. Qian, and W. Y. Wang, "A Survey on Natural Language Processing for Fake News Detection," *arXiv preprint arXiv:1811.00770*, 2018.

[50] D. Orso, N. Federici, R. Copetti, L. Vetrugno, and T. Bove, "Infodemic and the Spread of Fake News in the COVID-19 Era," *European Journal of Emergency Medicine*, 2020.

[51] T. L. I. Diseases, "The COVID-19 Infodemic," *The Lancet. Infectious Diseases*, vol. 20, no. 8, p. 875, 2020.

[52] A. Vaezi and S. H. Javanmard, "Infodemic and Risk Communication in the eEra of CoV-19," *Advanced Biomedical Research*, vol. 9, 2020.

[53] X. Meng, W. Ren, and Y. Zhou, "Transformer-Based Language Model Fine-Tuning Methods for COVID-19 Fake News Detection," in *Combating Online Hostile Posts in Regional Languages during Emergency Situation: First International Workshop, CONSTRAINT 2021, Collocated with AAAI 2021, Virtual Event, February 8, 2021, Revised Selected Papers*, Springer Nature, 2021, p. 83.

[54] A. Wani, I. Joshi, S. Khandve, V. Wagh, and R. Joshi, "Evaluating Deep Learning Approaches for Covid19 Fake News Detection," in *Combating Online Hostile Posts in Regional Languages during Emergency Situation: First International Workshop, CONSTRAINT 2021, Collocated with AAAI 2021, Virtual Event, February 8, 2021, Revised Selected Papers*, Springer Nature, 2021, p. 153.

[55] G. Preda, *Tweets with the Hashtag #covid19*, Retrieved on 2020-09-30 from https://www.kaggle.com/gpreda/covid19-tweets, 2020.

[56] G. Pennycook, T. D. Cannon, and D. G. Rand, "Prior Exposure Increases Perceived Accuracy of Fake News," *Journal of Experimental Psychology*, vol. 147, no. 12, p. 1865, 2018.

[57] J. P. Guidry, C. A. Miller, A. J. Ksinan, J. M. Rohan, M. A. Winter, K. E. Carlyle, and B. F. Fuemmeler, "COVID-19–Related Misinformation among Parents of Patients with Pediatric Cancer," *Emerging Infectious Diseases*, vol. 27, no. 2, p. 650, 2021.

[58] F. Scaggs Huang, P. Spearman, N. Baldwin, and J. K. Schaffzin, "Pediatric Infectious Disease Specialists: An Answer to Social Media Misinformation on Coronavirus Disease 2019," *Journal of the Pediatric Infectious Diseases Society*, Dec. 2020.

[59] P. Patwa, M. Bhardwaj, V. Guptha, G. Kumari, S. Sharma, S. PYKL, A. Das, A. Ekbal, S. Akhtar, and T. Chakraborty, "Overview of CONSTRAINT 2021 Shared Tasks: Detecting English COVID-19 Fake News and Hindi Hostile Posts," in *Proceedings of the First Workshop on Combating Online Hostile Posts in Regional Languages during Emergency Situation (CONSTRAINT)*, Springer, 2021.

[60] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," *arXiv preprint arXiv:1301.3781*, 2013.

[61] F. Godin, B. Vandersmissen, W. De Neve, and R. Van de Walle, "Multimedia Lab @ ACL W-NUT NER Shared Task: Named Entity Recognition for Twitter Microposts using Distributed Word Representations," in *Proceedings of the Workshop on Noisy User-Generated Text*, 2015, pp. 146–153.

[62] E. Alpaydm, "Combined 5× 2 CV f test for Comparing Supervised Classification Learning Algorithms," *Neural Computation*, vol. 11, no. 8, pp. 1885–1892, 1999.

[63] J. Mansky. (2018). "The Age-Old Problem of "Fake News"." Accessed on 2021-11-24, [Online]. Available: https://www.smithsonianmag.com/history/age-old-problem-fake-news-180968945/.

[64] (2021). "Methodology." Accessed on 2021-11-24, [Online]. Available: https://mediabiasfactcheck.com/methodology/.

[65] (2021). "International fact-checking network." Accessed on 2021-11-24, [Online]. Available: https://www.poynter.org/ifcn/.

[66] (2021). "Verified signatories of the ifcn code of principles." Accessed on 2021-11-24, [Online]. Available: https://ifcncodeofprinciples.poynter.org/signatories.

[67] S. Xu, R. Huang, L. S. Sy, S. C. Glenn, D. S. Ryan, K. Morrissette, D. K. Shay, G. Vazquez-Benitez, J. M. Glanz, N. P. Klein, *et al.*, "Covid-19 vaccination and non–covid-19 mortality risk—seven integrated health care organizations, united states, december 14, 2020–july 31, 2021," *Morbidity and Mortality Weekly Report*, vol. 70, no. 43, p. 1520, 2021.

[68] (2021). "China's wandering elephants on the move again," [Online]. Available: https://twitter.com/ABC/status/1403990816373231616?ref_src=twsrc%5C%5Etfw (visited on 07/29/2022).

[69] (2021). "Millions of people in china can't stop watching a pack of wandering elephants," [Online]. Available: cnn.com/2021/06/09/china/elephants-china-yunnan-intl-hnk/index.html (visited on 07/29/2022).

[70] S. Mishra, S. Suryavardan, A. Bhaskar, P. Chopra, A. Reganti, P. Patwa, A. Das, T. Chakraborty, A. Sheth, A. Ekbal, and C. Ahuja, "Factify: A multi-modal fact verification dataset," in *Proceedings of the First Workshop on Multimodal Fact-Checking and Hate Speech Detection (DE-FACTIFY)*, 2022.

[71] D. Küçük and F. Can, "Stance detection: A survey," *ACM Computing Surveys*, vol. 53, no. 1, pp. 1–37, 2020.

[72] A. Hanselowski, A. PVS, B. Schiller, F. Caspelherr, D. Chaudhuri, C. M. Meyer, and I. Gurevych, "A retrospective analysis of the fake news challenge stance-detection task," in *Proceedings of the 27th International Conference on Computational Linguistics*, Santa Fe, New Mexico, USA: Association for Computational Linguistics, Aug. 2018, pp. 1859–1874. [Online]. Available: https://aclanthology.org/C18-1158.

[73] B. Riedel, I. Augenstein, G. P. Spithourakis, and S. Riedel, "A simple but tough-to-beat baseline for the fake news challenge stance detection task," *arXiv preprint arXiv:1707.03264*, 2017.

[74] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Nov. 2019. [Online]. Available: https://arxiv.org/abs/1908.10084.

[75] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1251–1258.

[76] F. Chollet *et al.*, *Keras*, https://keras.io, 2015.

[77] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.

[78] Y. Pan, D. Sibley, and S. Baird, *Fake News Challenge - Team SOLAT IN THE SWEN*, https://github.com/Cisco-Talos/fnc-1, 2018.

[79] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16, ACM, 2016, pp. 785–794, ISBN: 978-1-4503-4232-2. DOI: 10.1145/2939672.2939785. [Online]. Available: http://doi.acm.org/10.1145/2939672.2939785.

[80] A. Hanselowski, A. PVS, B. Schiller, and F. Caspelherr, *Athene_system*, https://github.com/hanselowski/athene_system, 2018.

[81] S. Özcan, santiagonasar, Rusty, Rushat, L. Lopez, and a. Piotti, *Preprocessor*, https://github.com/s/preprocessor, 2020.