THESIS

COVID-19 MISINFORMATION ON TWITTER: THE ROLE OF DECEPTIVE SUPPORT

Submitted by

Fateme Hashemi Chaleshtori

Department of Computer Science

In partial fulfillment of the requirements

For the Degree of Master of Science

Colorado State University

Fort Collins, Colorado

Summer 2022

Master's Committee:

Advisor: Indrakshi Ray

Charles W. Anderson
Yashwant K. Malaiya
Henry Adams

ABSTRACT

COVID-19 MISINFORMATION ON TWITTER: THE ROLE OF DECEPTIVE SUPPORT

Social media platforms like Twitter are major dissemination point for information and the COVID-19 pandemic is no exception. But not all of the information comes from reliable sources, which raises doubts about their validity. In social media posts, writers reference news articles to gain credibility by leveraging the trust readers have in reputable news outlets. However, there is not always a positive correlation between the cited article and the social media posting. Targeting the Twitter platform, this study presents a novel pipeline to determine whether a Tweet is indeed supported by the news article it refers to. The approach follows two general objectives: to develop a model capable of detecting Tweets containing claims that are worthy of fact-checking and then, to assess whether the claims made in a given Tweet are supported by the news article it cites. In the event that a Tweet is found to be trustworthy, we extract its claim via a sequence labeling approach. In doing so, we seek to reduce the noise and highlight the informative parts of a Tweet. Instead of detecting erroneous and invalid information by analyzing the propagation patterns or ensuing examination of Tweets against already proven statements, this study aims to identify reliable support (or lack thereof) before misinformation spreads. Our research reveals that 14.5% of the Tweets are not factual and therefore not worth checking. An effective filter like this is especially useful when looking at a platform such as Twitter, where hundreds of thousands of posts are created every day. Further, our analysis indicates that among the Tweets which refer to a news article as evidence of a factual claim, at least 1% of those Tweets are not substantiated by the article, and therefore mislead the reader.

ACKNOWLEDGEMENTS

My sincere thanks go out to my advisor, Professor Indrakshi Ray, for the help, guidance, and support she provided regarding the completion of this project.

My family and friends have been very supportive of me in this journey, and I would like to thank them for their love and support.

Last but not least, I want to thank the Computer Science Department staff for all their help during my time at Colorado State University, especially in the tough situation of COVID-19.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

1.1 Problem motivation

The detection of misinformation propagation is among the most important tasks in natural language processing. Fake news, propaganda, conspiracy theories, etc., are all examples of misinformation [101, 104, 105]. This issue becomes even more urgent when a crisis like a pandemic strikes. Additionally, the popularity of online social media platforms these days, where there are no restrictions applied to the veracity of published content, further accentuates the significance of this task.

On March 11, 2020, the World Health Organization (WHO) declared COVID-19 as a pandemic – "the worldwide spread of a new disease" [90, 92]. The word "pandemic" highlights the severity of this condition, while simultaneously it can arouse fear and panic among a large number of people. COVID-19 represents a significant threat to global human well-being. It presents a high level of novelty and uncertainty, and as a result, many individuals are turning to online media sources to obtain answers to their questions, to learn more about what threat they are facing, and how they can minimize the risk and protect themselves [35, 48]. And thus, a tremendous amount of data related to this pandemic is being circulated on social media platforms to the extent that the WHO reported that "we're not just fighting an epidemic; we're fighting an *infodemic*" [91, 114]. Not all the information found online is reliable; it contains invalid information as well and in this environment, it is critical to develop tools that mark misinformation on the web. In the wake of the COVID-19 pandemic, it has become increasingly important to counter misinformation. Misinformation is the existence of objectively untrue or false information, which is lacking sufficient evidence and scientific and expert assessment [78]. The consequence of this can be devastating for individuals, who may take decisions that could have severe negative outcomes when they have received incorrect information.

It is particularly alarming to see how pervasive this kind of misinformation is. More than 600 worth of Tweets related to COVID-19 were studied recently, and about 70% of what was shared contained medical claims and public health information, and nearly 25% contained misinformation, while 107 Tweets (17.4%) spread untrue information [58]. Every major disease outbreak, including Ebola [80], Zika [71], Yellow Fever [79], and now COVID-19, is accompanied by misinformation. Many studies have been devoted to examining the dissemination patterns of misinformation (*e.g.*, [102]) or identifying fake news based on a pool of fact-checked statements related to the discussed topic. Others have focused on identifying the rumor-bearing posts within a specific topic [36]. In essence, efforts such as these are only possible after the propagation of false information, and they cannot be used to prevent misinformation from spreading. There are four crucial steps in disaster response: *Prevention*, *Preparedness*, *Response*, and *Recovery*. Additionally, effective communication during a crisis is important in minimizing the damage caused by the event [15]. This study focuses on the first step, prevention, by relying only on the content that has been shared (the Tweet itself and the news article it cites) and no additional resources. Preventing the misinformation to be propagated can also facilitate proper communication between the government and the public during the pandemic by removing the obstacles of untrue information. When misinformation begins to spread, it is very difficult to contain it. This is because previous exposure to false information makes it more likely that the new information will appear to be true [82]. Consequently, misinformation spreads much faster than accurate news [79, 97, 109]. It is crucial to identify misinformation on social media *early on* so as to prevent the spreading of false claims.

Posts with original content on social media are fundamentally different from those which include re-transmissions. Arif et al. [8] have distinguished between them as "original content" and "derivative content". They report that when a claim enters the network through an influential account, *e.g.* one with many followers, it encourages a substantial volume of derivative content, which leads to a snowball effect. The ordinary users of social media are unlikely to purposefully propagate inaccurate information in the network. Rather, the information is propagated because it *appears*, on the surface, to be credible (as illustrated by Fig. 1.1 and its propagation in Fig. 1.2).

Figure 1.1: Original content entering the social media information space through the "New York Post" institutional Twitter account. With 2.1M followers (accessed: May 21, 2021), this is an entry with a large footprint.

Figure 1.2: A corresponding derived content: re-transmission of the source with added remarks.

To establish credibility, users often cite trustworthy sources, such as renowned news outlets, when sharing information [31]. While journalism has long been viewed as a profession based on the accuracy and veracity of reported information [98], social media users are not subject to the same restrictions on commentary they may post when referring to news articles. Often, such commentary deviates considerably from the claims made in the cited source, to the point that the source becomes totally irrelevant to the commentary. Yet readers of such posts often rely on the credibility of cited sources and presume that the commentary is true simply because it cites a well-known source, without digging deeper and comparing the source with the post; the belief is formed without having examined the original material. Perhaps homophily in social networks contributes to this phenomenon, as many readers read the commentary in part due to confirmation bias [25, 103]. Posts like this are problematic and harmful because misinformation is spread in a seemingly trustworthy manner. Obviously, this is what we would like to keep from happening. To serve this goal, this study is designed based on two objectives: (i) identifying the posts that carry factual claims derived from credible sources that are worth checking and (ii) investigating and comparing the post with the cited source to determine whether the source supports the claim in the post or whether the writer is attempting to gain credibility by misrepresenting the source.

1.2 Problem definition

Our objective is to detect Tweets that refer to irrelevant news articles to deceivingly support their claims. As such, the authors abuse the trust that readers place in trustworthy news outlets and refer to an article to gain trust, even though it does not necessarily support their claims. The following questions are posed about each COVID-19-related post that cites a news article:

(I) Does the post contain an objective claim and is that claim considered important enough to be verified?

(II) Which part of the Tweet constitutes a claim?

(III) Are the claims in the post supported by the cited news article?

In response to the first question, we developed a model for identifying tweets that are *check-worthy*, that is, those which contain factual claims. Following this, we extract the claim sequence from the Tweets for the purpose of noise reduction. The last step is to determine whether the posted claim is consistent with the cited sources. In cases where a post cites a news article but makes claims that are not supported by the report, that can be considered rumor propagation or misinformation.

1.3 Target data

The information disseminated through social media may be analyzed and viewed from many perspectives. Imran et al. [52] categorize these dimensions in terms of time, location, topic, type of information, subjectivity (*i.e.*, factual claims as opposed to opinions or other emotional content), information source, and credibility. Our research on misinformation on social media is unique among existing studies because we investigate "perceived credibility" in posts. Our concern is whether the content derived from the original content is a true and corresponding reflection of the source, as it is re-transmitted across the social network. We only examine Tweets that: are related to the COVID-19 pandemic, contain factual claims, are check-worthy, and appear to provide support by referring to a news article. The scope of this report does not include tweets that express

opinions, share emotional content or present factual claims without explicit external support to provide credibility. Following is a more detailed discussion of these restrictions.

1. **Controlling for the topic**

   We use a large dataset of Tweets related to COVID-19. This open dataset was gathered by Banda et al. [10] for the purpose of integrated research in epidemiology, misinformation, and related directions. This is a continually growing dataset and it encompasses 383 million Tweets at the time this work was conducted.

2. **Filtering subjectivity**

   A significant number of posts do not include any subjective information. As an example, Tweets frequently share personal experiences, use emotive language, and include ironic or sarcastic statements. Therefore, our first step will be to pull apart tweets with factual statements from the rest of the data.

3. **Check-worthiness**

   In addition to the above controls, prior Detecting fake news in times of crisis, such as natural disasters and epidemics, often involves ranking information nuggets in order of importance (*e.g.*, [60]). As a result of this approach, a considerable body of work has been published on the subject of scoring information nuggets based on check-worthiness [9, 43, 121]. With the increasing popularity of social networks and the abundance of information available on the Internet, it has become increasingly difficult to separate check-worthy information from the rest. We, therefore, incorporate the consideration of check-worthiness into our analysis to only focus on Tweets whose veracity deserves investigation and discard unimportant Tweets.

4. **Controlling for perceived credibility**

   Posts that make factual claims are not necessarily credible. A user who posts derived content from an original article creates this perception by including a link to the original news article

in the Tweet. In fact, the citation to an independent publication lends credibility to the claim. Consequently, we retain only Tweets that contain a link to a news article.

The above steps describe the first task in our entire misinformation detection pipeline. After this step, we are left with a dataset of factual check-worthy claims in the form of Tweets that cite news articles. This forms the input to our second task, which determines whether or not a Tweet indeed propagates the claim made in the cited news article.

1.4 Proposed approach

In this study, we attempt to detect Tweets that contain deceptive citations intended to gain readers' trust and mislead them. We build a system that isolates Tweets that make factual claims and are worth checking, as the first step. Toward this end, Tweets must be converted into vector representations, and then they are categorized as either check-worthy or non-check-worthy. Using Transformer-based models, we generate the vector representations and add a classification layer to the top to categorize the Tweets.

The remaining Tweets are fed to two other modules for claim sequencing and identifying whether the cited news article is indeed supporting the Tweet or not. Claim sequencing allows us to extract the exact claim from the Tweets. Thus, it is possible to reduce the amount of noise in the Tweet by omitting any uninformative tokens, and so, we can focus on the main argument in the Tweet. As in the previous task, we are using language models based on Transformers to encode the text. Following that, we are utilizing sequence labeling techniques, similar to the approach used for Named Entity Recognition, to analyze the Tweet and determine which parts of it constitute a claim.

The last task is to analyze the connection between a Tweet and the news article it cites in order to identify whether the cited news article truly supports the claim made in the Tweet. In several cases, we observe that the Tweets only reprint the headlines with slight variations. Therefore, we divide this task into two sub-tasks: the first is to examine only the headline of the news story; if

the headline and Tweet do not match, we move on to the second part, which involves analyzing the news body in depth. Once again, we are converting the text to numbers using Transformer-based models. We will then use a classification model, rather than approaches such as cosine similarity, to determine whether a Tweet is misleading or not.

1.5 Our contribution

Our objective in this study is to detect a specific type of fake news on Twitter where the author of a Tweet contains a citation to an irrelevant news article in the Tweet, solely for the purpose of gaining the audiences' trust and making them believe the Tweet content by exploiting the reputability of the highly-regarded news agencies. There is a tendency among well-known news outlets to publish trustworthy information often, so merely citing a news article from one of these outlets almost assures the reader that the content is correct without even examining the source. This is while the news article can be totally irrelevant to the content of the Tweet. To detect such Tweets, we developed a system that has three main components:

1. Detecting check-worthy Tweets: Separates those Tweets with factual claims that are important enough to be analyzed. This step filters out 14.5% of the Tweets in our dataset with an average precision of 84.4%. It can also be used as a preprocessing step for other objectives to reduce the noise of the data.

2. Claim extraction: Defines the tokens in the Tweet that contributes to the claim made in the Tweet. Using this method can reduce the noise in the text and has a wide range of applications in other fields as well. Our claim sequencing module can extract the claim from text with an average accuracy of around 74%.

3. Evaluating the relationship between the Tweet and the cited news source: Analyzes whether or not the news articles referred to in the Tweet correspond with the content of the Tweet. Our final results show that at least 1% of the Tweets that refer to a news article are deceptive and include the citation to the news to fool the readers into believing the Tweet content.

1.6 Publications and attribution

This thesis is based on the paper "Seeing Should Probably not be Believing: The Role of Deceptive Support in COVID-19 Misinformation on Twitter" published in the *Journal of Data and Information Quality* [120] and two manuscripts, "Claim Sequencing and Stance Detection in COVID-19 Twitter Data" and "Topic-oriented Tempo-Spatial Analysis of COVID-19 Tweets".

In the first article, the author of this thesis focused on Task 1, retaining check-worthy Tweets, discussed in Chapter 5, and Dr. Chaoyuan Zou focused on the implementation of Task 2, classifying the Tweets as misinformation or authentic based on the cited news article, discussed in Chapter 7. The next report is mainly about claim extraction using the sequence labeling technique. The author of this thesis managed the manual annotation process and prepared the train data for different sub-tasks. She also carried out experiments related to claim existence detection, claim extraction, and stance detection. The last manuscript is a different analysis report on the huge dataset of COVID-19-related Tweets the thesis's author has gathered over 18 months. Detailed information about these two studies is provided in Chapter 6.

In the remainder of this report, we discuss our work in the greater context of prior research in this field and provide background knowledge in Chapter 2 and we present our proposed approach and the detailed architecture of our pipeline in Chapter 3 and the data preparation steps in Chapter 4. In Chapter 5 our solution to identify check-worthy factual claims is elaborated. Chapter 6 explains our proposed method for obtaining claim information from text. In Chapter 7 we discuss the procedure we follow to distinguish faithfully represented derived content from potential misinformation and unverifiable claims. Subsequently, we conclude the work along with a brief discussion of future research directions in Chapter 8.

Chapter 2

Literature Review and Background Knowledge

Systems designed for early detection of misinformation often rely on a combination of signals from the user, the dissemination pattern, and the content of the post [116, 117]. As an example, Jain et al. [53] collected and clustered the Tweets, found similar content from credible news channels as ground-truth information, and then, they compared the Tweet's content to the reliable content by sentiment and semantic analysis. In case of a mismatch, the authors labeled the Tweet as misinformation. In this body of work, a fixed set of sources were assumed to be trustworthy – an approach that has been criticized by qualitative research for its potential implicit bias [49, 107]. There are very few exceptions to this approach, *e.g.*, Al-Rakhami and Al-Amri [4], that rely on large-scale manual annotations – a particularly time-intensive approach to resolve a time-sensitive issue.

Accessing high-quality data is crucial in detecting misinformation in social networks by machine learning techniques. Various research attempted to address this challenge. Banda et al. [10] released a very large open-source dataset with more than 383 million Tweets. The dataset includes only the Tweet IDs but is accompanied by the required scripts to rehydrate the Tweets, *i.e.*, retrieve the contents of a Tweet through the use of the Twitter API. The original dataset contains both Tweets and retweets, which allows tracking the dissemination of Tweets; but, a cleaned version has been released as well that has no retweets, which is suitable for analysis of the context of the Tweets. On average, this cleaning step removes 75% of the Tweets. This work does not detect misinformation, but the dataset they published is invaluable to others who intend to research misinformation and examine ML models for this purpose.

Multilingual Datasets. While most released COVID-19 Twitter datasets are in English, the dataset released by Banda et al. [10] includes Tweets in other languages, such as French, German, Russian, and Spanish. Gao et al. [33] released another multilingual dataset of English and Japanese posts on Twitter, and Chinese posts on *Weibo*, while Alqurashi et al. [7] released an Arabic COVID-19

dataset of Tweets. Haouari et al. [42] presented a large Arabic language dataset of Tweets related to COVID-19, along with the propagation networks. In English language datasets, propagation has been studied extensively. Rumor propagation patterns have been studied for several years now, with application in early detection, determining support, and determining their veracity [40, 87], while for other languages it is not well-studied. In this study, we limit the scope of our work to only English Tweets without losing generality.

2.1 Misinformation spread analysis

Huang and Carley [51] collected more than 67 million Tweets from 12 million users with metadata related to geographical information, social identities, and the political orientation of users by tracking COVID-19 Twitter conversations. The data includes metadata related to geographical information, social identities, and the political orientation of users. By analyzing the information about these 12 million users, they reported that misinformation is more likely to be spread by regular users and within the source country, not internationally. In addition, they reported that many of the Tweets speaking of disinformation storylines and referring to unreliable news sites, are posted by regular users, some of them are bots. Similarly, others have reported that misinformation spreads significantly faster than the truth [97, 109].

Shahi et al. [97] conducted an exploratory study and relied on a list of 7,623 COVID-19-related fact-checked news articles and searched for news articles that are cited in Tweets, resulting in a set of 1,565 unique Tweets. Four classes of *False*, *Partially False*, *True*, and *Other* have been defined. Their analysis reveals that in 70% of the false and partially false categories of misinformation verified Twitter handles such as celebrities and organizations are involved either by helping to spread or creating the content. The authors have not proposed a ready-to-use model that can be applied for misinformation detection tasks but their approach and the parameters they used for analysis can be considered in future works.

Vosoughi et al. [109] investigated the publication of fake, verified, and mixed information on Twitter. Instead of focusing on a specific topic, they considered a longer duration: 2006 to 2017.

The diffusion of rumor cascades has been analyzed by considering the replies and retweets and reported that false information on Twitter tends to be retweeted by many more users and gets spread much faster compared to true information, especially when it is about a political issue.

Some recent work has looked at the spread of misinformation using epidemiological models as well. For example, Cinelli et al. [17] analyzed the spread of more than 8 million posts on social networks with epidemic models using reproduction number (R0), *i.e.* the average number of secondary cases an infectious individual will create. They concluded that both questionable and reliable news spread with similar diffusion patterns, which indicates that it is impossible to detect fake news solely using meta-data, and analyzing the language and the content is crucially important.

2.2 Misinformation detection

Memon and Carley [70] manually annotated more than 4.5K COVID-19-related Tweets. The dataset has a diverse set of categories for 17 types of information and misinformation; *i.e. Irrelevant*, *Conspiracy*, *True Treatment*, *Fake Cure*, *Fake Treatment*, *etc.* One cause for concern is that the data has been annotated by only one annotator. In this work, they looked at various attributes of two target groups: (i) misinformed users (who are actively posting misinformation) and (ii) informed users (who are actively spreading true information). Their methodology involves two steps. In the first step, the authors used a keyword-based Twitter search API for data collection. In the second step, the annotator categorized and labeled the Tweets into 17 classes, based on the types of information. The authors concluded that misinformed users' communities may be denser and more organized, while informed users use more narrative language. The authors observed that bots exist in both misinformed and informed communities, noticeably more among the misinformed users.

Hossain et al. [50] divided misinformation detection task into two sub-tasks of (i) retrieval of misconceptions relevant to posts being checked for veracity, and (ii) stance detection to identify whether the posts *Agree*, *Disagree*, or express *No Stance* towards the retrieved misconceptions. Authors then collected and rephrased a set of COVID-19-related misconceptions from a Wikipedia

entry, paired with 6.7K Tweets, and determined the stance of the Tweets against that misconception. Their goal was to determine whether NLP models can be adapted to the task of detecting misinformation without further training. The authors used relevant datasets to pre-train the models and to make the models domain-specific.

They have selected multiple NLP models, some that are suitable for misconception retrievals such as BM25 and Cosine Similarity with different embedding models like BERTSCORE, and some that can be used for stance detection. The stance detection sub-task can be considered to be equivalent to Natural Language Inference (NLI) problem, and thus, the authors used linear classifiers trained on NLI datasets combined with other models such as average GloVe embeddings as well as Sentence-BERT and Bidirectional LSTM encoding. Their results demonstrate that domain adoption, retraining language models on a corpus of COVID-19 tweets, increases the performance noticeably in both tasks of misconception retrieval and stance detection. Keeping the dataset updated is challenging as new rumors are being circulated and older ones may get obsolete as the pandemic continues. In addition, many of the Tweets in the dataset may not be available due to various reasons, *e.g.* have been deleted by users or removed by Twitter because they are detected as misinformation.

Kim and Walker [56] used a different strategy for defining misinformation. This study relied on the official recommendations of reputable health institutions to find the reply Tweets that make the same claim. They confirm that this method is more effective at identifying Tweets with misinformation than searching based on keywords. The authors investigated the applicability of the proposed model with an example of advice from WHO related to *antibiotics* and *COVID-19 cure*. They collected more than 16K English reply Tweets during three months based on a specific combination of keywords closely related to the selected authentic advice, and parent Tweets were then obtained. These parent Tweets could potentially contain misinformation. Ignoring non-English and self-reply parent Tweets and filtering them based on another set of keywords, 573 pairs of the parent-reply pair Tweets were collected. Afterward, the sentence-BERT model converted reply Tweets and the advice to vectors, and the cosine similarity between each vector of reply Tweets

and the vector of the advice is calculated. 200 reply Tweets with unique parent Tweets are selected where they have the highest cosine similarity scores calculated between the reply Tweet and the advice vectors. By manual inspection, authors detected parent Tweets with misinformation and then they added meta-data obtained from the users posting Tweets with misinformation, like timelines of friends and followers, to realize the extent of the spread of misinformation locally. In this approach, there should be replies in response to a misinformation Tweet with authentic information. Consequently, misinformation without replies containing authentic information will not be detected. In addition, this approach requires manual checking which is laborious and error-prone.

One example of studying non-English misinformation detection has been done by Kar et al. [55] on Indic languages (Bengali and Hindi) using Multilingual BERT (mBERT)[1]. Authors used the labeled English Tweets in the Infodemic COVID-19 dataset [5] as well as their translation into Bengali with Google Translate API, while retaining the same labels, as a part of their training dataset. They also used the Bengali dataset released in [34], and manually annotated 100 randomly selected Tweets. The Hindi dataset has been created in the same manner; they collected a set of Tweets by keyword searching and then added their Hindi translation. The authors used a zero-shot learning approach; in general, meaning that the set of labels in the training data and the set of labels for the data that the model will be used to classify are disjoint [111]. To perform zero-shot learning in this work, they had experiments in which Tweets in one language were kept for testing and the rest of Tweets in other languages for training the model. They have further augmented the datasets by adding metadata of the Tweets, including the number of retweets and the number of likes, and 22 more features. The authors also defined three novel features. First, *Fact Verification Score*, which is obtained by searching the Tweet text in the Google search engine and taking the average Levenshtein distance between the Tweet text and the titles of search results only from reliable websites. Second, *Bias Score*, which is defined using a Linear Support Vector Machine (SVM) Classifier for specifying the probability that a Tweet contains offensive language. And third, *Source Tweet Embedding*, which is the vector representation of the Tweet text using BERT-

[1]https://github.com/google-research/bert/blob/master/multilingual.md

based models. Four classifiers Multi-Layer Perceptron (MLP), Random Forest Classifier (RFC), SVM, and mBERT were examined and their results show that fine-tuned mBERT achieved the best F1-score of 89% in detecting Tweets with fake news. The disadvantage of this work is the need for manual annotation of a relatively large dataset.

Madani et al. [67] proposed a similar approach for the Moroccan language, using both Tweet and other metadata. For data collection, they got a dataset of fake news represented in [100], that is based on ground truth information from fact-checking websites. Based on that, the authors collected 10K Tweets with fake news related to COVID-19 by keyword searching, and they manually annotated the Tweets as fake or real. These English Tweets and the metadata that they extracted from them, such as Tweet length, Tweet sentiment, friends and followers number of Tweet's owner, and 10 more, form their training and testing dataset. To gather the unlabeled Tweet dataset, they used the Tweepy library and translated the Tweets to Moroccan. For fake Tweet detection, six different machine learning models (Decision Tree, Random Forest, Naive Bayes, Gradient Boosting, and Support Vector Machines, and Multilayer perceptron (MLP)) have been used. In this study, the authors made three important observations. First, the Random Forest classifier outperformed all other models, including the MLP model, with respect to four evaluation metrics, accuracy, precision, recall, and F1-score. Positive correlation between the sentiment of a Tweet and its authenticity, meaning that Tweets with positive sentiment are more likely to be authentic and Tweets with negative sentiment most probably contain misinformation, and the positive effect of metadata on performance are two other observations. In our work, we do not use metadata as we are finding a connection between the Tweet text and the news article that is cited, and thus, what matters most in our work is the Tweet content itself.

Gupta et al. [39] implemented a semi-supervised ranking model that assesses the credibility of Tweets in real-time. They have collected more than 10M Tweets about different events and among them, they randomly selected 500 Tweets for annotation to build a training set for their model. They used crowdsourcing to classify the Tweets into four classes: *Definitely credible, Seems credible, Definitely incredible, and None of the above (skip Tweet).* The model extracts

45 content-related features from the Tweets and the users posting those Tweets, such as number of characters, swear words, pronouns, positive and negative emoticons, number of retweets and replies by the users, and based on these features it gives credibility scores to the Tweets, ranging from 1 (low) to 7 (high). They tested four models that are commonly used for information retrieval, namely, Coordinate AdaRank, RankBoost, Ascent, and SVM-rank. To compare these models they used two evaluation metrics: Normalized Discounted Cumulative Gain (NDCG) to obtain correctness and model running time. Finally, they chose the SVM-rank model which is the second-best model in terms of NDCG@n[2] and is the best one in terms of training time. The model has been used in browser plugins and tested on 1,127 Twitter users over a course of three months, and 5.4 million Tweets credibility scores computed. They observed that features extracted from the Tweets content are more effective in credibility assessment compared to the features extracted from the user accounts. We are also focusing on the content of the Tweets in our work to identify misinformation among the Tweets. The difference between this approach and our view is that we do not look at misinformation detection as a ranking problem, but we offer a binary classification model that either labels a Tweet as misinformation or authentic.

Nguyen et al. [76] designed a shared task, WNUT-2020, to automatically identify informative COVID-19 Tweets, as manual annotation is a cost-intensive solution. This work is not focused on misinformation detection but can be considered as a data filtering step needed for fake news detection. The authors defined an informative Tweet as it offers specific and clear information, and not rumor or prediction, about suspected, affirmed, healed, and deceased COVID-19 cases along with the travel history or location of the cases. From March 1st to June 30th, about 23M non-repeating Tweets related to COVID-19 have been gathered. Authors filtered this corpus by particular keywords like “positive”, “discharge”, “death”, *etc.* to separate candidates for informative Tweets. Among this dataset, a random sample set of 2K Tweets are manually annotated by three annotators with two labels, *informative* and *uninformative*. A classifier is trained on this subset to predict the probability of Tweets being informative for the rest of the Tweets in the dataset.

[2]This means that to calculate the NDCG, the first n records in the ranked list are considered.

Authors sampled 8K Tweets with different informative probabilities. These Tweets are also manually annotated; altogether, they formed a set of 10K Tweets as the final gold standard corpus used for training, validation, and testing the models for the shared task. Authors used fastText [54], a text classification task, as a baseline. The baseline classifier achieves the F1-score, harmonic mean of precision and recall, of 75%. Considering the F1-score, 48 out of 55 participants outperform the baseline model; most of the teams are benefiting from pre-trained language models such as BERT, RoBERTa, XLNet, *etc.* The top 6 teams used CT-BERT while more than half of the teams are leveraging ensemble techniques. The best participant's model reached the F1-score of 96.06% and the accuracy of 91.50%. This work confirms our choice of using pre-trained transformers and fine-tuning them. While eliminating some of the Tweets is a similar task between our work and this study, we considered different definitions based on which we decide to ignore a Tweet; we keep a Tweet if it contains a factual claim which is of interest to the public, while in this work a Tweet is classified as *informative* if it provides direct and clear information about COVID-19 cases.

2.3 Transformer-based language representation models

Our work uses Transformer-based language representation models for different tasks. With the advent of these new natural language representation models based on Transformers, starting with BERT, many of the previous models [69, 85] have been outperformed on a variety of downstream NLP tasks. Because of their superior performance on a number of natural language tasks, they have garnered much attention from researchers. Transformer-based models have an architecture designed for sequence-to-sequence tasks where the goal is to transform a sequence of objects into another sequence, for example, the task of natural language text translation. Transformers are building blocks of language representation models that learn contextual relations between words by determining the weights assigned to different words in an input sentence depending on their importance for a specific task [27]. We point out the important features of transformer-based models that we utilize in our experiments and highlight their main differences.

Bidirectional Encoder Representations from Transformers (BERT) [26] is the new generation of language representation models. This model has been pre-trained on a huge corpus of unlabeled text data to produce a bidirectional vector representation of the input. The first step in the training process is tokenizing the input sentence. BERT has a fixed-size dictionary of vocabulary containing around 30K tokens. Some words are considered input tokens for their entirety, while others are broken down into smaller parts and each part is a token. There are also two special tokens: *[SEP]* for separating the sentences and *[CLS]* that marks the end of the input. BERT is pre-trained on two tasks: Masked Language Modeling (MLM) and Next Sentence Prediction (NSP). In MLM task, around 15% of the input tokens are masked and the model should generate the masked tokens. NSP task is used to help the model understand the relations between sentences. The input for this task is a pair of two sentences, A and B. In half of the cases, B is the actual next sentence of A, and in half of the cases, it is a random sentence from the corpus. The model should predict whether B is the next sentence of A or not.

The RoBERTa model [64] was proposed by Facebook and the University of Washington in July 2019. RoBERTa is trained on a much larger dataset than BERT. Note that, BERT is trained on the English Wikipedia and BookCorpus [118] of size 16G, whereas the dataset used for training RoBERTa contains CC-NEWS data (76G), OpenWebText data (38G), and Stories (31G) as well as the English Wikipedia and BookCorpus (16G). Additionally, it has been optimized for the task of Masked Language Modeling (MLM) with a Dynamic Masking Pattern instead of a Static Masking Pattern. Masked Language Modeling is an NLP task in which some of the tokens in the sentence are masked and the model attempts to predict what those tokens are and reproduce them. This helps the model understand the connection between the tokens in a sentence. With a Static Masking Pattern, the tokens that are veiled are chosen randomly yet the sets of masked and unmasked tokens continue as before during the training phase. On the contrary, Dynamic Masking Pattern picks a new set of tokens to mask whenever a new sequence is fed to the model [64]. Another difference between the BERT model and the RoBERTa model is that RoBERTa does not train for the task of Next Sentence Prediction.

XLNet [113] is another transformer-based language representation model published by Carnegie Mellon University and Google AI Brain Team in Jan 2020. The XLNet model presents permutation language modeling to improve the training, in which all tokens are predicted but in random order. This assists the model with learning bidirectional connections and consequently, it better handles dependencies and relations between words. XLNet was trained on a very large data corpus with more than 130 GB of text data which is much bigger compared to the volume of data used for training the BERT model.

The DistilBERT model [95] is proposed by Hugging Face in March 2020. It is a 40% smaller model compared to BERT (BERT has 110M parameters and DistilBERT has 66M parameters.) but with the same general architecture. As it is smaller, it is 60% faster and has achieved almost the same performance as BERT in the language understanding tasks.

The Electra model [18] is published by Stanford University and Google Brain in March 2020. This model is proposing a task for training, similar to Masked Language Modeling but more efficient, named Replaced Token Detection. In comparison to the BERT model that is a generative model and tries to reconstruct the original token based on the unmasked tokens, the Electra model is using a discriminative module only. The input data to this model is corrupted instead of masked, meaning that some of the tokens are replaced with acceptable alternatives. The model's objective is to predict whether each token is original or replaced. The Masked Language Modeling technique only learns from the masked tokens – typically 15% of the data – while this discriminator module is using all tokens. The idea behind training the Electra model is similar to Generative Adversarial Network (GAN) in that both are trying to distinguish between the original tokens and those that have been replaced, and thus, the loss value is calculated over all tokens of the input rather than just masked tokens.

Chapter 3

Proposed Approach

In our discussion of the overall architecture of the proposed system, we begin by conferring the basic requirements of a fake news detection algorithm, as discussed by Rubin et al. [94], and then present the primary components of the pipeline, which are responsible for (i) data collection, (ii) data preprocessing, (iii) identifying check-worthy factual claims and (iv) discriminating verifiable claims from others.

3.1 Components of the system

Figure 3.1 shows the overall system architecture, with the complete pipeline and its components. To provide a correspondence between the steps in our pipeline and the data, we also present examples of Tweets in Table 3.1.

Data collection

We use the open dataset created by Banda et al. [10] as the starting point, where we obtain the large collection of Tweets pertaining to the COVID-19 pandemic. In parallel, we also collect the complete news articles cited by the Tweets in this dataset. The news articles are collected only for those Tweets that are retained after the data filtering step.

Data preprocessing

On one hand, each Tweet is passed through multiple filters, token-level cleaning such as removal of function words and non-linguistic features (discussed in greater detail in Chapter 4). On the other hand, the news articles cited by these Tweets are collected and processed as well, thereby removing spurious material around the article's content and then splitting the article's content (along with its title) into sentence-level chunks for subsequent use in our final task.

Task 1: Identification of check-worthy factual claims

This is designed as a supervised binary classification task, where each Tweet is designated as check-worthy (CW) or non-check-worthy (NCW). We present the details of this component in Chapter 5.

Task 2: Claim extraction

This module takes the check-worthy Tweets from Task 1 as input and determines which subsection of the Tweet is indeed a claim. In this task, it is assumed that the input Tweets contain a claim since they are check-worthy, however, if a false positive instance finds a way through, our claim sequencer is able to handle it fairly accurately.

Task 3: Assessing the connection between the Tweet and the cited news

Among the multiple models developed for the first task, we use the one with the best performance to feed Tweets with the CW label into the second task. This, too, is designed as a binary classification task. Multiple models and experimental setups are explored and discussed in Chapter 7.

3.2 Requirements

In their analysis of fake news detection systems within the scope of natural language processing (NLP) research, Rubin et al. [94] present nine fundamental requirements. In this work, we take care to ensure that our approach meets these criteria:

1. Our data satisfies the *availability of both truthful and deceptive instances*.

2. It also satisfies *digital textual format accessibility*.

3. It offers *verifiability of "ground truth"* by virtue of the manual annotation of two datasets with ground-truth labels. Our annotations offer high inter-annotator scores (details are discussed in the context of data preparation in Chapter 4 and experimental results in Chapter 5).

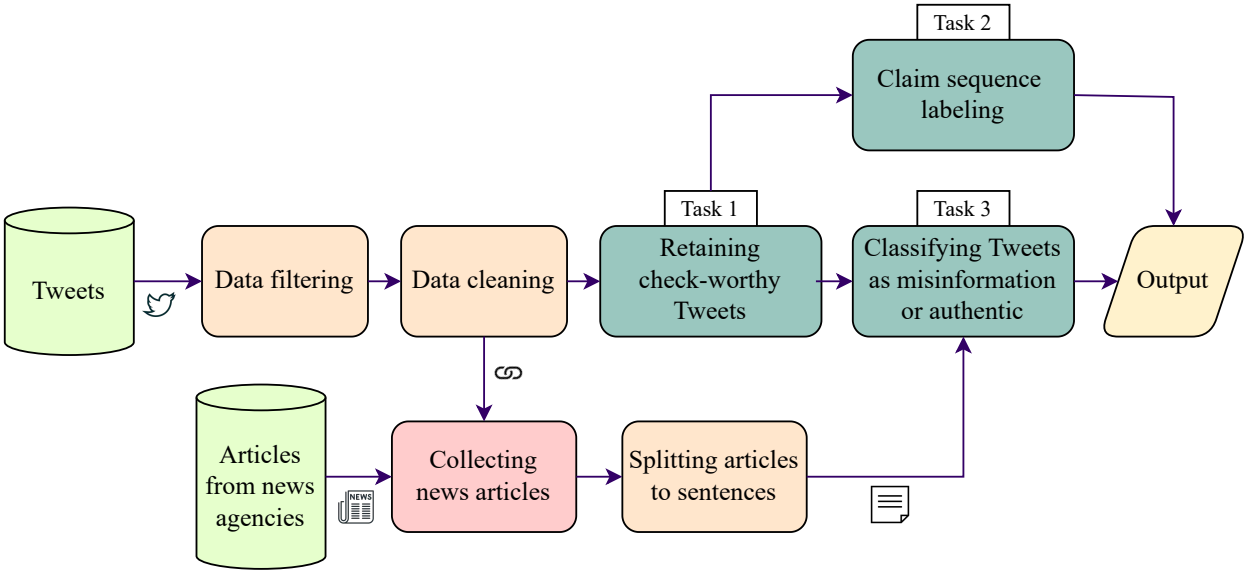


Figure 3.1: System architecture. The pipeline comprises (i) the data collection from Twitter posts and news articles, (ii) data preprocessing – which includes the filtering, cleaning, and splitting into sentence-level chunks, (iii) the first task of identifying Tweets containing check-worthy factual claims, and (iv) the second task of distinguishing the information faithful to the original news content from the rest.

4. Since we use Twitter posts, which are limited to 280 characters, our data adheres to *homogeneity in length*. Further, even though Twitter expanded its character count limit to 280 in November, 2017 [83], only 5% of the English language Tweets over the subsequent one year were longer than 190 characters, and only 9% used more than 140 [84], thus providing even more homogeneity in length than one would expect.

5. Our work also adheres to *homogeneity in writing matter*, in terms of both topic (the COVID-19 pandemic) and genre, and offers comparison across multiple news agencies and social media users.

6. The data used in this work was collected over a period of three months, during the prevalence of the COVID-19 pandemic, and therefore has a *predefined timeframe* of data collection, thereby reducing arbitrary variations that are typically present in corpora collected over shorter "snapshot" periods.

Table 3.1: Sample Twitter posts (Tweets) from our data. Tweets often cite news articles to lend credibility to the shared information: (1) a post not containing terms related to COVID-19, or a link to a news article; (2) a post without any specific check-worthy claim; (3) a statement worth checking vis-à-vis the headline of the linked news article; (4) a statement worth checking vis-à-vis the body of the linked news article; and (5,6,7) a check-worthy claim that is not supported by the cited article, thus merely *appearing* trustworthy.

	Tweet (derived content)	**Corresponding original content (cited news article)**
(1)	Africa deporting Europeans we love to see it *Last accessed*: June 6, 2021	*– no news cited –*
(2)	Coronavirus Map: How To Track Coronavirus Spread Across The Globe via @forbes [https://bit.ly/3upHDao] *Last accessed*: June 6, 2021	**Headline:** Coronavirus Map: How To Track Coronavirus Spread Across The Globe **Body:** As COVID-19 (coronavirus) spreads across the globe, it is helpful and interesting to track the transmission patterns through a coronavirus map
(3)	Native American Health Center Receives Body Bags Instead of Coronavirus Supplies. [https://bit.ly/39LBBJc] *Last accessed*: June 6, 2021	**Headline:** Native American health center receives body bags instead of coronavirus supplies **Body:** A community health center treating Native Americans in the Seattle area issued an urgent call for medical supplies . . .
(4)	Misinformation about Mr. Gates is now the most widespread of all coronavirus falsehoods – New York Times [https://nyti.ms/3fLCoO2] *Last accessed*: June 6, 2021	**Headline:** Bill Gates, at Odds With Trump on Virus, Becomes a Right-Wing Target **Body:** . . . Misinformation about Mr. Gates is now the most widespread of all coronavirus falsehoods . . .
(5)	Italy coronavirus: Italians who attempt to flee lockdown may face jail – CNN [https://cnn.it/3rVRZx8] *Last accessed*: June 6, 2021	**Headline:** All of Italy is in lockdown as coronavirus cases rise **Body:** (CNN)Italy has been put under a dramatic total lockdown, as the coronavirus spreads in the country
(6)	Dow drops 200 points as unemployment claims surge once again via CNBC #news #CNBC [https://rb.gy/jxhy55] *Last accessed*: Feb. 6, 2022	**Headline:** Stocks rise slightly, led by tech; Netflix hits record **Body:** Stocks rose slightly on Thursday, led by tech, as Wall Street grappled
(7)	Federal officials accuse two groups of selling fake coronavirus vaccines and treatment - CNN [https://cnn.it/3eewwck] *Last accessed*: Feb. 6, 2022	**Headline:** Memorial Day weekend: Americans visit beaches and attractions with pandemic warnings in mind **Body:** The country has started a most unusual kind of Memorial Day weekend.

7. We also control for *the manner of delivery* of the information, since we only consider posts that contain links to reputable news agencies, and discard content derived from other kinds of user-generated content (*e.g.*, blogs or other social media platforms).

8. The corpus is created from publicly available data (in particular, based on the open dataset created for research by Banda et al. [10]). As such, it is not hindered by any of the *pragmatic concerns* cited by Rubin et al. [94].

9. *Language and culture* are important factors affecting any NLP-based research, of course. Thus, we use only English-language Tweets in this work (although the approach can be applied to other languages, subject to availability of adequate volume of data in that language).

Chapter 4

Data Preparation

In this section, we provide the details of the primary Twitter dataset used as the starting point of our pipeline, the data filtering steps to retain only relevant posts, the preprocessing done to clean the natural language data on which we conduct the classification experiments, and our own additional data collection of newswire articles.

Our pipeline begins by leveraging a large open dataset of Tweets related to COVID-19, developed and made available by Banda et al. [10]. This is a continually growing collection, and at the time of this work, it offered 46.86 million Tweets collected from March through May 2020. We inject additional filtering and data cleaning steps to it, however, which are discussed next.

Table 4.1: COVID-19 keywords. The 52 keywords used to filter out Tweets.

Keywords related to the COVID-19 pandemic
case, CDC, China, corona, covid, crisis, die, disease, distancing, drug, economy, emergency, Fauci, global, government, hands, health, hospital, immune, infected, kill, lab, lockdown, mask, medical, medicine, news, NHS, nursing, outbreak, pandemic, panic, patient, prevent, public, quarantine, recovery, restrictions, risk, safe, sick, social, spread, stock, symptoms, test, treatment, vaccine, virus, wash, watching, Wuhan

4.1 Data filtering

Even though this Twitter dataset is related to COVID-19, it is not immediately suitable for the natural language processing tasks in our work. We have the following conditions to filter a significant part of this dataset:

Retweets A Retweet is a re-posting of a Tweet, intended to facilitate quick sharing and re-transmission of information in the network. The original large dataset includes Retweets, which are often derived content, but with no additional information or commentary. While this may be

useful for analyses of information propagation in a network, it is not useful for our study. Thus, we remove all Retweets.

Non-English Tweets As we discussed earlier in Chapter 3, controlling for language is an important requirement [94]. The dataset, however, includes Tweets from five different languages. We therefore insert a step to filter out non-English entries.[3]

Tweets not containing topic-specific keywords Compared to the original dataset, we impose a stricter condition to establish relevance of each post to the COVID-19 pandemic. We do this by using a set of 52 keywords, and retain only those Tweets that contain at least one of these keywords. This set, shown in Table 4.1, was created by removing all function words[4] as provided by the English-language list of function words in the Python Natural Language Toolkit (NLTK) [13], sorting the remaining words by frequency, and then manually selecting from the most frequent entries. The Tweets collected by Banda et al. [10] include responses to other posts. Often, a response by itself has no content relevant to COVID-19, even if it were relevant in the context of the original Tweet. Most common examples include emotive expressions of sorrow, faith, hope, anger, or sarcasm.

Tweets without a link to a news agency of repute Our work focuses on identifying instances where the original content (the cited news article) belies that claim made in the derived content (the Tweet). Thus, we further restrict our attention to Tweets that include a link to a news article. To this end, we check whether the external link from a Tweet is to a top English-language news

[3]Given the ID of a Tweet, the Twitter API allows for the retrieval of many of its properties, a process known as *hydration*. A hydrated Tweet has several attributes, including one that specifies its language. We use the value of this attribute to determine if it is in English.

[4]Function words are words that play an important role in syntactic correctness of a sentence, but offer little semantic content. They consist mainly of determiners, pronouns, prepositions, and conjunctions. For example, “the”, “and”, “his”, “she”, “although”.

Table 4.2: List of news agencies used as original content. News agencies in the top-50 English-language news sources, as ranked by Alexa Website Ranking. In this work, we remove some domains from the original list due to paywall models, difficulty of data crawling, or topic/genre-specificity (*e.g.*, weather news). The remaining 27 domains are shown here.

List of new agencies we verified Tweets
reuters.com, theguardian.com, wsj.com, washingtonpost.com, nytimes.com, cnn.com, cnbc.com, cbsnews.com, nypost.com, foxnews.com, usatoday.com, theatlantic.com, sfgate.com, latimes.com, hollywoodreporter.com, bbc.com, thehill.com, chicagotribune.com, usnews.com, thedailybeast.com, chron.com, time.com, nbcnews.com, bbc.co.uk, dw.com, variety.com, euronews.com

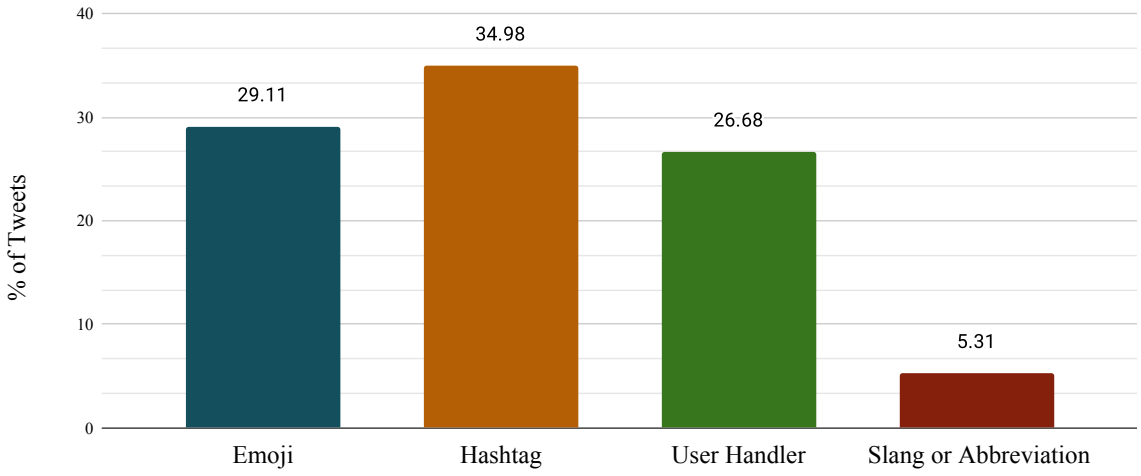


Figure 4.1: Extent of informal register usage. Percentage of different parameters to total Tweets remaining after all filtering steps.

website in the Alexa website ranking[5]. Table 4.2 shows the list of these news agency domains. Tweets with no external link to one of these domains are removed from our study.

4.2 Data preprocessing

After applying the filters described above, we retain over 246k Tweets, and prepare them for the subsequent NLP components of our pipeline by adding a few preprocessing steps. Some of these are standard domain-nonspecific practice in NLP research, while the others are particularly meant for the social media landscape.

[5]https://www.alexa.com/topsites/category/Top/News (this service was last available on Sep 17, 2020)

First, we remove non-linguistic tokens (*i.e.*, non-words) in each Tweet. This comprises a removal of punctuation, URLs, and Twitter user handles. Links to the relevant news agencies (shown in Table 4.2) are decoupled from the post and maintained separately. Twitter extensively uses hashtags too. We remove the hash symbol, but retain the term. For example, "#quarantine" and "#staysafe" are converted to "quarantine" and "staysafe", respectively.

Social media users frequently depart from dictionary-based lexicon and make ample use of informal register. Most commonly, this includes emojis and colloquial non-standard abbreviations and misspellings that have become socially accepted. One may argue that emojis convey information (albeit not in the traditional linguistic sense) and thus, removing them alters the information content in a post. We therefore use the `demoji` library[6] to replace each emoji with its corresponding text form. Abbreviations, especially if non-standard, are seldom handled well by readily available NLP tools (*e.g.*, a syntactic parser), and may not even have a meaningful representation in language models unless the model was trained on large amounts of data containing these tokens. The same holds true for misspellings that have recently gained social acceptance on a platform. Therefore, we use a list of more than 5,700 such terms[7] and replace them with their formal register counterparts. This results in abbreviations like "wru" being converted to "where are you", and misspellings such as "wutevr" being replaced by "whatever". Finally, we observe that some Tweets are duplicated in the dataset, so we remove the spurious copies and retain only one.

4.3 Newswire data collection

As mentioned earlier, this work investigates whether original claims found in news articles are faithfully reproduced in a Tweet. This is the reason behind discarding Tweets that do not contain a link to a news agency of repute (see Section 4.1). The data obtained from Banda et al. [10] do not contain this external information, however. Therefore, we collect the newswire articles linked

[6] Available at pypi.org/project/demoji

[7] Gathered from www.noslang.com/dictionary.

from the Tweets. For this data collection, we use the `Newspaper3k` library[8]. Some articles could not be collected due to paywall restrictions, leading to a final corpus of $46,117$ Tweets together with $23,841$ unique newswire articles from the 27 news agency domains shown in Table 4.2. The number of unique articles is understandably lower, since multiple Tweets often propagate the same article published by widely known news agencies.

For each newswire article, we retain full text of the article, as well as the headline. Any images, videos, and metadata information (*e.g.*, authors, date of publication) are discarded. Subsequently, the articles are tokenized and split into individual sentences using the Python Natural Language Toolkit (NLTK) [13].

[8] github.com/codelucas/newspaper

Chapter 5

Task 1: Identification of Check-worthy Tweets

After all the filtering and data cleaning steps have been taken, the first component of our pipeline is the identification and retention of check-worthy Tweets (as shown earlier in Figure 3.1). This is a precursor to the final objective, because social media posts do not always contain check-worthy factual claims. It thus behooves us to decouple this task from the final analysis of faithful representation and propagation of information in social media. The task itself is designed as a supervised binary classification, where each Tweet is given one of two possible labels: *check-worthy* (CW), or *not check-worthy* (NCW).

Classical supervised learning consists of training followed by evaluation on a test dataset. With the advent of Transformer-based deep learning models [108], however, supervised learning in NLP research is now often divided into (i) the use of embeddings that have been pre-trained on a large corpus, thus yielding a pre-trained language model, and (ii) tuning the embedded representations for a specific task. This is the approach we adopt in our work as well. To this end, we experiment with multiple pre-trained language models, tuning each model in task-specific ways. In the remainder of this section, we first present a short discussion of the pre-trained language models, followed by the datasets on which they are further tuned, before discussing the results.

5.1 Pre-trained language models

We use ten language models pre-trained on general data, plus two domain-specific pre-trained models. These models all rely on the Transformer-based learning of contextual word representations, known as *Bidirectional Encoder Representations from Transformers* (BERT) [26]. BERT is pre-trained on two NLP tasks, *viz.*, masked language modeling – where some input tokens are replaced with [MASK] and the model is trained to reconstruct the original tokens, and next sentence prediction – where the model is trained to understand whether or not one sentence can logically come after another. There are two variants of this model, BERT-Base and BERT-Large,

which differ in the size of the network used for training (see Devlin et al. [26] for details). BERT demonstrated state-of-the-art performance on multiple downstream natural language understanding (NLU) tasks on benchmark datasets, and inspired variations of the original model. These include

1. DistilBERT [95], which pre-trains a smaller general-purpose language model while providing comparable performance on the NLU benchmarks.

2. RoBERTa [65], which discards the next sentence understanding task from pre-training, but uses additional corpora. While the original BERT was pre-trained on approximately 16 GB of unlabeled plain text data, RoBERTa used over 160 GB and achieved improved performance on several NLU benchmarks.

3. COVID-Twitter-BERT [75], two BERT models pre-trained on Tweets related to COVID-19 – CT-BERT-v1 and CT-BERT-v2, the latter pre-trained on a much larger collection of 97 million Tweets.

A closely related model is ELECTRA [18], which is Transformer-based, but instead of the generative approach of BERT's masked language modeling, uses a discriminative approach where some input tokens are intentionally replaced. The model is then trained to identify the replaced tokens. When pre-trained using comparable amounts of data and similar model sizes, ELECTRA outperforms the original BERT models on various NLU benchmarks.

Yet another set of state-of-the-art NLU results were achieved by XLNet [113], which uses a generalized autoregressive pre-training to capture bidirectionality in a token's linguistic context (in contrast to BERT, which uses denoising autoencoder to capture bidirectionality). Like BERT, it is a Transformer-based model, but it uses Transformer-XL [23] to overcome the restrictions of the basic Transformer models (*e.g.*, fixed-length context).

As pre-trained models, we use the multiple versions of BERT, DistilBERT, RoBERTa, CT-BERT, ELECTRA, and XLNet, giving us 12 models altogether. These are tuned on datasets specific to our first task, as discussed next.

Table 5.1: Summary statistics of the three collections used for supervised learning in Task 1.

	Dataset	Size			Description
		CW	NCW	Total	
DS1	Barrón-Cedeño et al. [2020]	231 (34.40%)	441 (65.60%)	672	COVID-19 Tweets
DS2	Hassan et al. [2017a]	5,413 (24.06%)	17,088 (75.94%)	22,501	U.S. Presidential debates
DS3	*This report* [2021]	55 (55.00%)	45 (45.00%)	100	COVID-19 Tweets

5.2 Ground-truth data for model tuning

Prior research on identification of fake news, while different from the investigation in this work, provides several noteworthy datasets that can be leveraged for supervised learning in this first task in our pipeline. In particular, we use three corpora under the monikers DS1, DS2, and DS3. Their basic statistics are shown in Table 5.1.

DS1: As the amount of information available on the Internet grew, so did the amount of false information. Realizing that human participation in fact-checking is likely to remain necessary in the foreseeable future, Barrón-Cedeño et al. [11] designed a shared task for fact-checking in social media, where the first step was to rank information nuggets based on their "check-worthiness". The dataset does, however, provide binary ground-truth labels for check-worthiness, and can thus be directly used for supervision in our task.

DS2: The second dataset we use to supervise our classifiers is the well-known *ClaimBuster* corpus [43]. This collection provides three ground-truth labels for each datum: (i) check-worthy factual sentences, which present a factual claim whose authenticity is of interest to the general public, (ii) unimportant factual sentences, which contain factual claims but the claims are deemed to be not of interest to the general public, and (iii) non-factual sentences, which do not contain factual claims but instead consist of opinions, beliefs, questions or other subjective content. In this work, we use the first category as CW and coalesce the remaining two into NCW.

DS3: We manually annotate 100 randomly selected Tweets from the corpus created based on the dataset available from [10]. Three annotators carry out this task, and thus, each Tweet was

Table 5.2: Performance on Task 1: Identification of check-worthy Tweets. The classification results on 12 models, each fine-tuned on DS1, DS2, and both. The evaluation is done on DS3, showing the **P**recision, **R**ecall, $\mathbf{F}_1$ score, and the number of true positives (**TP**) out of the 55 check-worthy elements in DS3. Models considered as candidates for providing input to our second task are marked by ‡. XLNet-Base, shown in bold, is the pre-trained model that achieves (upon fine-tuning) the highest precision among the candidates.

Model	**DS1**				**DS2**				**DS2 + DS1**			
	P	**R**	$\mathbf{F}_1$	**TP**	**P**	**R**	$\mathbf{F}_1$	**TP**	**P**	**R**	$\mathbf{F}_1$	**TP**
BERT-Base	57.6	89.1	70.0	49	86.5	58.2	69.6	32	86.8	60.0	71.0	33
BERT-Large	57.3	100	72.8	55	90.9	36.4	51.9	20	82.4	50.9	62.9	28
RoBERTa-Base	55.6	100	71.4	55	77.8	76.4	77.1‡	42	79.6	70.9	75.0‡	39
RoBERTa-Large	55.6	100	71.4	55	79.6	70.9	75.0‡	39	80.0	58.2	67.4	32
DistilBERT-Base	69.7	41.8	52.3	23	75.9	80.0	77.9‡	44	77.2	80.0	78.6‡	44
CT-BERT-v1	57.8	94.5	71.7	31	84.1	67.3	74.7	37	78.0	38.0	51.0	39
CT-BERT-v2	68.4	47.3	55.9	26	85.7	10.9	19.4	6	79.3	41.8	54.8	23
Electra-Base	56.4	96.4	71.1	53	88.5	41.8	56.8	23	85.7	43.6	57.8	24
Electra-Small	57.5	76.4	65.6	42	70.2	60.0	64.7	33	71.0	62.1	66.3	22
Electra-Large	62.2	92.7	74.5	51	80.0	43.6	56.5	24	81.6	56.4	66.7	31
XLNet-Base	87.8	65.5	75.0‡	19	88.0	64.5	74.4	36	84.4	69.1	76.0‡	38
XLNet-Large	58.1	65.5	61.5	36	84.4	49.1	62.1	27	78.4	72.7	75.5‡	40

assigned a CW or NCW label by each annotator independently. To measure the consensus on check-worthiness, we use Fleiss' kappa [30] – a measure of inter-rater reliability, but unlike the more commonly used Cohen's kappa, this can be applied in scenarios with more than two raters. We achieve $\kappa = 0.822$, indicating that the annotators are in near-perfect agreement [93]. There were disagreements only on 13 Tweets, where one of three annotators disagreed with the other two. In these cases, we used majority voting to assign the final label.

5.3 Experiments and results

Our experiments for the first task are categorized based on the pre-trained model, and the corpus on which that model was tuned. Thus, each experiment can be represented as a ⟨model,dataset⟩ pair. We conduct three sets of experiments, where each model is tuned (i) on the COVID-19 Tweets corpus (DS1), (ii) on ClaimBuster (DS2), and (iii) on both corpora, tuning first on ClaimBuster and then on COVID-19 Tweets (DS2+DS1). We then evaluate each ⟨model,dataset⟩ pair on the manually annotated sample, DS3. The results are shown above in Table 5.2.

Since this first task in our pipeline is meant to feed check-worthy Tweets as input to the second task, the immediate and natural step is to select the "best" tuned model. Unfortunately, no single ⟨model,dataset⟩ pair achieves a clearly superior performance across the three standard metrics of precision, recall, and F_1 score. As lower precision means a greater number of falsely labeled check-worthy (CW) Tweets will enter the second task, it is clear that we need to prioritize a high-precision model even at the expense of potentially lower recall. However, extremely low recall will quite likely cause the next tasks to receive inadequate amount of input data, and therefore, build a less robust model. We thus use a threshold F_1 score of 75 to remove some models from further consideration. Among the remaining (shown in Table 5.2 with ‡), ⟨XLNet-Base, DS1⟩ and ⟨XLNet-Base, DS2+DS1⟩ achieve the best precision. However, due to the extremely low recall of the former, we move forward to the next tasks with XLNet-Base model tuned on DS2+DS1 as our choice.

Chapter 6

Task 2: Claim Sequence Labeling

Extracting factual claims is essential for the accurate analysis of texts in many tasks and is an important yet challenging process. A claim is defined as an assertion that is disputed and that we try to back up with some reasoning [37]. In other words, an argument's central component is its claims [16] but there is no widely accepted procedure for defining what constitutes a claim and even human beings do not have a substantial agreement on determining the claim part of a sentence. An example of a claim detection system that has gained widespread attention is ClaimBuster [44]. Their study included a large collection of annotated debates on televised American events. They use a Support Vector Machine (SVM) [46] classifier to combine Term Frequency-Inverse Document Frequency (TF-IDF), Part-of-Speech (POS) tags [110], and Named Entity Recognition (NER) [73] features to assess the importance of a claim, a statement which is of interest to the public, for fact-checking. However, in this work, the goal is to define exactly which parts of a sentence qualify as claims rather than assigning a binary label of whether the sentence contains a claim or not. For this, we use a sequence labeling approach.

He et al. [45] discussed the latest deep learning techniques applied to sequence labeling. They reviewed three common sequence labeling tasks, NER, POS, and text chunking, different machine learning approaches which are used for these tasks, as well as the deep learning methods. They have divided the deep learning approaches into three parts: (I) *Embedding Module* that converts words into their vector representations, (II) *Context Encoder Module* that is used for contextual features extraction, and (III) *Inference Module* that predicts the labels and produces the optimum label sequence as the model's output. As a baseline for most subsequent POS work, the Bi-LSTM-CNN-CRF model proposed by Ma and Hovy [66] has been used. They have used GloVe [81] as their word-level and a Convolutional Neural Network (CNN) [57] as their character-level embedding module, Bidirectional Long Short-Term Memory (Bi-LSTM) as their context encode, and CRF as their inference module. This work represents the first end-to-end model for sequence label-

ing without the need for feature engineering or preprocessing of the data. Their model achieved an accuracy is 97.55%. For the NER task, Li et al. [61] adopted novel techniques, such as local context reconstruction and delexicalized entity identification to develop a model that prioritizes rare entities; their model achieves an average F1-score of 92.67%. In the text chunking task, the model proposed by Liu et al. [63] outperforms the other models with an F1-score of 97.3% by leveraging pre-trained language models. They used an innovative hierarchical neural model and introduced a sentence-level representation into the embedding module to capture global information at the sentence level. Overall, He et al. [45] concluded that models taking advantage of external resources performed better on all three tasks, especially language models pre-trained on an unlabeled text corpus. These models, however, require a larger neural network with more processing power and a longer training time.

Sequence labeling is a fundamental research subject that encompasses a wide range of activities such as Part-Of-Speech (POS) tagging, Named Entity Recognition (NER), Text Chunking, and many more. Despite its widespread use and effectiveness in many downstream applications (*e.g.*, information retrieval, question answering, machine translation, knowledge graph embedding, *etc.*), traditional sequence labeling systems rely largely on hand-crafted or language-specific features. However, Deep Learning has recently been employed for sequence labeling tasks due to its remarkable ability to automatically learn complex features of inputs and to deliver state-of-the-art results [45].

One of the classical sequence labeling tasks is Name entity Recognition, also known as Named Entity Identification or Entity Chunking. The purpose of NER is to automatically identify named entities in text that usually carry important information and classify them into predefined categories [73], such as person, organization, location, time, date, currency, percentage, *etc.*

Among the most common tagging formats used in computational linguistics to tag tokens in sequence labeling tasks is the IOB format [88], short for ***I**nside, **O**utside, **B**eginning*. An O tag suggests that a word belongs to no pre-defined category. The B- prefix before a tag indicates that

the tagged word is the beginning an entity. The prefix I- signifies that a tagged word belongs to the inside an entity.

The tag that is assigned to a token consists of two parts: position of the token in the entity followed by the category within the predefined taxonomy that this entity belongs to. for example, *B-LOC* means **B**eginning of a **LOC**ATION entity or *I-PER* indicates **I**nside a **PER**SON entity.

Text Chunking process, also known as Shallow Parsing, involves the extraction of phrases from unstructured text. This means analyzing the sentence to identify the constituents (Noun phrase (NP), Verb phrase (VP), Adjective phrase (ADJP), Adverb phrase (ADVP), and Prepositional phrase (PP)) present in it. However, neither their internal structure nor their role within the main sentence is specified by chunking. Basically, the chunking task takes the POS tags as input and outputs the corresponding chunks. IOB format is used for chunk tags as well. For instance, *B-VP* tag means that the tagged token is the beginning of a verb phrase.

The following discusses the various steps we took to accomplish this objective.

6.1 Sequence labeling for claim extraction

The underlying assumption is that a claim is made up of a coherent and continuous series of tokens. In other words, we assume that a claim cannot be made up of multiple sections with non-claim tokens in between. Thus, similarly to the tasks discussed earlier, the process of claim extraction involves the identification of a sequence of tokens that form a claim within a given text and then attempt to identify these tokens.

We also use the IOB format to tag the tokens and prepare our datasets. Hence, we have three different labels:

1. **B-CLAIM:** marks the beginning of the sequence of the claim,
2. **I-CLAIM:** tags the rest of the tokens in the claim sequence other than the first one, and
3. **O:** is assigned to the non-claim tokens.

Consider the Tweet in Figure 6.1. This Tweet consists of three parts and only one of them constitutes a factual claim. Table 6.1 shows the list of tokens in this Tweet along with their corresponding IOB tags.

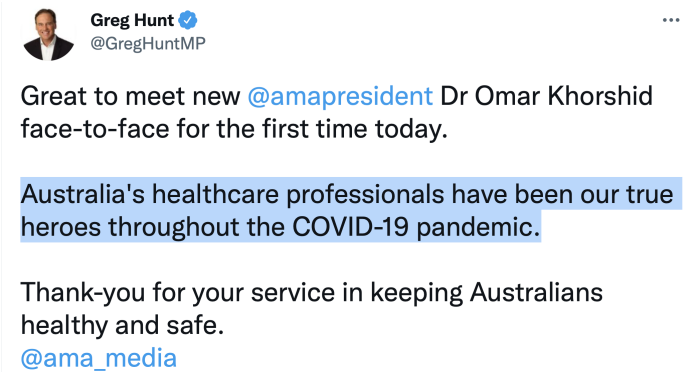


Figure 6.1: A sample Tweet with a factual claim. A claim can sometimes be formed only by a particular part of the Tweet text. In this case, the highlighted part is a claim.
(Source: https://twitter.com/GregHuntMP/status/1332193744377016320)

Table 6.1: Tweet tokens are tagged to specify the claim part.

Token	Tag	Token	Tag
Great	O	our	I-CLAIM
to	O	true	I-CLAIM
meet	O	heroes	I-CLAIM
new	O	throughout	I-CLAIM
@amapresident	O	the	I-CLAIM
Dr	O	COVID-19	I-CLAIM
Omar	O	pandemic	I-CLAIM
Khorshid	O	.	I-CLAIM
face-to-face	O	Thank-you	O
for	O	for	O
the	O	your	O
first	O	service	O
time	O	in	O
today	O	keeping	O
.	O	Australians	O
Australia's	B-CLAIM	healthy	O
healthcare	I-CLAIM	and	O
professionals	I-CLAIM	safe	O
have	I-CLAIM	.	O
been	I-CLAIM	@ama_media	O

Unlike the Tweet in Figure 6.1, the entire text of some Tweets appears as a single claim. In this case, we place these Tweets in a category named **"All Claim"**, meaning that there are no non-claim tokens in them, and the rest of the Tweets with a claim in a **"Partial Claim"** class, meaning that just a portion of the Tweet text is considered as a claim.

6.2 Tweet collection

In this section, we provide details of our primary Twitter dataset. This dataset is the starting point for our pipeline of tasks in the next steps.

Starting from September 2020, in the early months of the COVID-19 pandemic, we used Twitter streaming API to download COVID-19-related Tweets. By March 2022, our dataset has more than 600M Tweets and 4.6TB of Tweet content and metadata. After dedpulication, we were left with 192M Tweets.

We are collecting the dataset based on a list of COVID-19-related keywords. This list includes 42 keywords, shown in Table4.1, that are directly (*coronavirus* and *corona*) or indirectly (*workfromhome* and *washyourhands*) related to the topic. We only kept the English Tweets that have at least one of the keywords in Table 4.1. Figure 6.2 shows the average number of English Tweets per day for each month over a 13-month period with a maximum of about 719K Tweets per day in December 2020.

Due to the sheer volume of the dataset, we created a sample dataset with more than 4.7 million Tweets and used it in further experiments. This sample is 1% of Tweets of each day selected randomly. Figure 6.3 shows the number of Tweets, in thousands, containing each keyword in total in the sample dataset. It only shows keywords with at least 40K Tweets.

6.3 Manual annotation

One limitation of supervised models in Machine Learning and Deep Learning is the requirement for large labeled datasets related to the target task. There are available train sets for some popular tasks, however, for many others, there is no off-the-shelf data to be used. One alternative

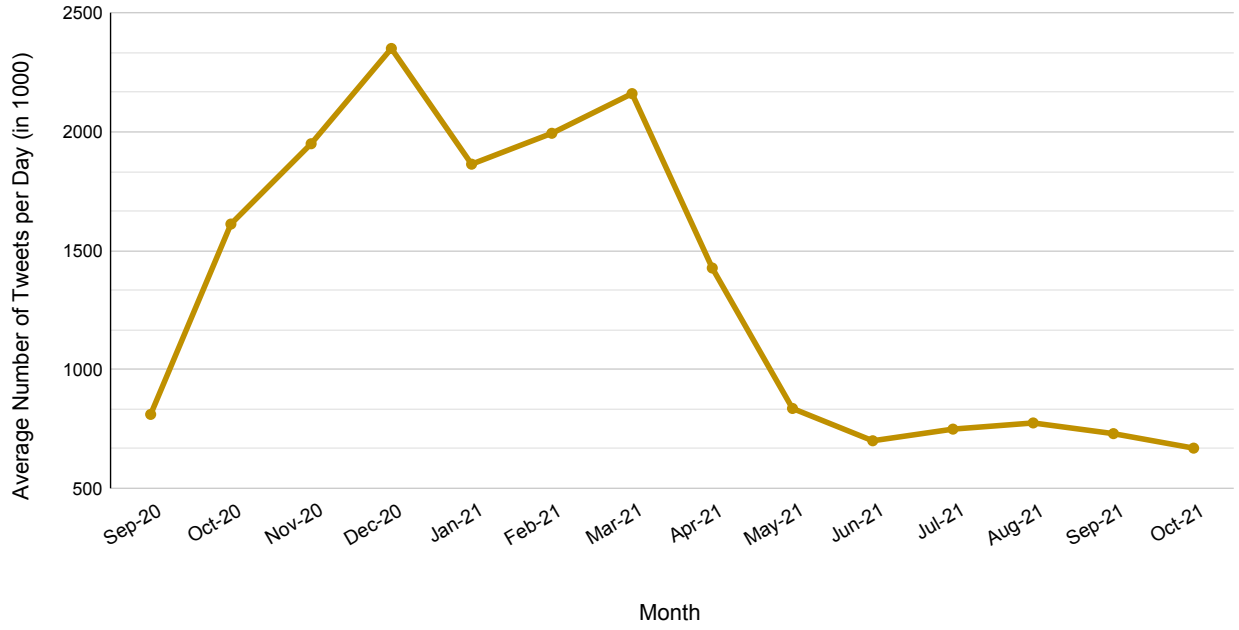


Figure 6.2: Average #Tweets/month. The average number of Tweets in our dataset per day for 13 month.

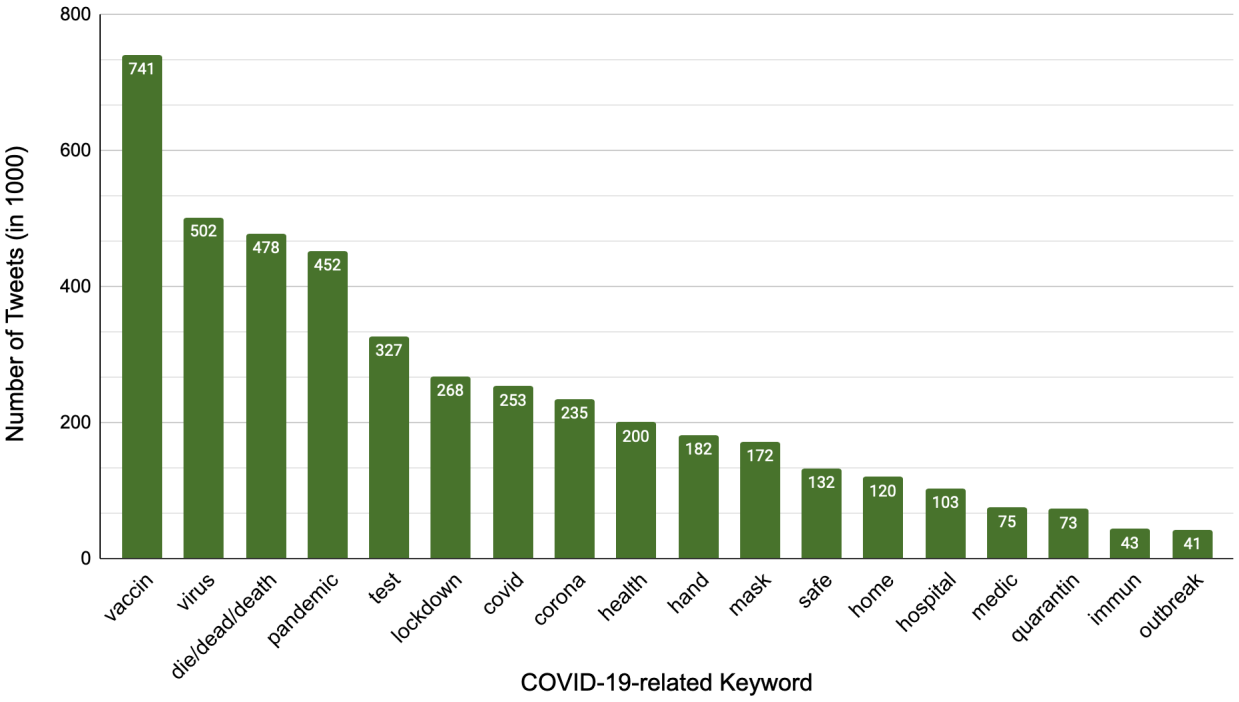


Figure 6.3: Number of Tweets (divided by one thousand) per keyword in the sample data of size 4.7M Tweets.

solution is to manually annotate data. Despite the fact that the process is slow and expensive, if experts are hired to annotate the data, we might end up with high-quality data.

As we do not have a gold standard training dataset for claim extraction, we gathered a team of 12 students studying the field of Computer Science to annotate Tweets for us. For each given Tweet, 2 students were supposed to define which part of the Tweet constitutes a claim. Should there be any conflict between the annotations provided by the two primary annotators for each Tweet, a third annotator will act as a judge and resolves the conflict.

6.3.1 Tweet annotation website

We designed and developed a website to facilitate and speed up the manual annotation process. Each Tweet is displayed to two different annotators. They decide if any part of the Tweet falls under the purview of a factual claim. In addition, there is a section for identifying the stance of the Tweet's author that is not used here. When they are uncertain about a Tweet, annotators have the option to skip it. Figure 6.4 shows a preview of the website.

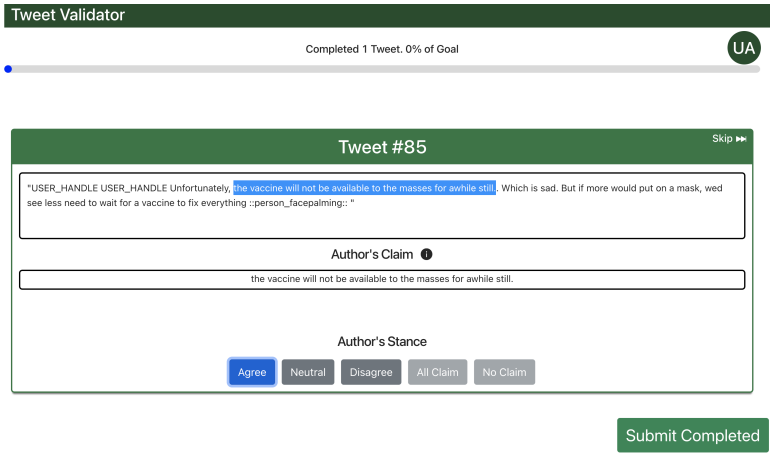


Figure 6.4: Tweet annotation website. Using this website, annotators can easily determine the claim part by clicking and dragging over the text to highlight the claim. There are also two options: All Claim and No Claim for cases when the whole Tweet is a claim or the Tweet does not have a factual claim.

The annotators have a unique ID each to log in. We can set a goal for each annotator and there is a progress bar that shows them how many Tweets they have annotated so far. Export options are available for both annotated Tweets and skipped Tweets.

We keep the following features and content for each annotated Tweet:

- Tweet ID, Tweet Text
- 1st Annotator ID, Claim 1, Stance 1, Validate Time 1
- 2nd Annotator ID, Claim 2, Stance 2, Validate Time 2

The website prioritizes tweets that have already been annotated by one annotator to show them to a second annotator, if there are any, otherwise it displays a new tweet. Eventually, for each fully annotated Tweet, if Claim 1 and Claim 2 match, the Tweet can be added to the train set. Otherwise, a third annotator acts as a judge, choosing either Claim 1 or Claim 2 or defining a new Claim 3.

6.4 Data augmentation

As the name suggests, Data Augmentation (DA) is a term used to describe how to increase the variety of training examples without actually gathering new data. Many strategies include the addition of slightly modified copies of existing data or the creation of synthetic data, with the purpose of enhancing ML models and reducing overfitting as a result by acting as a regularizer [47, 99]. In recent machine learning (ML) research, it has received active attention in the form of well-received, general-purpose techniques such as UDA [112], which used backtranslation [96] AutoAugment [21], and RandAugment [22], and MIXUP [38, 115].

These are frequently explored first in computer vision (CV), where techniques like cropping, flipping, and color jittering are a standard component of model training. DA's adaptation for Natural Language Processing (NLP) appears to be secondary and underexplored, perhaps due to the challenges posed by the discrete nature of language, which eliminates continuous noising and makes maintaining invariance more difficult [29]. Despite these obstacles, there has been a surge in interest and demand for NLP DA. There are more tasks and domains to study as NLP increases owing to the availability of huge pre-trained models off-the-shelf. Many of them are low-resource

and lack training examples, resulting in a plethora of use cases in which DA might be useful. DA research is particularly scarce for many non-classification NLP tasks, including span-based tasks and generation, despite their prevalence in real-world environments.

Many different methods are available for augmenting text data. Below are a few of the more popular techniques for easy data augmentation.

1. **Back Translation (BT):** In this method, we translate the text into a second language and then translate it back to the original language [28]. For example, our English tweets can be translated into Spanish and then back into the English language. This method can lead to erroneous copies of the data, especially when semantic features are important since the meaning might change when a complex text is translated twice.

2. **Synonym Replacement:** This technique picks a few words from the given sentence at random that are not stop words. Then for each of those words, it replaces the word with one of its synonyms selected randomly.

3. **Substitution:** This method substitutes some random words in the sentence with other words. As opposed to synonym replacement, this method often avoids using strings that are semantically close to the original data. One approach is substituting words with their misspelled version [19, 89].

4. **Swapping:** Despite the importance of the sequence of words within a sentence, a few words swapped within a small and reasonable range can still convey the same meaning [59]. This opens up the possibility to swap random, non-stop words that are relatively close to one another (considering their position in the sequence of the words in the sentence) for the purpose of augmenting the text.

5. **Insertion:** Means that the sentence is altered by inserting a random word. It is common to choose a non-stop word, find a synonym for it, and then insert that synonym at a random position in the sentence [60].

6. **Deletion:** In this technique, a random part of the given text is deleted. If we have sentences, then a few random words get deleted but if we have documents, usually random sentences are deleted [59].

Because the annotation process is fairly slow, we ended up with 1315 annotated Tweets after six months, including 196 high-impact Tweets (set 1) and 1119 other Tweets (set 2). We consider high-impact Tweets to be those from verified Twitter users who have more than 100K followers. For the Tweets in set 2, there is no filter on the user's verification or the number of followers. Table 6.2 shows the number of Tweets in each set.

Table 6.2: Number of test Tweets per each category of claim in set 1 and set 2.

	All Claim	**Partial Claim**	**No Claim**	**Total**
Set 1 (High-impact Tweets)	74	29	93	196
Set 2 (All Tweets)	45	624	450	1119

We expanded the size of our train set by augmenting the data. The three methods we employed were insertion, substitution, and synonym replacement. We used the BERT model, RoBERTa model, and DistilBERT model for insertion and substitution. A total of 7900 new samples were added to the train set via augmentation.

6.5 Pre-trained word embeddings

The first step of this task is to convert text to numerical representations. We have combined different types of pre-trained word embeddings for this purpose:

1. **Classic Word Embeddings:** In classic word embeddings, a static pre-computed embedding is assigned to each word which means the embeddings are static and word-level. Examples are GLoVe [81] and Word2Vec. For our experiments, we have used the WordEmbeddings class provided by the Flair Library [1] that was initialized with FastText embeddings [14] pre-trained over Wikipedia.

2. **Contextual String Embeddings:** The context-based word embeddings are capable of capturing hidden syntactic and semantic information that extends beyond classical embeddings [2]. This approach forms word-level embeddings based on character-level language modeling and its use is particularly advantageous when the NER task is approached as a sequential labeling problem. For our experiments, we have used the FlairEmbeddings class [3] for the English language provided by the Flair Library.

3. **Transformers-based Word Embeddings:** Bidirectional Encoder Representations from Transformers (BERT) [26] is based on a multi-layer bidirectional transformer-encoder, where the transformer neural network uses parallel attention layers rather than sequential recurrence [108]. These kinds of embeddings are commonly pre-trained over very large corpora to capture latent syntactic and semantic similarities between words.

Another option that the Flair library provides is Stacked Embedding which allows us to combine different embeddings. With this approach, we will have a new vector representation for each word that is the concatenation of different embeddings.

The next step is classifying the tokens into the pre-defined classes we have: B-CLAIM, I-CLAIM, and O. We use the sequence tagger model that the Flair library provides. We choose to use the Conditional Random Field (CRF) as the classification head and train it using our custom data.

6.6 Results and discussion

We keep the high-impact Tweets (set 1) as well as 111 annotated Tweets from set 2, including 43 Tweets with no claim, 4 Tweets that are all claim, and 64 Tweets that a part of them is considered to be a claim, for testing. The rest of the Tweets along with their augmented versions are used for training the sequence labeling model. To evaluate the performance, we analyze the predicted labels for each Tweet separately. This means that, if we consider labels 'B-Claim' and 'I-Claim' to be positive and label 'O' to be negative, we calculate the number of True Positives (TP), False Positives (FP), True Negatives (TN), and False Negatives (FN) for the tokens in each Tweet. Then,

we sum up these four numbers for all Tweets to have the overall numbers of TP_{total}, FP_{total}, TN_{total}, and FN_{total}. Now, using these values we calculate different evaluation metrics, such as precision, recall, F1-score, and accuracy. Table 6.3 shows the results of this task. The average F1-scores for test set 1 is 83.14% and for test set 2 is 73.33%.

Table 6.3: Claim extraction results. Evaluation of the claim extraction module using micro-averaged performance metrics (in percentage) for two different Tweet distributions.

Data Distribution	Class	Precision	Recall	F1-Score	Accuracy	Support	
						#Tweets	#Tokens
High-impact Tweets	All Claim	100	78.2	87.77	78.2	74	1445
	Partial Claim	77.23	52.61	62.59	57.82	29	1029
	Average	96.08	73.27	83.14	72.85	103	2474
All Tweets	All Claim	100	75.36	85.95	75.36	4	211
	Partial Claim	83.1	59.6	69.41	75.46	64	2966
	Average	87.2	63.27	73.33	75.44	68	3177

This claim sequence labeling module expects that every Tweet it receives has a claim. However, this module can still handle Tweets without claims pretty accurately. Table 6.4 shows the accuracy scores that our module achieved (72.5% for high-impact Tweets and 91.49% for other Tweets) when the input has no claim.

Table 6.4: Results for Tweets with no claim. Weighted average accuracy (in percentage) of the claim extraction module when the input does not have a claim.

Data Distribution	Accuracy	Support	
		#Tweets	#Tokens
High-impact Tweets	72.5	90	2295
All Tweets	91.49	43	2432

Chapter 7

Task 3: News Verification

Of the $46,117$ Tweets retained after the filtering and preprocessing steps described in Chapter 5, the $\langle$XLNet-Base, DS2+DS1$\rangle$ model (described above in Chapter 5) feeds $39,458$ Tweets into the News Verification component in our pipeline. Here, our goal is to identify whether or not the claim made in a Tweet containing a link to a news article is *actually* supported by the cited article.

The Tweets that reach this task have already been labeled as check-worthy by the best-performing classifier in the previous step. We add another filter, however – removing Tweets that consist of multiple sentences. This is done in order to remove the noise of lengthy posts where one sentence may have a check-worthy factual claim, thus justifying the CW label, but the other sentences may be subjective opinions or expressions of sentiment, sarcasm, humor, etc. Figure 7.1 presents such an example, where a check-worthy factual claim is followed by a possibly sarcastic question posed by the person sharing the piece of information. This filtration reduces the corpus size to $29,392$ Tweets. We keep $11,800$ Tweets for training, $12,335$ for validation and hyperparameter tuning, and $5,257$ for testing.

7.1 Design and setup of experiments

We observe that Tweets are often a near-verbatim reproduction of the news headline. Indeed, approximately 54% of all the Tweets provided as input to our third task fall into this category. The remaining cases, however, require a deeper understanding of the body of the news article to determine if the claim made in the Tweet is supported by the cited article. Thus, we further divide the third task into two steps where we consider (i) only the headline of the cited news article, and (ii) the entire body of the article. The complete flowchart for this task is shown in Figure 7.2.

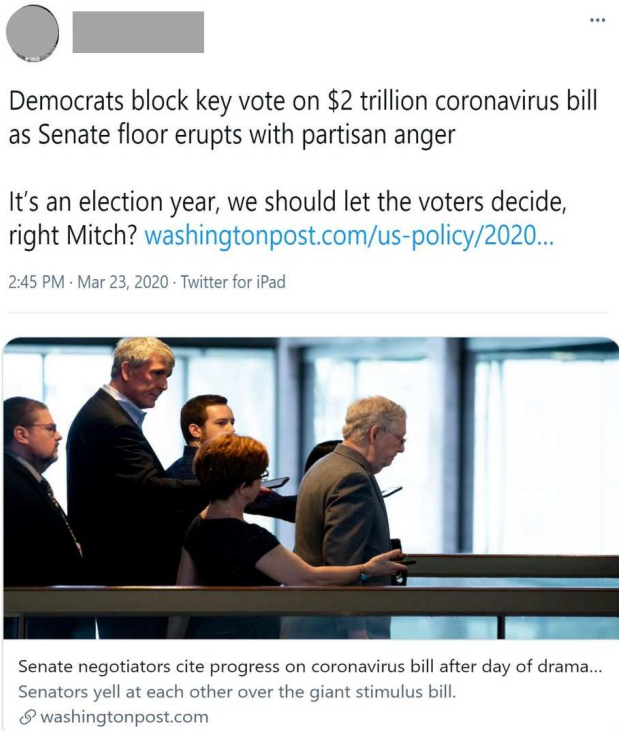


Figure 7.1: A Tweet comprising multiple sentences. The first sentence is objective, and contains a check-worthy factual claim, while the second sentence does not.

7.1.1 Distant supervision

For both steps, the initial challenge is to obtain sufficient labeled data for training any supervised learning algorithm. We address this by employing *distant supervision*, an approach originally motivated by the use of *weakly labeled data* in bioinformatics [20]. In this approach, an assumption is made about the unlabeled data obtained or extracted from a corpus. Its success in learning relations from natural language, for instance, relied on a relation-triple $\langle \text{entity}_1, \text{entity}_2, \text{relation} \rangle$ being obtained from the Freebase corpus, and *assuming* that any sentence mentioning the two entities express their relation in some way [72]. Similarly, the presence of specific emoticons and keywords has been used to obtain large amounts of distantly supervised Tweets for sentiment classification and topic identification [24, 68]. In our work, the assumption made for distant supervision is that if a news article is hyperlinked by a Tweet, then the article supports the claim made in the Tweet. In the absence of such a hyperlink, the $\langle$Tweet, news$\rangle$ pair is marked as unsupported. Our collection, by design, would yield only positive labels according to the above assumption of

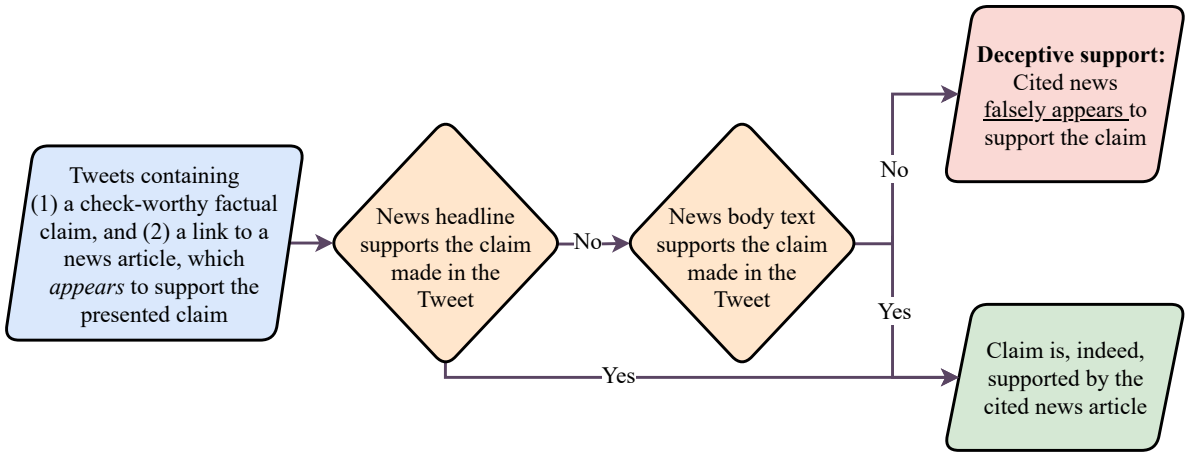


Figure 7.2: Information verification in Task 2. The input comprises Tweets containing check-worthy factual claims that offer a news article as supporting evidence for that claim. The output is a binary decision about whether the support is deceptive.

distant supervision. Thus, all ⟨Tweet, news⟩ pairs in the training set are given the weak label of "supported". We then create ⟨Tweet, news⟩ pairs by coupling each Tweet in the training set with an arbitrary but different headline from the collection of news articles. These pairs are given the weak label of "unsupported", thus forming the negative sample. This strategy of creating negative samples by random pairing has shown promise in prior work on fact-checking [41, 77]. We use this same method to generate positive and negative weak labels for the validation set as well. This weakly labeled corpus of ⟨Tweet, headline⟩ pairs is utilized in the first step (shown in Figure 7.2). For the second step, we build a weakly labeled corpus of ⟨Tweet, article⟩ pairs using the same method, where each Tweet is paired with the entirety (*i.e.*, the headline plus the body) of a news article.

7.1.2 Step 1: Determining support from the cited headline

For this first step, we use five pretrained language models (the base version when applicable): BERT [26], CT-BERT-v2 [75], XLNet [113], RoBERTa [65], and DistilRoBERTa [95]. We described the first four models earlier in Chapter 5. The last model, DistilRoBERTa, is a lighter version of RoBERTa, pretrained on a smaller general-purpose language model. Additionally, we also use DistilRoBERTa trained on a large paraphrase dataset (henceforth denoted by DistilRoBERTa^p),

which has been shown to achieve state-of-the-art performance on multiple tasks on semantic similarity. Our inclusion of this additional model is motivated by prior studies corroborating that a claim and its supporting evidence are bound to have relatively high semantic similarity [6, 74]. All the models are tuned on the ⟨Tweet, headline⟩ weakly labeled collection.

7.1.3 Step 2: Determining support from the cited article's text

When a news article presents a factual claim, there may exist a single sentence in the article from which this claim can be distilled. It is, however, also possible that the claim can only be gleaned from multiple sentences in the article. We thus follow a two-pronged strategy to determine support. On one hand, we split the body of the article into a sequence of sentences, and pair each sentence with the Tweet citing this article. Each such ⟨Tweet, sentence⟩ pair is then provided to the classifiers used in the first step described above (7.1.2), since the data are structurally identical to that used in determining support from the cited headline. If any pair created from the article is labeled as "supported", the ⟨Tweet, article⟩ pair is deemed "supported". Otherwise, it is deemed "unsupported". On the other hand, we also conduct experiments on the ⟨Tweet, article⟩ pairs directly, without any sentence-splitting of the text. The same models are used again, except for DistilRoBERTa^p, which is not designed for long token sequences. To account for longer texts, we use Longformer instead [12], which combines local windowed attention and global attention, thus allowing it to process sequences that are thousands of tokens. Indeed, compared to RoBERTa, it has demonstrated superior performance on long-document tasks.

7.1.4 Technical runtime setup

All our experiments are conducted on NVIDIA Tesla V100 GPUs. We train every model for 1 and 2 epochs, with batch sizes of 16 and 24, and a learning rate set to 5×10^{-5}. For the first step, where only the news headline is paired with the Tweet, we set the maximum sequence length to be 128, and for the second step, we set it to 512. The only exception to this being Longformer, where the maximum sequence length is $4,096$.

7.2 Evaluation, results, and discussion

On the validation set, all models achieve an $F1$ score of nearly 0.98, whether they classified ⟨Tweet, headline⟩ pairs, or ⟨Tweet, article⟩ pairs. Given that our *weak labeling* builds the negative samples by combining a Tweet with a randomly selected different news article, the extremely high score is not unexpected, as discussed by Zuo et al. [119]. A more important point, arguably, concerns the false negatives of these models. In contrast to a standard supervised learning setup, these pairs are only *weakly false* negatives. That is, the Tweet does provide a link to a news article, but the model predicts the claim to be unsupported by the news article's headline. Essentially, all pairs in the test set are labeled as positive or supported according to our assumption for building the train and evaluation datasets. Hence, we will only have false negative or true positive labels. So, the false negative pairs are the most likely candidates where the hyperlink is deceptive, and the news does not actually support the claim being made by the social media post. At the very least, these are the candidates for which the support is not obvious from the news headline alone. Thus, we collect these *weakly false* negative ⟨Tweet, headline⟩ pairs, and feed them to the second step where the entire article is investigated by the classifiers.

7.2.1 Sample annotation

Since this is a downstream task, some errors from the previous component are likely to pass through. Thus, before starting the second step, we analyze these weakly false negative pairs by performing another annotation task. The number of such pairs varies from one model to another, and the first step yields a total of 258 of them among the samples in the test set. Three annotators work independently on this collection, each answering the following:

(1) *Is the given Tweet check-worthy?* The annotators answer this question on the basis of the same guidelines provided to them during the first task.

(2) *If the Tweet is check-worthy, does the cited article support the Tweet?* Each annotator peruses the entire article vis-à-vis the Tweet, and determines whether any information provided in

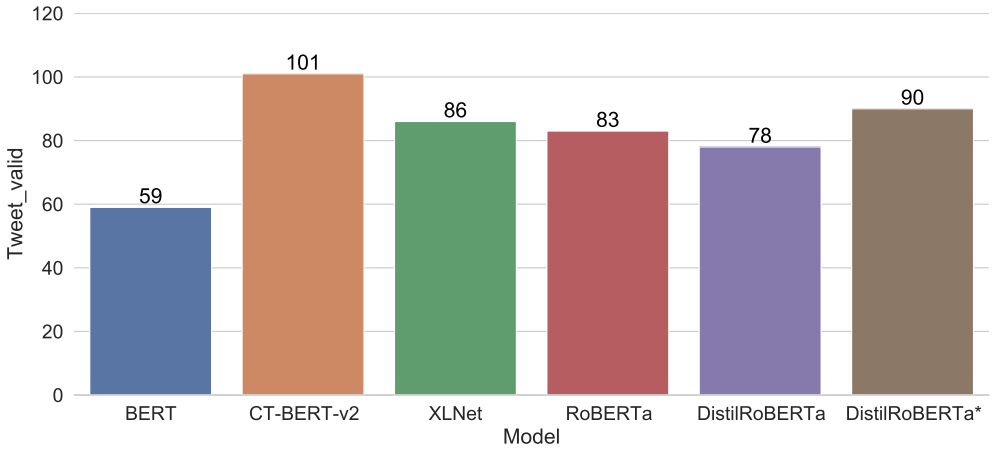


Figure 7.3: Number of *weakly false* negative pairs for each model. These are check-worthy factual claims made in Tweets that link to a news article as a cue of external support, but the model labels them as *unsupported*, based on the ⟨Tweet, headline⟩ pair.

the article supports the claim made in the Tweet. Accordingly, they assign one of two labels to the pair: *supported*, or *unsupported*.

Of the 258 pairs, 51 were labeled as *not check-worthy* by at least two annotators. We discard these from the evaluation of the second step. Further, there were disagreements on 7 other Tweets, which we discard as well. Out of the remaining 200 pairs, 55 were labeled as *unsupported* by at least two annotators. This annotation process showed substantial agreement among the three members, yielding a Fleiss' kappa score of $\kappa = 0.756$. Our inspection finds two main reasons for the disagreements. First, it is due to differing opinions on expressions of causality in human language. For instance, a Tweet announced "Dow drops 200 points as unemployment claims surge once again", while the corresponding news article mentioned the two events "Dow drops" and "unemployment claims surge" in separate paragraphs. For some readers, this is an indication of causality, but no explicit mention of a causal relation between the two. A second reason is a difference among the annotators regarding the inclusion of metadata in the verification process, going beyond the purely linguistic expression of a claim. For example, a Tweet states "Yesterday more than 2K in the US died of coronavirus", where the dates of the post and the news article are, clearly, relevant.

Out of the 200 manual annotations discussed above, 55 are labeled as deceptive (*i.e.*, 27.5%). This, however, is sampled from the test of approximately $5,000$ Tweets. Thus, our test data shows that *at least* 55 out of $5,000$ Tweets (*i.e.*, 1%) contain deceptive hyperlinks. In Figure 7.3, the number of ⟨Tweet, headline⟩ pairs predicted to be *unsupported* by the models are shown after the removal of erroneous samples propagated by Task 1 (*i.e.*, claims that are not check-worthy). Also, Table 3.1 includes three Tweets with deceptive hyperlinks, each citing a news article from a well-known news agency. However, the news article doesn't support the Tweet, as shown with examples (5, 6) in Table 3.1, or is even irrelevant (see Table 3.1 example (7)).

7.2.2 Evaluation and discussion

The performance of each model is evaluated on the 200 annotated pairs, with the annotation labels serving as the ground-truth. For both steps of Task 2, we measure the performances using macro-average precision, recall, and F_1 score. Given the class imbalance, where only a minority of the samples offer deceptive support to the reader, macro-average associates more value to the minority class by disregarding the overwhelming effect of the majority class. For step 2, we provide two ways of evaluating each model:

(1) First, we feed all 200 annotated samples into Step 2. That is, the entirety of the news articles are checked by the sentence-level models tuned on ⟨Tweet, headline⟩ pairs, as well as the article-level models tuned on ⟨Tweet, article⟩ pairs. This evaluation is effectively an ablation study to understand how well our system can detect deceptive cues of support, in the absence of a separate first step in Task 2.

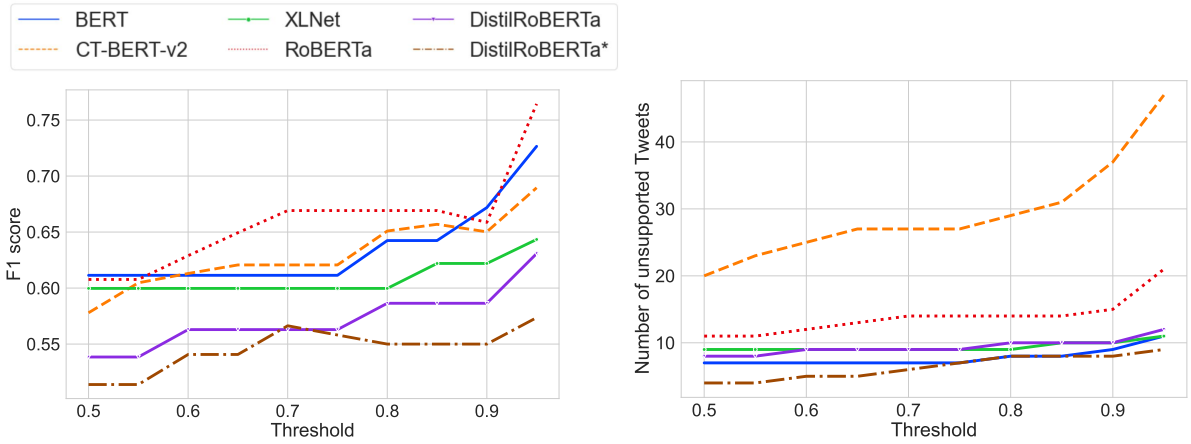
(2) Second, we follow the pipeline approach shown in Figure 7.2, and provide only the check-worthy *weakly false* negative samples from step 1 into step 2. For example, BERT labels 59 check-worthy ⟨Tweet, headline⟩ pairs as *unsupported*, and we evaluate BERT in step 2 using only these 59 pairs. Since we use Longformer only in step 2, for this evaluation we use the results of DistilRoBERTa^p from step 1.

Table 7.1: Experiment results. Model tuned on the paraphrase dataset marked with *. The number of check-worthy pairs labeled *unsupported* in step 1 are shown as $U^{\dagger}$. The numbers of *unsupported* are shown as $U^{\#}$. The number of pairs that are labeled *unsupported* by the model and indeed *unsupported* by annotation is shown as $TN^{\#}$. The ratio of truly unsupported claims to predicted unsupported claims is shown as TN.

	Step 1				**Step 2**										**Pipeline**					
					Sentence					**Full News**					**Sentence**			**Full News**		
Transformer	**P**	**R**	**F1**	$\mathbf{U^{\dagger}}$	**P**	**R**	**F1**	$\mathbf{U^{\#}}$	$\mathbf{TN^{\#}}$	**P**	**R**	**F1**	$\mathbf{U^{\#}}$	$\mathbf{TN^{\#}}$	$\mathbf{U^{\#}}$	$\mathbf{TN^{\#}}$	**TN**	$\mathbf{U^{\#}}$	$\mathbf{TN^{\#}}$	**TN**
BERT	47.2	47.3	47.2	59	56.0	81.3	53.8	8	7	45.4	53.1	49.0	47	25	7	6	85.7	31	18	58.1
CT-BERT-v2	38.9	41.2	37.9	101	55.5	60.4	55.0	24	11	56.3	44.3	49.6	70	31	20	10	50	54	26	48.1
XLNet	**50.4**	**50.4**	**49.1**	86	**59.6**	**84.1**	**59.6**	12	11	45.4	**73.5**	56.1	34	25	9	8	88.9	26	18	69.2
RoBERTa	46.4	47.1	45.7	83	58.4	79.6	57.8	12	10	58.1	61.5	59.8	52	32	11	9	81.8	34	21	61.8
DistilRoBERTa	44.4	45.3	44.3	78	54.7	74.7	51.9	8	6	**67.8**	72.7	**69.4**	39	25	8	6	75	32	20	62.5
DistilRoBERTa*	49.1	49.2	47.5	90	53.6	86.9	49.3	4	4	-	-	-	-	-	4	4	100	-	-	-
Longformer	-	-	-	-	-	-	-	-	-	49.0	52.9	50.9	51	27	-	-	-	38	21	55.3

Table 7.1 shows the comprehensive results of our evaluation of the third task. In the first step, where only the ⟨Tweet, headline⟩ pairs are used, CT-BERT-v2 provides the worst performance. It labels the highest number of pairs as *unsupported*, which leads to low precision. But it achieves the lowest recall as well. This is perhaps not surprising, given that our task spans two genres: Twitter and newswire text, while CT-BERT is a language model with domain-specific pre-training only on Twitter. Thus, it may not be able to properly account for the lexical context of words found in newswire sentences.

We can also see that across all models, the second step, where the entire article is fed sentence-by-sentence, achieves significantly better performance when compared to only working with the headlines. A major difference between the two strategies used in step 2 – using (i) ⟨Tweet, sentence⟩ pairs, and (ii) ⟨Tweet, article⟩ pairs – is that the former tends to tag significantly fewer pairs as *unsupported*. This happens because the classifiers often find a sentence that is similar to the Tweet, and labels the pair as *supported*. Their true negative rate (also known as *specificity*), is thus significantly lower than the models using the latter strategy. It is worth noting, however, that for each model, the *negative predictive values* (*i.e.*, the ratio of truly unsupported claims to predicted unsupported claims) are comparable across the two strategies. With the exception of CT-BERT-v2, we can see that if a model labels a pair as unsupported, it is highly likely that the citation is, indeed, deceptive.



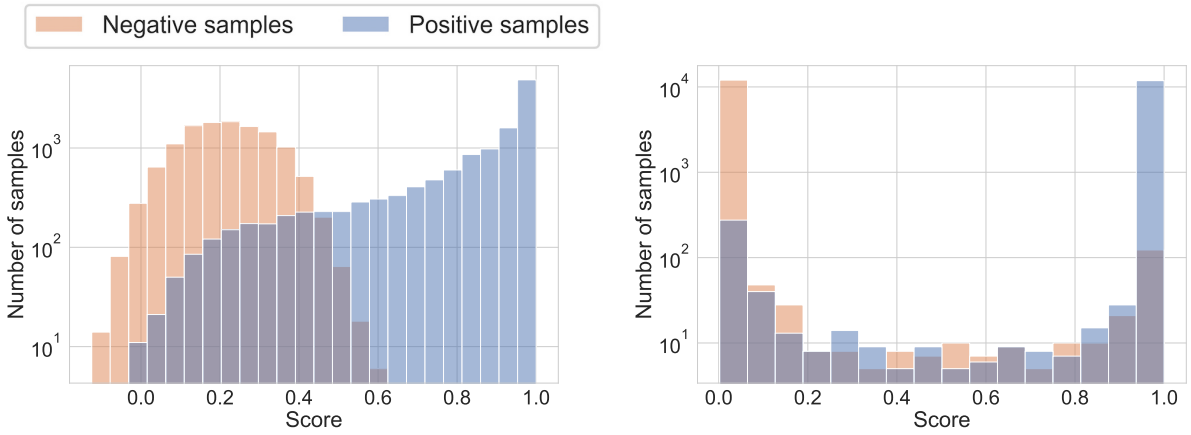
(a) F1 score

(b) Number of unsupported Tweets

Figure 7.4: Varying threshold and results. The results under different thresholds in step 2 as a sentence-level pipeline. Model tuned on the paraphrase dataset marked with *.

There is no consistent improvement between DistilRoBERTa and DistilRoBERTa^p, even though the latter was expected to perform better due to its training on a large number of paraphrases. We believe it is the topic-specific nature of our work which removes the advantage. That is, if DistilRoBERTa^p were trained on a paraphrase corpus related to COVID-19, its improvements would have been more significant. We also do not see Longformer exceeding the other models, in spite of it being designed for longer texts. This can be attributed to the "inverted pyramid" structure of newswire articles, which attempts to place all the essential information in the lead paragraph [86]. Thus, the other models can also capture the relevant information to a similar extent, eroding the relative advantage enjoyed by Longformer in many other tasks with long texts.

Throughout our experiments, each ⟨Tweet, news⟩ pair – whether sentence-by-sentence or as the entire article – was put through a binary classifier, and the classification probability scores were used to determine the final label. A question may be raised at this point regarding the choice of the threshold probability score (0.5) that works as the decision boundary. In Figure 7.4, we show the results of varying the threshold for the second step in Task 2, where ⟨Tweet, news⟩ pairs were labeled on the basis of sentence-level analysis (discussed previously in Section 7.1.3).



(a) Cosine similarity score

(b) Probability score of XLNet

Figure 7.5: Distribution of scores for Tweet-headline pairs on the development set. The y-axis is the number of Tweet-news pairs in log scale within the score range, with (a) showing the distribution of cosine similarity scores among the negative and positive samples respectively, and (b) showing the classification probability score calculated by XLNet on those samples.

7.3 Additional experiments and discussion

Our approach has, in part, been motivated by indications from prior research that a claim and its supporting evidence are semantically similar [6, 74]. A pertinent question, thus, is whether measuring semantic similarity is enough to identify support. In order to investigate this question, we design an additional experiment where the Tweet and the corresponding cited headline are converted to vectors, and their cosine similarity is computed. This is in contrast to the experiments in the previous sections, where the ⟨Tweet, news⟩ pairs were put through a binary classifier, and the classification probability scores were used to determine the final label.

Now, we use the pre-trained DistilRoBERTa language model to obtain the vector representations of each Tweet and headline in the development set. The distribution of the cosine similarity scores are shown in Figure 7.5 (a). For almost all the negative samples, the similarity is under 0.5, but this is true for a significant portion of the positive samples as well. Indeed, 12.2% of the positive samples have a cosine similarity score less than 0.5. A manual inspection of a random sample, however, reveals that only 5% of these are *unsupported*. In contrast, our investigation of the first step of Task 2 shows that 24%-33% (varying between the various models) of the weakly false negative samples are, indeed, *unsupported*. Further, we juxtapose the cosine similarity scores obtained

from DistilRoBERTa with the probability scores of XLNet, shown in Fig 7.5 (b). It immediately becomes clear that the classification approach we took is significantly better at distinguishing the claims accompanied by genuinely supporting news articles from those with deceptive support. The cosine similarity scores obtained using the other pretrained language models provide very similar results, and have not been included for the sake of brevity.

The results of this comparison decidedly indicate that our classifiers, which used the language models and further tuned them for this task, learn certain linguistic signals beyond just semantic similarity. This in turn leads to the system achieving significantly higher specificity (*i.e.*, true negative rate). A higher specificity is a crucially important measure in a practical "real world" scenario of misinformation detection. After all, higher specificity means that fewer genuine Tweets are mislabeled as containing deceptive support. A low-specificity detection system, on the other hand, is likely to annoy the typical user by labeling more of their social media posts as misinformation, and may gradually lead to consumers leaving the platform.

Chapter 8

Conclusions and Future Direction

This work started by approaching the claim extraction task as a sequence labeling problem. We developed a new approach to evaluating the performance of this task by taking into account the portion of the claim that the module gets right for each individual Tweet. This method shows that our model can achieve an F1-score of 77.62%. We also tested this module using Tweet that has no claim and it outputs the proper labels with 82.27% accuracy.

Claim extraction can enhance the speed and accuracy of fact-checking and misinformation detection. Using this technique can help reduce the amount of noise in a text by removing non-informative parts of a statement. As a next step, it might be beneficial to cluster the remaining claims to facilitate the classification of them as credible or untrustworthy. We intend to add a Stance Detection module as well, so as to discard the Tweets with disagreeing stance. The reason is that detecting misinformation depends heavily on the author's stance. In cases where a Tweet contains misleading claims about a certain topic, for example, the COVID-19 vaccine, but the Tweet author disagrees with those claims by adding comments, it is inaccurate and incorrect to label that Tweet as misinformation.

We also investigate a previously unexplored aspect of misinformation, *viz.*, where information is presented in social media with the *appearance* that it is supported by valid and reputable news agencies, but the appearance is deceptive. That is, a claim is made on social media, and a news article is cited, but the article does not actually support the claim! It is often the case that users trust the existence of such support, without verifying any further. Our focus here has been on Twitter posts pertaining to the COVID-19 pandemic. To this end, we provide a new dataset of COVID-19 Tweets, where each Tweet cites a newswire article. We model this as an information retrieval task, where check-worthy claims are first separated from other social media posts, and then, put through classifiers to determine whether or not the apparent support is deceptive. Our approach relies on distant supervision and shows that this is a viable option in the face of a dearth

of annotated data. Our findings reveal that a significant fraction of check-worthy claims – 27.5% of the annotated sample – contain deceptive support. Further, we provide experimental evidence that while semantic similarity plays an important role in finding support for a claim, there are deeper linguistic signals at play, captured by task-specific fine-tuning of language models.

Our work here is a first step in the direction of identifying deceptive support across two genres – social media and newswire articles. There is significant scope for improvement, which we intend to pursue in the near future with larger data sets and seek collaborators to gain access to other social media platforms like Facebook or WhatsApp, where misinformation has been a highly discussed issue [32, 62, 106]. Our study indicates that in order to fight such an infodemic, there is a need to look across genres instead of attending exclusively to social media posts. We hope that our findings can stimulate discussions aimed at making the Internet a more trustworthy landscape among its users, as well as making social media a more reliable source of information. Beyond the claims, our work will also be extended to study counterclaims and counter-beliefs expressed on social media in the form or replies to posts or comments. Analyzing the stance, emotive content, and argumentation in such responses will offer methodological and epistemic breadth to our understanding of misinformation. By offering a holistic view of the issues pertaining to misinformation, we hope that this work, along with our future endeavors, will help us all to discover the truth in a timely fashion.

Another avenue of research enabled by our massive corpus of Tweets accumulated over a period of 18 months is to retrain a Transformer-based model that is domain-specific for Twitter. There are a number of reasons why the language used on Twitter differs slightly from the language we speak, and much more from formal written English, which is how Wikipedia is written – one of the main training sources for the BERT model. Hence, it is valuable to have a model that is trained on Twitter data to better perceive the meaning of Tweets and generate more accurate vector representations of them. In this regard, we have carried out initial experiments to retrain the Electra model using our dataset of more than 150M unique Tweets, although this is a smaller corpus compared to what is used for training the BERT model. Electra is particularly well suited for this objective due to

its innovative training task that can acquire knowledge from the whole input sequence rather than from only masked tokens, as opposed to the Masked Language Modeling task used for training the BERT model. Therefore, less training data is no longer an issue.

Bibliography

[1] Alan Akbik, Tanja Bergmann, Duncan Blythe, Kashif Rasul, Stefan Schweter, and Roland Vollgraf. 2019. FLAIR: An easy-to-use framework for state-of-the-art NLP. In NAACL 2019, 2019 Annual Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations). 54–59.

[2] Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In Proceedings of the 27th international conference on computational linguistics. 1638–1649.

[3] Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual String Embeddings for Sequence Labeling. In COLING 2018, 27th International Conference on Computational Linguistics. 1638–1649.

[4] M. S. Al-Rakhami and A. M. Al-Amri. 2020. Lies Kill, Facts Save: Detecting COVID-19 Misinformation in Twitter. IEEE Access 8 (2020), 155961–155970. https://doi.org/10.1109/ACCESS.2020.3019600

[5] Firoj Alam, Shaden Shaar, Alex Nikolov, Hamdy Mubarak, Giovanni Da San Martino, Ahmed Abdelali, Fahim Dalvi, Nadir Durrani, Hassan Sajjad, Kareem Darwish, and Preslav Nakov. 2020. Fighting the COVID-19 Infodemic: Modeling the Perspective of Journalists, Fact-Checkers, Social Media Platforms, Policy Makers, and the Society. arXiv:2005.00033 https://arxiv.org/abs/2005.00033

[6] Aimée Alonso-Reina, Robiert Sepúlveda-Torres, Estela Saquete, and Manuel Palomar. 2019. Team GPLSI. Approach for automated fact checking. In Proceedings of the Second Workshop on Fact Extraction and VERification (FEVER). Association for Computational Linguistics, Hong Kong, China, 110–114. https://doi.org/10.18653/v1/D19-6617

[7] Sarah Alqurashi, Ahmad Alhindi, and Eisa Alanazi. 2020. Large Arabic Twitter Dataset on COVID-19. arXiv:2004.04315

[8] Ahmer Arif, Kelley Shanahan, Fang-Ju Chou, Yoanna Dosouto, Kate Starbird, and Emma S. Spiro. 2016. How Information Snowballs: Exploring the Role of Exposure in Online Rumor Propagation. In Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing (CSCW '16). Association for Computing Machinery, New York, NY, USA, 466—-477. https://doi.org/10.1145/2818048.2819964

[9] Fatma Arslan, Naeemul Hassan, Chengkai Li, and Mark Tremayne. 2020. A Benchmark Dataset of Check-Worthy Factual Claims. Proceedings of the 14th International AAAI Conference on Web and Social Media (ICWSM 2020) 14, 1 (2020), 821–829.

[10] Juan M. Banda, Ramya Tekumalla, Guanyu Wang, Jingyuan Yu, Tuo Liu, Yuning Ding, and Gerardo Chowell. 2020. A large-scale COVID-19 Twitter chatter dataset for open scientific research - an international collaboration. arXiv:2004.03688 https://arxiv.org/abs/2004.03688

[11] Alberto Barrón-Cedeño, Tamer Elsayed, Preslav Nakov, Giovanni Da San Martino, Maram Hasanain, Reem Suwaileh, and Fatima Haouari. 2020. CheckThat! at CLEF 2020: Enabling the Automatic Identification and Verification of Claims in Social Media. In Advances in Information Retrieval, Joemon M. Jose, Emine Yilmaz, João Magalhães, Pablo Castells, Nicola Ferro, Mário J. Silva, and Flávio Martins (Eds.). Springer International Publishing, Cham, 499–507. https://doi.org/10.1007/978-3-030-45442-5_65

[12] Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The Long-Document Transformer. arXiv:2004.05150

[13] Steven Bird, Edward Loper, and Ewan Klein. 2009. Natural Language Processing with Python. O'Reilly Media Inc., Sebastopol, CA, USA. https://www.nltk.org/

[14] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016. Enriching Word Vectors with Subword Information. arXiv preprint arXiv:1607.04606 (2016).

[15] Declan T Bradley, Marie McFarland, and Mike Clarke. 2014. The effectiveness of disaster risk communication: a systematic review of intervention studies. PLoS currents 6 (2014).

[16] Tuhin Chakrabarty, Christopher Hidey, and Kathleen McKeown. 2019. IMHO fine-tuning improves claim detection. arXiv preprint arXiv:1905.07000 (2019).

[17] Matteo Cinelli, Walter Quattrociocchi, Alessandro Galeazzi, Carlo Michele Valensise, Emanuele Brugnoli, Ana Lucia Schmidt, Paola Zola, Fabiana Zollo, and Antonio Scala. 2020. The covid-19 social media infodemic. Scientific Reports 10, 1 (2020), 1–10.

[18] Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators. In 8th International Conference on Learning Representations. OpenReview.net, Addis Ababa, Ethiopia, 18 pages.

[19] Claude Coulombe. 2018. Text data augmentation made simple by leveraging nlp cloud apis. arXiv preprint arXiv:1812.04718 (2018).

[20] Mark Craven and Johan Kumlien. 1999. Constructing Biological Knowledge Bases by Extracting Information from Text Sources. In Proceedings of the Seventh International Conference on Intelligent Systems for Molecular Biology. AAAI Press, 77–86.

[21] Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. 2019. Autoaugment: Learning augmentation strategies from data. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 113–123.

[22] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. 2020. Randaugment: Practical automated data augmentation with a reduced search space. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. 702–703.

[23] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc Le, and Ruslan Salakhutdinov. 2019. Transformer-XL: Attentive Language Models beyond a Fixed-Length Context. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, Florence, Italy, 2978–2988. https://doi.org/10.18653/v1/P19-1285

[24] Dmitry Davidov, Oren Tsur, and Ari Rappoport. 2010. Enhanced Sentiment Learning Using Twitter Hashtags and Smileys. In Coling 2010: Posters. COLING 2010 Organizing Committee, Beijing, China, 241–249.

[25] Michela Del Vicario, Alessandro Bessi, Fabiana Zollo, Fabio Petroni, Antonio Scala, Guido Caldarelli, H. Eugene Stanley, and Walter Quattrociocchi. 2016. The spreading of misinformation online. Proceedings of the National Academy of Sciences 113, 3 (2016), 554–559. https://doi.org/10.1073/pnas.1517441113

[26] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Jill Burstein, Christy Doran, and Thamar Solorio (Eds.). Association for Computational Linguistics, Minneapolis, MN, USA, 4171–4186. https://doi.org/10.18653/v1/n19-1423

[27] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018).

[28] Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. 2018. Understanding back-translation at scale. arXiv preprint arXiv:1808.09381 (2018).

[29] Steven Y Feng, Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Eduard Hovy. 2021. A survey of data augmentation approaches for nlp. arXiv preprint arXiv:2105.03075 (2021).

[30] Joseph L. Fleiss. 1971. Measuring nominal scale agreement among many raters. Psychological Bulletin 76, 5 (1971), 378–382. https://doi.org/10.1037/h0031619

[31] B. J. Fogg, Gregory Cuellar, and David Danielson. 2007. Motivating, influencing, and persuading users: An introduction to captology. In The Human Computer Interaction Handbook: Fundamentals, Evolving Technologies and Emerging Applications, Julie A. Jacko, Julie A. Jacko, and Andrew Sears (Eds.). CRC Press, New York, NY, USA, 159–172. https://doi.org/10.1201/9781410615862

[32] Sheera Frenkel. 2021. White House Dispute Exposes Facebook Blind Spot on Misinformation. The New York Times. Retrieved August 1, 2021 from https://www.nytimes.com/2021/07/19/technology/facebook-misinformation-blind-spot.html

[33] Zhiwei Gao, Shuntaro Yada, Shoko Wakamiya, and Eiji Aramaki. 2020. NAIST COVID: Multilingual COVID-19 Twitter and Weibo Dataset. arXiv:2004.08145

[34] Avishek Garain. 2020. COVID-19 tweets dataset for Bengali language. https://doi.org/10.21227/wdt0-ya78

[35] Dana Rose Garfin, Roxane Cohen Silver, and E Alison Holman. 2020. The novel coronavirus (COVID-2019) outbreak: Amplification of public health consequences by media exposure. Health psychology 39, 5 (2020), 355.

[36] Amira Ghenai and Yelena Mejova. 2017. Catching Zika Fever: Application of Crowdsourcing and Machine Learning for Tracking Health Misinformation on Twitter. In 2017 IEEE International Conference on Healthcare Informatics (ICHI). IEEE, Park City, UT, USA, 518–518. https://doi.org/10.1109/ICHI.2017.58

[37] Trudy Govier. 2013. A practical study of argument. Cengage Learning.

[38] Hongyu Guo, Yongyi Mao, and Richong Zhang. 2019. Augmenting data with mixup for sentence classification: An empirical study. arXiv preprint arXiv:1905.08941 (2019).

[39] Aditi Gupta, Ponnurangam Kumaraguru, Carlos Castillo, and Patrick Meier. 2014. TweetCred: Real-Time Credibility Assessment of Content on Twitter. In Social Informatics - 6th International Conference (Lecture Notes in Computer Science, Vol. 8851). Springer, Barcelona,Spain, 228–243. https://doi.org/10.1007/978-3-319-13734-6_16

[40] Sardar Hamidian and Mona Diab. 2016. Rumor Identification and Belief Investigation on Twitter. In Proceedings of the 7th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis. Association for Computational Linguistics, San Diego, California, 3–8. https://doi.org/10.18653/v1/W16-0403

[41] Andreas Hanselowski, Hao Zhang, Zile Li, Daniil Sorokin, Benjamin Schiller, Claudia Schulz, and Iryna Gurevych. 2018. UKP-Athene: Multi-Sentence Textual Entailment for Claim Verification. In Proc. First Workshop on Fact Extraction and VERification (FEVER). ACL, Brussels, Belgium, 103–108. https://doi.org/10.18653/v1/W18-5516

[42] Fatima Haouari, Maram Hasanain, Reem Suwaileh, and Tamer Elsayed. 2021. ArCOV-19: The First Arabic COVID-19 Twitter Dataset with Propagation Networks. In Proceedings of the Sixth Arabic Natural Language Processing Workshop. Association for Computational Linguistics, Kyiv, Ukraine (Virtual), 82–91.

[43] Naeemul Hassan, Fatma Arslan, Chengkai Li, and Mark Tremayne. 2017. Toward Automated Fact-Checking: Detecting Check-Worthy Factual Claims by ClaimBuster. In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '17). Association for Computing Machinery, New York, NY, USA, 1803–1812. https://doi.org/10.1145/3097983.3098131

[44] Naeemul Hassan, Gensheng Zhang, Fatma Arslan, Josue Caraballo, Damian Jimenez, Siddhant Gawsane, Shohedul Hasan, Minumol Joseph, Aaditya Kulkarni, Anil Kumar Nayak, et al. 2017. Claimbuster: The first-ever end-to-end fact-checking system. Proceedings of the VLDB Endowment 10, 12 (2017), 1945–1948.

[45] Zhiyong He, Zanbo Wang, Wei Wei, Shanshan Feng, Xianling Mao, and Sheng Jiang. 2020. A survey on recent advances in sequence labeling from deep learning models. arXiv preprint arXiv:2011.06727 (2020).

[46] Marti A. Hearst, Susan T Dumais, Edgar Osuna, John Platt, and Bernhard Scholkopf. 1998. Support vector machines. IEEE Intelligent Systems and their applications 13, 4 (1998), 18–28.

[47] Alex Hernández-García and Peter König. 2018. Data augmentation instead of explicit regularization. arXiv preprint arXiv:1806.03852 (2018).

[48] Ignacio Hernández-García, Teresa Giménez-Júlvez, et al. 2020. Assessment of health information about COVID-19 prevention on the internet: infodemiological study. JMIR public health and surveillance 6, 2 (2020), e18717.

[49] Jennifer L. Hochschild and Katherine Levine Einstein. 2015. Do Facts Matter?: Information and Misinformation in American Politics. University of Oklahoma Press, Norman, OK.

[50] Tamanna Hossain, Robert L. Logan IV, Arjuna Ugarte, Yoshitomo Matsubara, Sean Young, and Sameer Singh. 2020. COVIDLies: Detecting COVID-19 Misinformation on Social Media. https://doi.org/10.18653/v1/2020.nlpcovid19-2.11

[51] Binxuan Huang and Kathleen M. Carley. 2020. Disinformation and Misinformation on Twitter during the Novel Coronavirus Outbreak. arXiv:2006.04278

[52] Muhammad Imran, Carlos Castillo, Fernando Diaz, and Sarah Vieweg. 2015. Processing social media messages in mass emergency: A survey. ACM Computing Surveys (CSUR) 47, 4 (2015), 1–38. https://doi.org/10.1145/2771588

[53] S. Jain, V. Sharma, and R. Kaushal. 2016. Towards automated real-time detection of misinformation on Twitter. In 2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI). IEEE, Jaipur, India, 2015–2020. https://doi.org/10.1109/ICACCI.2016.7732347

[54] Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2016. Bag of tricks for efficient text classification. arXiv preprint arXiv:1607.01759 (2016).

[55] Debanjana Kar, Mohit Bhardwaj, Suranjana Samanta, and Amar Prakash Azad. 2020. No Rumours Please! A Multi-Indic-Lingual Approach for COVID Fake-Tweet Detection. arXiv:2010.06906

[56] Hyunuk Kim and Dylan Walker. 2020. Leveraging volunteer fact checking to identify misinformation about COVID-19 in social media. Harvard Kennedy School Misinformation Review 1, 3 (2020), 10 pages. https://doi.org/10.37016/mr-2020-021

[57] Phil Kim. 2017. Convolutional neural network. In MATLAB deep learning. Springer, 121–147.

[58] Ramez Kouzy, Joseph Abi Jaoude, Afif Kraitem, Molly B El Alam, Basil Karam, Elio Adib, Jabra Zarka, Cindy Traboulsi, Elie W Akl, and Khalil Baddour. 2020. Coronavirus goes viral: quantifying the COVID-19 misinformation epidemic on Twitter. Cureus 12, 3 (2020), e7255.

[59] Bohan Li, Yutai Hou, and Wanxiang Che. 2022. Data augmentation approaches in natural language processing: A survey. AI Open (2022).

[60] Rui Li, Kin Hou Lei, Ravi Khadiwala, and Kevin Chen-Chuan Chang. 2012. TEDAS: A Twitter-based Event Detection and Analysis System. In 2012 IEEE 28th International Conference on Data Engineering. IEEE, Washington D.C., USA, 1273–1276. https://doi.org/10.1109/ICDE.2012.125

[61] Yangming Li, Han Li, Kaisheng Yao, and Xiaolong Li. 2020. Handling rare entities for neural sequence labeling. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. 6441–6451.

[62] Rupali Jayant Limaye, Molly Sauer, Joseph Ali, Justin Bernstein, Brian Wahl, Anne Barnhill, and Alain Labrique. 2020. Building trust while influencing online COVID-19 content in the social media world. The Lancet Digital Health 2, 6 (2020), e277–e278. https://doi.org/10.1016/S2589-7500(20)30084-4

[63] Yijin Liu, Fandong Meng, Jinchao Zhang, Jinan Xu, Yufeng Chen, and Jie Zhou. 2019. GCDT: A global context enhanced deep transition architecture for sequence labeling. arXiv preprint arXiv:1906.02437 (2019).

[64] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692 (2019).

[65] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. arXiv:1907.11692

[66] Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. arXiv preprint arXiv:1603.01354 (2016).

[67] Youness Madani, Mohammed Erritali, and Belaid Bouikhalene. 2021. Using artificial intelligence techniques for detecting Covid-19 epidemic fake news in Moroccan tweets. Results in Physics 25 (2021), 104266. https://doi.org/10.1016/j.rinp.2021.104266

[68] Micol Marchetti-Bowick and Nathanael Chambers. 2012. Learning for Microblogs with Distant Supervision: Political Forecasting with Twitter. In Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics. Association for Computational Linguistics, Avignon, France, 603–612.

[69] Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. 2017. Learned in translation: Contextualized word vectors. Advances in neural information processing systems 30 (2017).

[70] Shahan Ali Memon and Kathleen M Carley. 2020. Characterizing covid-19 misinformation communities using a novel twitter dataset. In Proceedings of the CIKM 2020 Workshops. CEUR-WS.org, Galway, Ireland, 9 pages.

[71] Michele Miller, Tanvi Banerjee, Roopteja Muppalla, William Romine, and Amit Sheth. 2017. What are people tweeting about Zika? An exploratory study concerning its symptoms, treatment, transmission, and prevention. JMIR Public Health and Surveillance 3, 2 (2017), e38. https://doi.org/10.2196/publichealth.7157

[72] Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP. Association for Computational Linguistics, Suntec, Singapore, 1003–1011. https://www.aclweb.org/anthology/P09-1113

[73] Behrang Mohit. 2014. Named entity recognition. In Natural language processing of semitic languages. Springer, 221–245.

[74] Mitra Mohtarami, Ramy Baly, James Glass, Preslav Nakov, Lluís Màrquez, and Alessandro Moschitti. 2018. Automatic Stance Detection Using End-to-End Memory Networks. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers). Association for Computational Linguistics, New Orleans, Louisiana, 767–776. https://doi.org/10.18653/v1/N18-1070

[75] Martin Müller, Marcel Salathé, and Per Egil Kummervold. 2020. COVID-Twitter-BERT: A Natural Language Processing Model to Analyse COVID-19 Content on Twitter. arXiv:2005.07503

[76] Dat Quoc Nguyen, Thanh Vu, Afshin Rahimi, Mai Hoang Dao, Linh The Nguyen, and Long Doan. 2020. WNUT-2020 Task 2: Identification of Informative COVID-19 English Tweets. In Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020). Association for Computational Linguistics, Online, 314–318. https://doi.org/10.18653/v1/2020.wnut-1.41

[77] Yixin Nie, Haonan Chen, and Mohit Bansal. 2019. Combining Fact Extraction and Verification with Neural Semantic Matching Networks. In Proc. AAAI Conference on Artificial Intelligence, Vol. 33. 6859–6866. https://doi.org/10.1609/aaai.v33i01.33016859

[78] Brendan Nyhan and Jason Reifler. 2010. When corrections fail: The persistence of political misperceptions. Political Behavior 32, 2 (2010), 303–330.

[79] Yeimer Ortiz-Martínez and Luisa F Jiménez-Arcia. 2017. Yellow fever outbreaks and Twitter: Rumors and misinformation. American Journal of Infection Control 45, 7 (2017), 816–817.

[80] Sunday Oluwafemi Oyeyemi, Elia Gabarron, and Rolf Wynn. 2014. Ebola, Twitter, and misinformation: a dangerous combination? BMJ 349 (2014), g6178. https://doi.org/10.1136/bmj.g6178

[81] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). 1532–1543.

[82] Gordon Pennycook, Tyrone D Cannon, and David G Rand. 2018. Prior exposure increases perceived accuracy of fake news. Journal of experimental psychology: general 147, 12 (2018), 1865. https://doi.org/10.2139/ssrn.2958246

[83] Sarah Perez. 2017. Twitter officially expands its character count to 280 starting today. TechCrunch. Retrieved June 6, 2021 from https://techcrunch.com/2017/11/07/twitter-officially-expands-its-character-count-to-280-starting-today/

[84] Sarah Perez. 2018. Twitter's doubling of character count from 140 to 280 had little impact on length of tweets. TechCrunch. Retrieved June 6, 2021 from https://techcrunch.com/2018/10/30/twitters-doubling-of-character-count-from-140-to-280-had-little-impact-on-length-of-tweets/

[85] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep Contextualized Word Representations. In NAACL.

[86] Horst Po¨ ttker. 2003. News and its communicative quality: The inverted pyramid—when and why did it appear? Journalism Studies 4, 4 (2003), 501–511.

[87] Vahed Qazvinian, Emily Rosengren, Dragomir R. Radev, and Qiaozhu Mei. 2011. Rumor has it: Identifying Misinformation in Microblogs. In Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, Edinburgh, Scotland, UK, 1589–1599.

[88] Lance A Ramshaw and Mitchell P Marcus. 1999. Text chunking using transformation-based learning. In Natural language processing using very large corpora. Springer, 157–176.

[89] Mehdi Regina, Maxime Meyer, and Sébastien Goutal. 2020. Text Data Augmentation: Towards better detection of spear-phishing emails. arXiv preprint arXiv:2007.02033 (2020).

[90] World Health Organization. 2010. What is a pandemic? Retrieved April 27, 2021 from https://www.who.int/csr/disease/swineflu/frequently_asked_questions/pandemic/en/

[91] World Health Organization. 2020. Munich Security Conference. Retrieved April 27, 2021 from https://www.who.int/director-general/speeches/detail/munich-security-conference

[92] World Health Organization. 2020. WHO Director-General's opening remarks at the media briefing on COVID-19 - 11 March 2020. Retrieved April 27, 2021 from https://www.who.int/director-general/speeches/detail/who-director-general-s-opening-remarks-at-the-media-briefing-on-covid-19---11-march-2020

[93] Landis. J. Richard and Gary G. Koch. 1977. The Measurement of Observer Agreement for Categorical Data. Biometrics 33, 1 (1977), 159–174.

[94] Victoria L Rubin, Yimin Chen, and Nadia K Conroy. 2015. Deception detection for news: three types of fakes. Proceedings of the Association for Information Science and Technology 52, 1 (2015), 1–4. https://doi.org/10.1002/pra2.2015.145052010083

[95] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. arXiv:1910.01108

[96] Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Improving neural machine translation models with monolingual data. arXiv preprint arXiv:1511.06709 (2015).

[97] Gautam Kishore Shahi, Anne Dirkson, and Tim A. Majchrzak. 2021. An exploratory study of COVID-19 misinformation on Twitter. Online Social Networks and Media 22 (2021), 100104. https://doi.org/10.1016/j.osnem.2020.100104

[98] Ivor Shapiro, Colette Brin, Isabelle Bèdard-Brûlé, and Kasia Mychajlowycz. 2013. Verification as a Strategic Ritual. Journalism Practice 7, 6 (2013), 657 – 673. https://doi.org/10.1080/17512786.2013.765638

[99] Connor Shorten and Taghi M Khoshgoftaar. 2019. A survey on image data augmentation for deep learning. Journal of big data 6, 1 (2019), 1–48.

[100] Kai Shu, Deepak Mahudeswaran, Suhang Wang, Dongwon Lee, and Huan Liu. 2020. Fakenewsnet: A data repository with news content, social context, and spatiotemporal information for studying fake news on social media. Big Data 8, 3 (2020), 171–188.

[101] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. 2017. Fake news detection on social media: A data mining perspective. ACM SIGKDD explorations newsletter 19, 1 (2017), 22–36. https://doi.org/10.1145/3137597.3137600

[102] Lisa Singh, Shweta Bansal, Leticia Bode, Ceren Budak, Guangqing Chi, Kornraphop Kawintiranon, Colton Padden, Rebecca Vanarsdall, Emily Vraga, and Yanchen Wang. 2020. A first look at COVID-19 information and misinformation sharing on Twitter. arXiv:2003.13907 [cs.SI]

[103] Beth St. Jean, Mega Subramaniam, Natalie Greene Taylor, Rebecca Follman, Christie Kodama, and Dana Casciott. 2015. The influence of positive hypothesis testing on youths' online health-related information seeking. New Library World 116, 3/4 (2015), 136–154. https://doi.org/10.1108/NLW-07-2014-0084

[104] James Thorne, Mingjie Chen, Giorgos Myrianthous, Jiashu Pu, Xiaoxuan Wang, and Andreas Vlachos. 2017. Fake news stance detection using stacked ensemble of classifiers. In Proceedings of the 2017 EMNLP workshop: natural language processing meets journalism. 80–83.

[105] James Thorne and Andreas Vlachos. 2018. Automated fact checking: Task formulations, methods and future directions. arXiv preprint arXiv:1806.07687 (2018).

[106] Mayowa Tijani. 2020. How to spot COVID-19 misinformation on WhatsApp. Agence France-Presse. Retrieved August 1, 2021 from https://factcheck.afp.com/how-spot-covid-19-misinformation-whatsapp

[107] Joseph E Uscinski and Ryden W Butler. 2013. The Epistemology of Fact Checking. Critical Review 25, 2 (2013), 162–180. https://doi.org/10.1080/08913811.2013.843872

[108] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In Advances

in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017. Curran Associates, Inc., Long Beach, CA, USA, 5998–6008.

[109] Soroush Vosoughi, Deb Roy, and Sinan Aral. 2018. The spread of true and false news online. Science 359, 6380 (2018), 1146–1151. https://doi.org/10.1126/science.aap9559

[110] Atro Voutilainen. 2003. Part-of-speech tagging. The Oxford handbook of computational linguistics (2003), 219–232.

[111] Wei Wang, Vincent W Zheng, Han Yu, and Chunyan Miao. 2019. A survey of zero-shot learning: Settings, methods, and applications. ACM Transactions on Intelligent Systems and Technology (TIST) 10, 2 (2019), 1–37.

[112] Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. 2020. Unsupervised data augmentation for consistency training. Advances in Neural Information Processing Systems 33 (2020), 6256–6268.

[113] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. arXiv preprint arXiv:1906.08237 (2019).

[114] John Zarocostas. 2020. How to fight an infodemic. The Lancet 395, 10225 (2020), 676. https://doi.org/10.1016/S0140-6736(20)30461-X

[115] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. 2017. mixup: Beyond empirical risk minimization. arXiv preprint arXiv:1710.09412 (2017).

[116] Zhe Zhao, Paul Resnick, and Qiaozhu Mei. 2015. Enquiring Minds: Early Detection of Rumors in Social Media from Enquiry Posts. In Proceedings of the 24th International Conference on World Wide Web (WWW '15). International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 1395—-1405. https://doi.org/10.1145/2736277.2741637

[117] Xing Zhou, Juan Cao, Zhiwei Jin, Fei Xie, Yu Su, Dafeng Chu, Xuehui Cao, and Junqiang Zhang. 2015. Real-Time News Certification System on Sina Weibo. In Proceedings of the 24th International Conference on World Wide Web (WWW '15 Companion). Association for Computing Machinery, New York, NY, USA, 983—-988. https://doi.org/10.1145/2740908.2742571

[118] Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In Proceedings of the IEEE international conference on computer vision. Santiago, Chile, 19–27. https://doi.org/10.1109/ICCV.2015.11

[119] Chaoyuan Zuo, Narayan Acharya, and Ritwik Banerjee. 2020. Querying Across Genres for Medical Claims in News. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics, Online, 1783–1789. https://doi.org/10.18653/v1/2020.emnlp-main.139

[120] Chaoyuan Zuo, Ritwik Banerjee, Hossein Shirazi, Fateme Hashemi Chaleshtori, and Indrakshi Ray. 2022. Seeing Should Probably not be Believing: The Role of Deceptive Support in COVID-19 Misinformation on Twitter. Journal of Data and Information Quality (2022).

[121] Chaoyuan Zuo, Ayla Karakas, and Ritwik Banerjee. 2019. To Check or not to Check: Syntax, Semantics, and Context in the Language of Check-worthy Claims. In Experimental IR Meets Multilinguality, Multimodality, and Interaction – Proceedings of the 10th International Conference of the CLEF Association (Lecture Notes in Computer Science, Vol. 11696), Fabio Crestani, Martin Braschler, Jacques Savoy, Andreas Rauber, Henning Müller, David E. Losada, Gundula H. Bürki, Linda Cappellato, and Nicola Ferro (Eds.). Springer International Publishing, Lugano, Switzerland, 271–283. https://doi.org/10.1007/978-3-030-28577-7_23