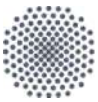

University of Stuttgart

Institut for Natural Language Processing

Pfaffenwaldring 5b
70569 Stuttgart

Bachelor Thesis

Influence of Claims and their Truthfulness on Engagement in Social Media

Tobias Schiebel

Study program:	Computer Science
1. Examiner:	PD Dr. Roman Klinger
2. Examiner:	-
Advisors:	PD Dr. Roman Klinger
start date:	10.12.2021
end date:	10.06.2022

Abstract

Motivation. Information can spread rapidly via social media, the spread of false information represents a serious threat to modern liberal democratic society. Therefore, it is crucial to research the way it is propagated. We propose the first approach to do so, covering an unresolved research area.

Research Question. Do social media posts containing false claims receive more user engagement than posts with true claims?

Method. We analyse a potential correlation between the veracity and the engagement of social media posts. Therefore, we identify claims with a binary claim classifier and estimate their truthfulness based on their similarity to known-veracity-claims.

Result. We report no clear correlation between the veracity and the received engagement of posts for all analysed classification models and metrics to measure engagement.

Conclusion. We are unable to demonstrate an impact of the veracity of claims in posts on the received engagements with our means. We conclude the necessity of further investigation into this research area.

Contents

1 Introduction

We are living in an age of information. The development of the Internet and social media has massively changed the way we interact with information in our lives. The flow of information has transformed away from traditional media towards online newsfeeds. Social media, in particular, is becoming increasingly important (Graham and Dutton, 2019). As of 2020, almost 61% of the world population is using social media, with an average engagement on 6.6 different social media platforms[1].

The three large social media platforms Twitter, Reddit, and Facebook[2] enable their users to post a short text, an image, or video online almost instantly. Usually, social media platforms involve a "follow" feature, which allows users to subscribe to another user's posts. The piece of information is then shared with the followers of the author and is also spread further through the recommendation algorithm of the platform. As a result, the post appears in the website's search, trends, or user recommendations.

Some of the enormous numbers of posts that pass through social media every day contain false information, which is now often referred to as "fake news". As Guess and Lyons (2020) elaborate, one has to distinguish between the two terms *misinformation* and *disinformation* within the terminology of false information. The crucial difference is made by the intent behind it. While misinformation simply designates wrong information regardless of the author's intent, disinformation, on the other hand, denotes intentionally spread misinformation.

Further, Guess and Lyons (2020) state that misinformation commonly results from

[1] https://backlinko.com/social-media-users
[2] https://twitter.com/, https://www.reddit.com/, https://www.facebook.com/

the slight accidental falsification of valid facts and therefore has a rather harmless impact. The authors of misinformation do not mean to spread it. Disinformation, on the other hand, is by far more dangerous as it is composed to be deceptive, destructive, and divisive. Authors of disinformation want to spread their malicious information as broadly as possible. The Internet, and especially social media platforms, provides an optimal breeding ground for these intentions, because it enables the authors to reach out fast to the entire user base of the Internet. Either way, posts containing misinformation bear the risk of being easily shared further by readers on social media, whether they just believe the contents to be true or intend to spread disinformation.

The diffusion of such misleading information poses a potential threat to democracy and broader society (Allcott et al., 2019). Therefore, countermeasures have already been taken, such as the collaboration of Facebook with fact-checking sites to evaluate the veracity of posts flagged as potentially false by Facebook users (Allcott et al., 2019). Nevertheless, Ferrara et al. (2020) have shown that even small so-called bot networks (multiple automated programs that mutually share and spread their posts) can have a significant impact on political discourse on social media. The bot networks can systematically spread disinformation, demonstrating a non-negligible threat of election manipulation. Further, disinformation is of high relevance for political propaganda, as seen in the ongoing conflict between Russia and Ukraine (Mejias and Vokuev, 2017). Fake news can have dangerous physical consequences too, as evidenced by the course of events of "pizzagate"[3], in which an American tried to rescue allegedly abused children with a gun from a restaurant because he read about a conspiracy online.

Fake news often become quite popular, such as the arguably most common myth spread on the internet, *"people swallow eight spiders a year while they sleep."*[4], which is highly implausible according to biologists[5]. This example leads to the assumption

[3] https://www.washingtonpost.com/local/pizzagate-from-rumor-to-hashtag-to-gunfire-in-dc/2016/12/06/4c7def50-bbd4-11e6-94ac-3d324840106c_story.html

[4] https://www.snopes.com/fact-check/swallow-spiders/

[5] https://www.scientificamerican.com/article/fact-or-fiction-people-swallow-8-spiders-a-year-while-they-sleep1/

that statements which concern a large part of the user community spread particularly well regardless of their veracity. Furthermore, the usage of headline-like language could amplify this effect. Therefore, the following questions are to be answered: Are authors of misinformation aware of the factors that influence the propagation of their posts? Do they deliberately use these techniques to inflict higher engagement on their posts and therefore spread them more efficiently? Are they more successful in propagating their posts than authors of regular posts? We approach these questions within our research in two steps: First, determining whether a claim occurs in a post; second, analysing the propagation of known-false-claims in comparison to known-true-claims.

For the first part, we implement a binary text classifier that assigns a post to either the class of claims or non-claims. The machine learning model is trained on samples for each class, learning what posts in each class look like. Thus, we can extract posts containing claims from a total set of posts, as only those are of interest to our study. Next, we want to assign a veracity to each claim. As fact-checking every single claim is extremely costly, we simply give an estimation of a claim's veracity based on its similarity to claims with known veracity. In the second part, we establish various metrics to assess the spread of the claims. Therefore, we can evaluate and compare the metrics for true and false claims.

The goal of our research is to explore the connection between the veracity of claims in social media posts and their propagation with the means of a calculated performance score to finally answer the research hypothesis: Do social media posts containing false claims receive more user engagement than posts with true claims? This is analysed for the textual contents of a set of Twitter posts ("tweets") on the topic of the COVID-19 pandemic in 2021.

2 Fundamentals

This section aims at giving a short introduction to fundamental methods of machine learning-based text classification. We describe the general methodology to process social media posts into a numerical representation that can be used as input for classification models. In addition, we shortly introduce the text classification models we use: support vector machine (SVM), random forest, and LightGBM. Finally, we introduce the standard metrics to evaluate the performance of classifiers.

2.1 Embeddings

In natural language processing (NLP) we typically start with a corpus consisting of multiple documents we want to analyse, in our case, social media posts. Before we can get into the analysis, we usually need to clean our documents within the step of preprocessing. This involves, for example, applying uniform (lower) case and getting rid of duplicate white space characters, URLs, and special characters. Further, it might be desirable to omit common words and standardise word variations. See Section 4.2.1 to read more about the so-called procedures of stop word removal and lemmatization.

Once we are done with preprocessing, we want to transform our cleaned documents so that machine learning models can process them. Therefore, we can compute the term frequency-inverse document frequency (TF-IDF) weights for all words occurring in the corpus. The TF-IDF weight measures the value of a word within a document, weighted by the frequency of the word in documents of the entire corpus. Thus, with the TF-IDF weight, we obtain a statistic that measures the importance

of a word to a post. As a result, we can represent our posts as multi-dimensional vectors of TF-IDF weights for each word, whereas every word in the corpus has its own dimension. Note, that the resulting document vectors are sparse as many TF-IDF weights are zero for all the words that do not occur in a post. With this established vector representation, we can consider documents with vectors that are close to each other as similar. This is helpful for text classification or clustering.

An alternative to TF-IDF is GloVe[6] (global vectors for word representation), a regression model for unsupervised learning developed by Stanford. GloVe aggregates global word-word co-occurrence statistics from a corpus to obtain the vector representations. In the resulting vector space, we can observe linear substructures. Linear substructures are considered similar representations of word-pairs. For example, the vector difference between the pair *woman* and *man* is almost parallel to the vector difference between *queen* and *king*, whereas *woman* and *queen* are on the same end of their pairs. Therefore, it can be seen that GloVe captures a lot of the meaning of words within such vector differences.

However, GloVe does not consider the context in which a word is used, which can be of high importance, as this example shows: Compare the sentence *"I got a new* ***bow*** *for the medieval festival."* with the sentence *"It is considered polite to* ***bow*** *as a greeting in Japan."*. The BERT (bidirectional encoder representations from transformers) language representation model (Devlin et al., 2019) accounts for that by considering the left and right context of a word. BERT is quite powerful and performs well in a wide range of NLP tasks.

2.2 Machine Learning Classifiers

2.2.1 Support Vector Machine (SVM)

We will describe SVMs based on the explanations from Manning et al. (2008). As they phrase it, SVM is a "vector space based machine learning method where the

[6] https://nlp.stanford.edu/projects/glove/

goal is to find a decision boundary between two classes that is maximally far from any point in the training data (possibly discounting some points as outliers or noise)". The decision boundary is a hyperplane represented by a few points in the vector space that we call the *support vectors*, hence the name of the classifier. For a binary decision problem and an unclassified data point, we determine its class according to the side of the decision boundary it is located on. Further, we speak of the smallest distance from the separator to a data vector as the *margin*. It is desirable to maximize the margin to reduce low certainty decisions.

Formally, we can define the decision hyperplane by an intercept term b and a perpendicular normal vector (weight vector) $\vec{w}$, so that all points $\vec{x}$ satisfy the equation

$$\vec{w}^{\mathrm{T}}\vec{x} + b = 0.$$

For a binary classification problem, we consider a set of data points (vectors) $\vec{x_i}$ each with a class label y_i, which is either 1 or -1 for the two classes. Thus, we can define a linear classifier with the sign operator as

$$f(\vec{x}) = \mathrm{sign}(\vec{w}^{\mathrm{T}}\vec{x} + b).$$

In the case of not linearly separable data, we map the original feature space to a higher-dimensional feature space. Therefore, we use so-called kernel functions, which must be continuous, symmetric, and have a positive definite gram matrix. Such kernel functions have a mapping to a vector space where the dot product between vectors is the same as in the original, keeping the original data constellation intact.

2.2.2 Random Forest

The random forest classifier is an ensemble-based learning method developed by Breiman (2001). It combines the outcomes of several decision trees to only one single result. A decision tree aims at finding an answer to a decision by going through a series of questions. It consists of multiple nodes representing these questions, which split the tree into branches for each possible answer. The leaf nodes at the very end of the branches denote the final decision, dependent on the path taken through the

questions, the split. Therefore, the goal of decision trees is to find the best split according to metrics like the gini impurity or mean square error. Decision trees are commonly trained with the classification and regression tree (also known as "CART") algorithm by Breiman et al. (1984).

The random forest algorithm creates a set of uncorrelated decision trees by using both bagging and feature randomness, the so-called random subspace method. The underlying method of bagging involves selecting random samples of data points with replacement, training them independently, and choosing the majority of the predicted classes as result for classification. The random subspace method established random feature subsets, which sustains the low correlation among trees. Furthermore, typically an out-of-bag sample of the training data is used for cross-validation.

2.2.3 LightGBM

LightGBM is based on the gradient boosting decision tree (GBDT) model. GBDT represents another ensemble decision tree model with the characteristic to iteratively learn the decision trees from the negative gradients (so-called residual errors).

Ke et al. (2017) proposed their implementation called LightGBM with the addition of two major improvements: gradient-based one-side sampling and exclusive feature bundling. Gradient-based one-side sampling focuses on data instances with larger gradients as they are proven to contribute more to the information gain. Therefore, LightGBM grows the decision trees leaf-wise in contrast to the usual level (depth)-wise implementation. To avoid overfitting, a maximum depth is specified. Exclusive feature bundling approaches the typically sparse feature space. It bundles the so-called exclusive features that almost never take non-zero values simultaneously. The implementation of these two techniques provides LightGBM with a training time over twenty times as fast as typical GBDT classifiers.

	Predicted Positive	Predicted Negative
Actually Positive	True Positive (TP)	False Negative (FN)
Actually Negative	False Positive (FP)	True Negative (TN)

Table 1: Confusion matrix for the evaluation of classifiers

2.3 Evaluation Metrics

For the evaluation of the performance of our classifiers, we use the metrics accuracy, precision, recall, and the F_1-score. To understand the following definitions of those, we first illustrate the evaluation confusion matrix in Table 1.

Therefore, the accuracy A is defined as the ratio of the number of right predictions to the total number of samples:

$$A = \frac{TP + TN}{TP + TN + FP + FN}.$$

The precision P is defined as the ratio of the number of true positives to the total positive predictions:

$$P = \frac{TP}{TP + FP}.$$

Furthermore, the recall R is defined as the ratio of the number of true positives to the total number of actual positives:

$$R = \frac{TP}{TP + FN}.$$

Finally, the standard F_1-score is defined as the harmonic mean of precision and recall:

$$F_1 = \frac{2PR}{P + R}.$$

We make use of all these metrics in Section 5.2 to report the performance of our used classification models. This concludes our short digression about NLP fundamentals.

3 Related Work

Our research process can be broken down into three essential parts: identifying claims in documents, estimating their veracity, and finally evaluating the propagation performance of such posts. Previous work has researched each of those fields on its own. Most of the presented projects consider posts from Twitter (or even more platforms) as data, as the platform is populated by a large majority of mobile users[7] and therefore represents "an ideal environment for the dissemination of breaking-news directly from the news source and/or geographical location of events" (Castillo et al., 2011). For our research task, we can consider both claims containing valid and false information as *breaking-news*. This makes Twitter a particularly exciting social media platform for research regarding claims and their veracity. To the best of our knowledge, there have not been any efforts to draw a connection between the truthfulness and the propagation of a tweet so far. Therefore, our research, representing a combination of these fields, covers a yet unexplored research area.

3.1 Argument Mining

3.1.1 Claim Detection

The first topic area of claim detection is often also referred to as rumour detection, as the term *rumour* describes a "story or a statement whose truth value is unverified or deliberately false"(Allport and Postman, 1965) in social psychology literature. We emphasize that this definition of *rumours* does not only include statements that

[7] https://www.statista.com/chart/1520/number-of-monthly-active-twitter-users/

claim to be factual, as one could assume, but also statements with, at the time, unknown veracity. In the following, we use *claim* synonymously to *rumour*.

Claim detection involves the challenge of finding features and/or patterns to reliantly recognise claims in documents. Therefore, for example, the sentence *"I hope that the weather will be sunny tomorrow."* would be classified as a non-claim compared with *"The weather was sunny yesterday in Stuttgart."*, which would be classified as a claim. However, it is not always that simple. Notice that, according to our definition, the set of claims also involves statements whose truthfulness can prove themselves to be hard or even impossible to verify. Claims involving personal preferences like *"My favourite ice cream flavour is chocolate."* or personal stories as *"Alan's sister got tested positive for COVID-19."* fall into that category. On the other hand, there are statements containing far more broad and general claims, such as *"Clapping is effective against the coronavirus."*, which are of high interest to our study as they can be easily verified and are more likely to be connected to deliberately spread disinformation.

As the general task of claim detection therefore proves itself quite difficult, other authors have previously addressed the more specific problem of context dependent claim detection (CDCD). In CDCD, the task of claim detection is narrowed down to detecting relevant claims connected to a given, often controversial, topic area. This problem definition has first been specified by Levy et al. (2014). Their first approach to CDCD is based on a cascade that continuously narrows down text passages to potential claims step-by-step. Further, they propose highly complex components of their cascade, such as their component to identify potential sentences containing claims based on, among other features, similarity to the topic, subjectivity, and sentiment. They select a small number of related Wikipedia articles to obtain the claims for the given topic. CDCD has also been explored in the area of evidence retrieval by Rinott et al. (2015). Unfortunately, social media posts are usually brief and hence rarely provide us with sufficient context to supply a context-dependent system for our task.

Nevertheless, claim detection can be approached independent of the context. It can also be seen as a specific task of binary text classification, which is one of the most

common tasks in the field of natural language processing. Lippi and Torroni (2015) apply a SVM-based system to detect claims based on common rhetorical structures. They represent these structures in parse trees and measure the similarity between such trees with so-called tree kernels. The authors train their SVM classifier on positive and negative examples and achieve results that are close to the performance of context-dependent systems.

Furthermore, neural networks are particularly well suited for claim detection as they excel in the challenge of figuring out and recognising patterns. For example, Ma et al. (2016) focus on the usage of recurrent neural networks (RNN) to detect rumours. A RNN differs from basic feed-forward neural networks in that it uses time-series data and information from previous inputs that influence the input and output of subsequent layers. As backpropagation is highly impractical for evaluation due to vanishing and exploding gradients, the authors make use of two different kinds of memory structures: long short-term memory (LSTM) and gated recurrent units. They obtained results outperforming state-of-the-art methods such as decision tree classifiers and linear SVM classification on two tweet datasets. Both datasets contained roughly equal numbers of claims and non-claims.

Ma et al. (2019) also proposed a follow-up approach using generative adversarial networks (GAN). In a GAN, two neural networks compete against each other. One of them is continuously provided with tasks it has to solve, so it is always training and getting better at finding solutions to the tasks it is presented with. The other neural network is meanwhile presented with the challenge of designing progressively harder tasks for the first one. Thus, the two networks improve each other throughout the process. In this application, a generative model is challenging a discriminative classifier over and over again to distinguish synthetic snippets from real ones. The reason behind this approach was the issue of promoted campaigns influencing automated rumour detection. The resulting, robust discriminator provides significant improvements to their previous results and beats even convolutional neural network- or RNN-based models in performance.

3.1.2 Fact-Checking

Second, large fact-checking organizations approach the topic of fact-checking and rumour verification in three major areas[8]: selection, research, and evaluation. Selection is the process of choosing claims to be fact-checked, while research describes the methodology of investigating a claim. Finally, evaluation concludes with determining the veracity of the claim. However, scientific research on this topic can be separated into only two major areas: credibility analysis and (automated) rumour verification. The first area of credibility analysis represents another feature engineering task that can be approached using neural networks. In the second area, it has to be determined what exactly is considered truthful and what is not. Therefore, either the given training data sets are usually labelled by multiple independent annotators or the domain is narrowed down to a few common statements, the truth of which can be easily assessed.

The veracity of a post can be determined based on the credibility of the author of the post. Canini et al. (2011) set up a study about source credibility similar to the psychologists Birnbaum and Stegner (1979) based on the hypothesis that source credibility can be modelled with a simple average. The experimental setup consists of participants estimating the fair market value of used cars. Several details about the car are presented to them, including the market price according to Kelley Blue Book[9] (a renowned vehicle valuation company). Further, the participants are provided with a slightly altered Kelley Blue Book price, proposed by a set of Twitter users with different ranges of expertise. The participants give an estimation before encountering the third party opinion and after. Finally, there can be drawn an implicit value of credibility for each third party Twitter user. Based on this value, the veracity of the posts from the user can be evaluated.

Also, Castillo et al. (2011) developed an automatic credibility analysis. The underlying data was obtained by automatically detecting news topics on Twitter with Twitter Monitor (Mathioudakis and Koudas, 2010) based on frequently used key-

[8] https://ballotpedia.org/The_methodologies_of_fact-checking
[9] https://www.kbb.com/

words. Furthermore, it was distinguished between news and conversation about the news. This distinction is required to only evaluate the tweets classified as news rather than the chatter about them. As a result, the authors present four kinds of features relevant for credibility analysis: message-, user-, topic-, and propagation-based features. As we have seen above, with these features, the credibility of a user can be determined to further evaluate the veracity of their posts.

These two previous approaches aimed at determining a credibility score for authors require a lot of context and information about the authors. However, most annotated datasets suitable for fact-checking do not contain much of this needed amount of information, such as the large LIAR dataset assembled by Wang (2017). The dataset designed for fake news detection consists of statements collected from PolitiFact[10] and sets the basis for automated fact-checking with this extensive corpus. In regard to datasets consisting of social media posts, they are a collection of posts by a wide variety of users instead of several posts per user, making it difficult to precisely measure the credibility. Thus, we assess user-based credibility analysis as not being a suitable approach for veracity estimation in our research.

Instead, the work of Kochkina et al. (2018) is similar to our approach by its concept. The authors cover a multi-task learning approach for rumour resolution designed as a pipeline. Their pipeline comprises the detection of rumours, tracking down sources, stance detection, and finally the verification of the found rumours. They use the PHEME (Zubiaga et al., 2016; 2017) and RumourEval (Derczynski et al., 2017) datasets, which contain Twitter conversations. The authors propose a system "consisting of an LSTM layer followed by several dense ReLU layers and a softmax layer".

Furthermore, the task of rumour verification within the scope of SemEval 2019, Gorrell et al. (2019) provides insight into the performance of different types of systems best suited for this challenge. The authors provided their dataset from RumourEval 2017 as training data, consisting of about 300 tweets about eight news events and several corresponding tweets discussing those. Additionally, they pro-

[10] https://www.politifact.com/

vided an annotated testing dataset containing tweets about natural disasters and a few Reddit discussion threads. The Reddit data was considered particularly interesting for testing, as most posts implicitly query the rumour, contrary to Twitter posts, which mostly present rumours as valid information. The 13 submitted systems were ranked according to their achieved macro-averaged F_1- and root mean square error scores. The best-performing systems implements "an ensemble of classifiers (SVM, random forest, logistic regression) [...], where individual post representations are created using an LSTM with attention". This shows, that we do not necessarily require neural network-based systems for this task and can instead rely on a range of classic models, which is ideal for the first exploration of our research hypothesis.

3.2 Popularity Assessment

Lastly, the topic area of popularity assessment is about accurately measuring the popularity and engagement of posts, usually in terms of the number of likes or answers they receive.

The problem of popularity prediction of posts is often approached with regard to posted images. Therefore, the image itself, features such as user statistics of the author, and metadata of the post are considered. Hidayati et al. (2017) examined these three features in over 400k posts. They use a model based on support-vector regression (SVR) as well as a regression tree model to establish two popularity scores for each post. Those two scores are then combined into one final popularity score estimate. In detail, their user profile features are based on the number of followers of the author, the average views their posts receive, and the number of groups a user follows. Further, they consider the textual description of the image as well as the title and its length, the used tags, and the time and date of the posting as metadata features. Lastly, they consider a range of image based features to measure the aesthetic score of the image, which we will omit here due to their complexity. While they observe some overfitting, their regression tree model consistently outperforms the SVR model.

Ding et al. (2019a) propose a similar approach for image popularity assessment. Besides providing a large-scale database for this research area, the authors use deep neural networks (DNN) on the database with learning-to-rank to predict popularity scores. They also consider user statistics (most importantly, the number of followers an author has), the image caption, and the upload time as three major factors impacting popularity. However, we choose to not consider images as data as they do not provide us with any statements or claims that we can analyse towards their veracity. Nevertheless, we note the findings on the impact of user-based and time-dependent features. Further, Ding et al. (2019b) also show the effectiveness of their DNN regression model not only for image popularity, but also for popularity prediction on social media. While their previous findings on images also play a major role, they again consider features like the tags, temporal, and user features. Additionally, they use a pre-trained BERT model to obtain deep text features. They come to the conclusion that the deep image and text features, as well as user features, have the greatest influence on popularity.

Another common approach to popularity assessment is to estimate the popularity of a post based on the influence the posting user has, similarly to the approach from Castillo et al. (2011) introduced in Section 3.1.2. For example, Nargundkar and Rao (2016) propose a machine learning-based system called "InfluenceRank" to predict the influence of a Twitter user. Therefore, the authors consider a range of features, such as the number of tweets, followers, and public lists the user is on. Further, they consider a series of ratios related to the number of favourites and retweets their tweets obtain, as well as the ratio of followers to the number of people who follow the user. Labelled data was obtained from a survey. Therefore, a regression model was fitted to the data for feature selection, yielding the feature contribution weights as a result. With these results, the authors trained an SVM-based model to predict the influence of unknown users, achieving decent accuracy. However, the trained model was suffering from overfitting due to the relatively small set of training data.

Muñoz-Expósito et al. (2017) address customer engagement for marketing in particular. They propose a measurement for engagement for Twitter. While the authors provide a variety of concepts regarding engagement, we will only focus on their work

concerning engagement on tweets and therefore ignore approaches for measuring user engagement. Further, their remarks about company management and marketing are of no interest to our study and are therefore omitted in the following. To get started, the authors list a series of quantifiable "actions of interest" such as, for instance, clicks on embedded media, hashtags or links, the number of times shared via email, or the number of likes, replies, and retweets. However, these quite detailed statistics are only accessible to the author of the post and therefore are not obtainable to the public via the Twitter API. Nevertheless, the authors propose a general way to describe engagement, simply phrased as the ratio of the number of interactions to the reach of the post. In our case, when not being able to access the advanced tweet metrics, the number of likes and retweets can be considered a suitable fit for the interactions of a tweet, as well as the author's number of followers as the reach of a posted tweet. Thus, the proposed metric represents a reach-normalized ratio to precisely measure the engagement and, therefore, popularity of a tweet.

4 Methods

We investigate the research hypothesis stating whether false information spreads more successfully on social media than valid information. Therefore, a first approach is introduced to provide insights into this unresolved research area. The main concept of the approach consists of an analysis of claims towards their veracity to ultimately investigate a possible correlation between the estimated veracity and the propagation metrics of those claims.

4.1 Pipeline

In the following, a pipeline is described to model the approach and its parameters. The pipeline identifies documents containing claims from a given set of documents. From the resulting subset of all claims, it produces subsets of both documents containing true and documents containing false claims. Figure 1 shows an overview of the structure of the pipeline. Therefore, the main dataset consists of social media posts, this is the data we want to analyse towards our research hypothesis. The training dataset for *identifying claims* consists of posts labelled as claim or non-claim. For *determining claim veracity*, the training dataset is composed of claims labelled with true or false veracity. It is of high importance that these labels are accurately fact-checked as they form the basis on which the model estimates the veracity of the unseen data. The main dataset and the training datasets are distinct. For further details on data creation see Section 5.1.1. Finally, the subsets of estimated true and false claims can then be used for the evaluation of their propagation metrics, respectively.

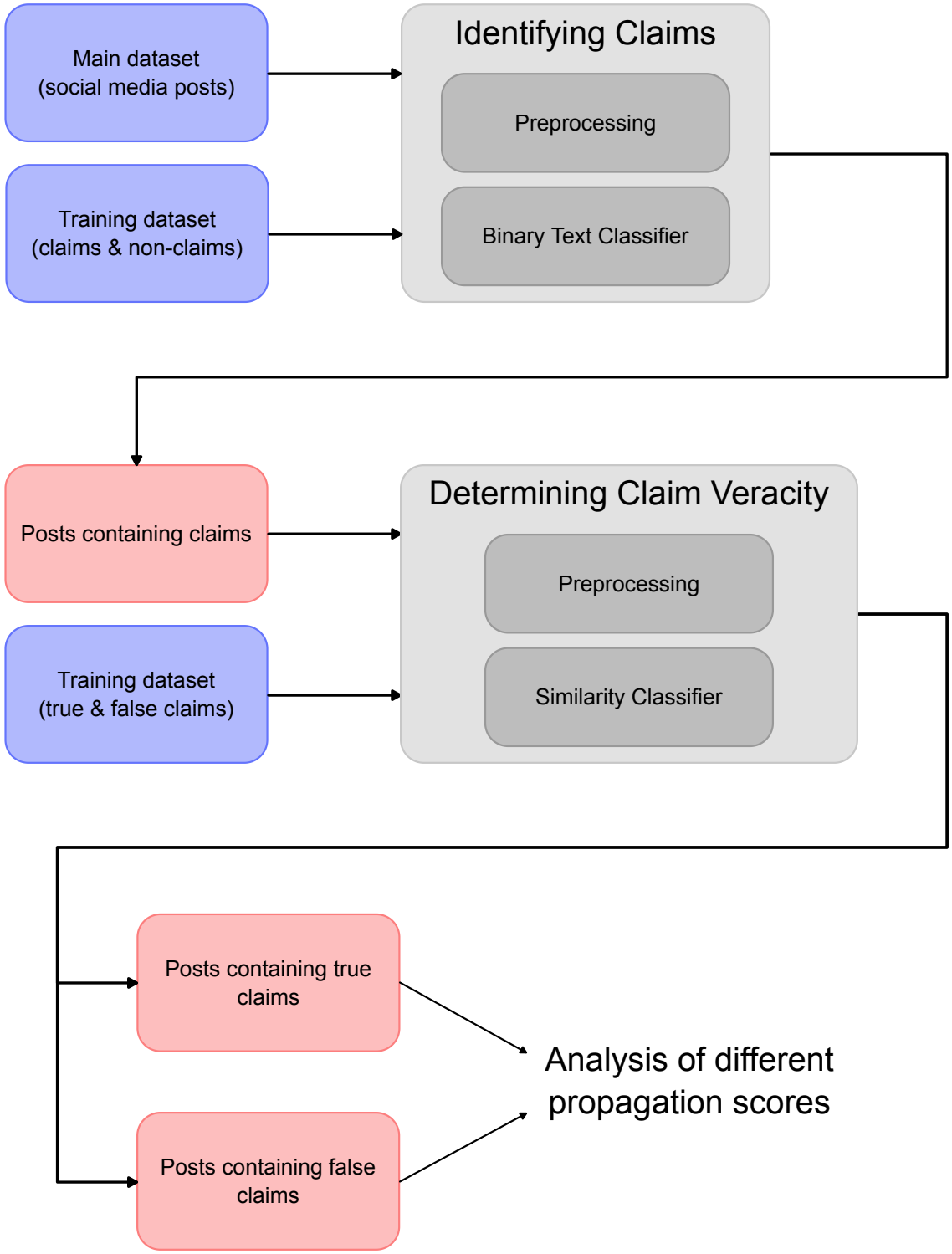

Figure 1: The pipeline for separately extracting posts from a dataset that contain true claims and posts that contain false claims.

4.1.1 Identifying Claims

As a first step, we need to identify the documents containing claims among the entire dataset. To solve the task of claim identification, we opt for an approach similar to the work from Lippi and Torroni (2015). Despite more sophisticated systems for claim detection, such as RNN and GAN, performing well in previous work in Section 3.1.1, we decide to rely on a basic approach which also produced decent results. The reasoning for this choice lies in the assumption that our research goal aims at merely exploring the existence of a possible connection between the veracity and the engagement of posts. We therefore suggest the deployment of more advanced methods for future research (see Section 6.1).

Hence, a binary claim detection model is trained on positive and negative examples of claims. The model divides the total set of tweets into subsets for claims and non-claims, respectively. The first module of the pipeline takes a dataset of documents as well as an annotated training dataset of claims and non-claims as inputs to yield a subset of claims as a result.

In more detail, the chosen binary text classification model is being fitted to the underlying training set of documents to learn the connection between the vector representation of a document and whether that document contains a claim or not. After fitting, the model can perform predictions for unseen documents based on its training experience. Thus, the model can decide for each document in the main dataset whether it is a claim or not. Afterwards, the documents are annotated accordingly so that the claims can be extracted in the next step.

4.1.2 Determining Claim Veracity

The second module of the pipeline is veracity estimation. It aims at assigning each previously identified claim either true or false veracity based on its similarity to a set of claims whose truthfulness is known and validated. This represents once more a rather primitive approach to this task compared with the approaches used in previous work (see Section 3.1.2), as we assessed user-based credibility analysis

as not suitable and do not have the resources needed to perform in-depth fact-checking to obtain training and validation data samples from our data necessary for approaches relying on neural networks. Further, we have seen that classic models like SVM also work well, so we assume our simple approach of estimating the veracity via similarity should be sufficient to show a strong correlation if one exists.

The subset of claims produced previously by the first module serves as input for this module. Further, additional training data is to be provided for the similarity classification process. The process is subject to the following concept: The found claims are transformed into some form of vector representation, so-called embeddings. The same is applied to the true and false claims in the training dataset. Now, for each pair of vectorized documents, a similarity score can be calculated according to the metric of choice. Metrics for sentence similarity can be distinguished into either similarity-based or distance-based metrics between vectors. Both kinds of metrics enable us to establish a ranking of the top k most similar documents for each training claim. In our work, we use the cosine-similarity as a metric to collect the 1000 most similar claims ($k = 1000$). It is computed as the cosine of the angle between two vectors. The cosine works great as a similarity metric because the cosine of an angle is highest when two vectors point in the same direction while it is lowest when they are exactly opposite. It is computed as follows for two vectors A, B of length n with angle α:

$$\cos(\alpha) = \frac{A \cdot B}{||A||\,||B||} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}}.$$

In the next step, the claims are classified as either true or false, depending on the annotated veracity of the training claim. Lastly, the top k rankings for each of the training claims can be combined into subsets of true and false claims as an output of this module.

4.2 Hyperparameters

The pipeline involves a variety of hyperparameters for fine-tuning. In Figure 1 there are different options for the subtasks visualized within the two modules (in grey) *identifying claims* and *determining claim veracity*. This includes choices for preprocessing and the choice of the underlying model for the module. For the final results there is a variety of ways to calculate the propagation score associated with a post which we analyse and compare. Furthermore, input datasets are illustrated in blue while module outputs are illustrated in red.

4.2.1 Preprocessing

The original documents from the main dataset might need to be preprocessed before they can be fed to the classification modules. Two impactful parameters are how to deal with stopwords and the way the documents are being tokenized.

First, so-called stopwords are common words that do not add any value to the meaning of a sentence. If we take a look at our previous example sentence, *"The weather was sunny yesterday in Stuttgart."*, the words *the*, *was*, and *in* are rather meaningless in contrast to the core keywords *weather*, *sunny*, *yesterday*, and *Stuttgart*. Often, stopwords are omitted as it is assumed that the tiny bit of valuable contextual information they might carry is irrelevant. However, Uysal and Gunal (2014) have shown that the removal of stopwords does not necessarily provide better results. They conclude that stopwords should not be removed at all, independent of the domain and language. Nevertheless, the choice to either keep or omit stopwords in both modules of the pipeline individually is provided.

Secondly, sentences need to be separated into tokens, which in most cases is equivalent to splitting them into all separate words. Once again, word plurals or verb conjugations mostly do not affect the meaning of a sentence in a significant way. Therefore, it is desired to normalize all occurrences of these forms to just one form each. The process of lemmatization serves this exact purpose, mapping the word variations to just one singular, potentially mangled, form. For example, *children* be-

comes *child*, *flying* is mapped to *fly*, or *was* is turned into *be*. As Uysal and Gunal (2014) found, the effect of lemmatization is highly dependent on the domain and language. Thus, it also provides the choice to either keep tokens as they can be found originally or make use of lemmatization when tokenizing documents.

4.2.2 Text Classification Models

The pipeline includes different classification models for the first module to determine whether a document contains a claim. It provides the hyperparameter choice of three models: random forest, SVM, and LightGBM. Random forest acts as a suitable model that performed well enough in previous work, as seen in Section 3 while sticking to a simpler and more comprehensible approach than better-performing neural network variations. The same goes for SVM as a classic, universally well-performing classifier. The models AutoML random forest and AutoML LightGBM were selected as two of the best-performing models in terms of accuracy by the Automated ML (AutoML) feature of the Microsoft Azure Machine Learning Studio[11]. It enables users to start a so called Automated ML run on a dataset. After specifying the target column and percentage of validation data for the train-validation split, AutoML automatically runs and tunes a series of classification models. As a result, all trained models are presented with their scored results and can be exported for further implementation. Note that these models are only trained in the Azure AutoML environment and cannot be manually trained within the pipeline.

Similar to the variety of choices for claim identification, there are two hyperparameters for claim veracity estimation. The pipeline includes a similarity classifier model based on GloVe as well as a BERT-based model using embeddings.

[11] https://ml.azure.com/, https://docs.microsoft.com/en-us/azure/machine-learning/overview-what-is-machine-learning-studio

4.2.3 Propagation Scores

The propagation score is a metric to precisely measure *engagement* on social media posts. The term *engagement* for a post refers to user interaction. In the context of Twitter, this involves especially the number of favourites ("likes") and retweets of a tweet, but technically also any interaction within replies to said tweet and even further the number of times the tweet has been shared otherwise outside of Twitter. As we have seen in previous work (see Section 3.2), user-based features and the upload time also play an important role for the popularity of a post. For the sake of simplicity, we only focus on the number of favourites and retweets of a tweet to quantify the engagement. The following formal notation will be used: $l(t)$ denotes the number of favourites and $r(t)$ represents the number of retweets of a tweet t, the propagation score s of a tweet t will be referred to as $s(t)$.

The most straight-forward metric to measure the engagement of a tweet is simply the number of favourites the tweet has obtained. It holds

$$s_{\text{fav}}(t) = l(t).$$

Secondly, it is evident from the previous metric to additionally consider the number of retweets of the tweet. To start with, these measures are summed up equally; a weighted sum poses an alternative if desired. Therefore, it holds

$$s_{\text{sum}}(t) = l(t) + r(t).$$

The measures of favourites and retweets are highly dependent on the reach of the tweet due to the way Twitter presents posts on its social media platform: Direct followers of a user will always find the user's tweets in their "timeline" (a feed of posts considered relevant for the user by a Twitter algorithm), while users close to the followers might see the post only occasionally as a recommendation, despite not even being a direct follower of the initial user. This is further increased by followers liking, retweeting, or replying to the post. Therefore, the number of followers of the author of a tweet can be considered as a sufficient measure for the reach of a tweet to normalize the previous metrics. We denote the number of followers of the user

who posted the tweet t as $f(t)$. With these three measures, the propagation score can be computed as a ratio of interactions to the number of followers, similar to the way described by Muñoz-Expósito et al. (2017):

$$s_{||\cdot||}(t) = \frac{l(t) + r(t)}{f(t)} \cdot 100, \qquad f(t) \neq 0.$$

If the author has no followers ($f(t) = 0$), we consider $f(t) = 1$ instead. Note that this score does not distinguish between a tweet being an original post or a retweet of a post, as the Twitter API does not directly distinguish between those cases and their metrics, such as the number of favourites or retweets. Therefore, posts from users with few followers of their own achieve an outstandingly high score when retweeting popular posts with a large number of retweets. Further, the original popular post might have a noticeably smaller score due to being normalized to the presumably larger number of followers of its author. The outliers resulting from this effect should balance out when comparing the average scores of multiple datasets. However, this does not apply if a dataset features a particularly large share of retweets.

5 Evaluation

5.1 Experimental Setup

5.1.1 Datasets

The main dataset used in the experiment consists of 1 559 575 tweets related to the COVID-19 pandemic from the COVID-19 Twitter chatter dataset (Banda et al., 2021). The chatter dataset contains all tweets from the official Twitter COVID-19 Twitter stream every day. The English tweets published on the first of each month in 2021 are chosen as a sample to cover a broader spectrum of tweets and topics rather than just analysing the tweets of twelve consecutive days. The tweets are extracted using the social media mining toolkit (SMMT) by Tekumalla and Banda (2020).

To obtain a baseline for the propagation scores, the English tweets of one day (5th July, 2021) were extracted from a dataset from the Internet Archive[12]. The SMMT was used for the extraction process as before. This provides the propagation scores of 2 143 717 tweets for comparison. Table 2 gives an overview to the statistics of both datasets. For the baseline, the statistics about average number of words and characters were not collected.

The training dataset for the task of identifying claims is constructed from two existing COVID-19-related datasets. The English data from the Infodemic dataset (Alam et al., 2021a;b) contains 504 tweets. Each tweet is already annotated as either a claim or non-claim, for a total of 305 claims and 199 non-claims. Additionally,

[12] https://archive.org/details/archiveteam-twitter-stream-2021-07

Dataset	Tweets	Words	Characters	s_{fav}	s_{sum}	$s_{\|\|\cdot\|\|}$
Main (COVID-19) dataset	1 559 575	23.308	167.688	9.858	12.612	1.574
Baseline Twitter sample	2 143 717	-	-	3.047	6 654.360	44 061.867

Table 2: Average number of words, characters, and average propagation scores in the main and baseline datasets, all values rounded to three digits

the Covidlies dataset (Hossain et al., 2020) provides 62 different misconceptions towards COVID-19, collected from 6692 tweets. The misconceptions from the Covidlies dataset all serve as positive examples for claims (regardless of their false veracity). Therefore, the resulting training dataset consists of 367 claims and 199 non-claims.

Further, for the task of determining claim veracity, a set of true and false annotated claims connected to COVID-19 forms the training dataset. The false claims mainly originate from the Covidlies dataset (Hossain et al., 2020), while the true claims are collected from official German and US government information websites about COVID-19[13]. To compensate for the imbalance of too many true claims in the training dataset, we negate a correspondingly large number of facts found on these websites and annotate them as false. Furthermore, we add alternative formulations, keeping the core statements of all false and true claims to the training data, making it easier to find similar tweets for the corresponding claims and enlarging the training dataset to counteract overfitting. In total, the training dataset consists of 28 claims with true veracity and 27 claims with false veracity. Table 3 provides an overview of the class distributions of both training datasets.

5.1.2 Parameter Choices

For the experiment, the following choices have been made for the areas of Preprocessing (Section 4.2.1), Text Classification Models (Section 4.2.2), and Propagation Scores (Section 4.2.3).

[13] https://www.zusammengegencorona.de/informieren/, https://www.cdc.gov/coronavirus/

Dataset	Statements	Claims	Non-claims	True	False
Claim detection training dataset	566	367	199	-	-
Claim veracity training dataset	54	54	0	28	27

Table 3: Number of statements and class distributions in the training datasets

Keeping all contained stopwords and lemmatizing resulting tokens are chosen as default preprocessing options intuitively for both modules *identifying claims* and *determining claim veracity*. This conforms to the recommendations from (Uysal and Gunal, 2014). The decision to keep stopwords is justified by the fact that we necessarily require such stopwords such as *not* that do provide the relevant context needed for our task. This is illustrated by the following preprocessed example where the presence of the stopword *not* has a decisive influence on the meaning of the sentence. Both *"injecting disinfectant is not an effective treatment against covid"* and *"injecting disinfectant is an effective treatment against covid"* produce *"injecting disinfectant effective treatment covid"* when removing stopwords, despite the contradictory statements. Further, lemmatizing tokens produced adequate results in manually reviewed test samples.

For both identifying claims and determining claim veracity, we use all available text classification model options. The four classifiers provide us with a range of different approaches to compare against. The diversity of using both GloVe and BERT enables exploration of the effects on the correlation analysis results of each approach.

Specifically, for our SVM model, we use a RBF kernel with a scaling γ kernel coefficient. The random forest model uses 200 estimators and measures the quality of a split with the entropy information gain. We consider a maximum number of $\log_2$ features when looking for the best split. The AutoML random forest model uses only 100 estimators, the gini impurity to measure split quality, and the square root number of features when looking for the best split. For the AutoML LightGBM

model we use 100 estimators with 119 leaves and a maximum depth of eight.

As the different propagation scores all have their own reasoning, naturally, we want to consider all options for the analysis here as well.

5.1.3 Implementation Details

The entire experiment is implemented in Python 3.7.9. For efficient processing and optimized runtime storage of the datasets, the `pandas` package (Wes McKinney, 2010) was used. The preprocessing of the datasets is performed with the `nltk` package (Bird et al., 2009) for removing stopwords, tokenizing, and lemmatizing.

The `scikit-learn` package (Pedregosa et al., 2011) was used for the SVM and random forest claim classification models.

For the module of determining claim veracity, the `gensim` package (Řehůřek and Sojka, 2010) was used for a GloVe implementation based on the pre-trained `glove-wiki-gigaword-50` model. Further, the BERT implementation relies on the `sentence-transformers` package (Reimers and Gurevych, 2019), using the pre-trained `msmarco-MiniLM-L6-cos-v5` model.

To maximize performance and minimize disk storage usage, the implementation uses an index-based system to store all subsets of the main dataset used in the process. Thus, for example, only a list of indices of the claims classified by the first module needs to be stored persistently to then look up the entire metadata of the corresponding indices in the main dataset whenever it is required.

5.2 Results

First, we compare the results for claim classification of the four models: random forest, SVM, AutoML random forest, and AutoML LightGBM obtained with the experimental setup described in Section 5.1. We split the training dataset into a subset used for training (70%) and a subset for testing (30%). To obtain the following results, we evaluate the performance on the test subset. Note that we only

Model	Class	Accuracy	Precision	Recall	F_1-Score
Random Forest	C	0.7485	0.78	0.87	0.83
	N		0.63	0.48	0.55
SVM	C	0.7427	0.82	0.81	0.82
	N		0.57	0.59	0.58
AutoML Random Forest	C	0.6784	0.81	0.68	0.74
	N		0.51	0.68	0.59
AutoML LightGBM	C	0.7836	0.83	0.88	0.85
	N		0.65	0.56	0.60

Table 4: Claim identification results (C: Claim; N: Non-claim)

analyse results obtained with the chosen preprocessing hyperparameters of keeping all stopwords and lemmatizing all tokens based on the reasoning described in Section 5.1.2.

Thus, we obtain the results shown in Table 4. The AutoML LightGBM model outperforms the other models with an accuracy (often also referred to as "micro average F_1") of 78.36% significantly. The F_1-Score for the claim class is particularly relevant as we only want to extract the found claims in the next step. As a result, the AutoML LightGBM model scores 0.85, slightly higher than random forest's 0.83 and SVM's 0.82. The AutoML random forest model yields the worst results of the four analysed models. Despite the quite low accuracy of 67.84% it still achieves a F_1-Score of 0.74 for the claim class. In total, the results obtained are close to those of Ma et al. (2019). Thus, all four models show sufficient performance. The class distribution for the different models is shown in Table 5.

Table 6 illustrates a random sample of tweets classified as claims by our best-performing model AutoML LightGBM. The tweets have been hand-selected and are presented in no particular order.

Model	Claims	Non-claims	s_{fav}	s_{sum}	$s_{\|\|\cdot\|\|}$
Random Forest	1 010 918	548 657	11.703	15.057	1.876
SVM	1 161 407	398 168	10.818	13.913	1.374
AutoML Random Forest	671 173	888 402	11.983	15.368	1.549
AutoML LightGBM	939 194	620 381	11.629	14.976	1.967

Table 5: Class distribution and average propagation scores for different claim detection models, all values rounded to three digits

Further, we cannot provide results for the accuracy of our similarity classifiers, because we do not have the resources to create sufficiently large test and validation subsets of our data, as this would require in depth fact-checking for accurate annotation. Instead, we take one step forward and analyse the average propagation scores for each of the true and false claim subsets we extracted for every model combination. Additionally, we compare the average scores with baselines computed for the entire COVID-19-related main dataset as well as the one-day sample from general Twitter. Table 7 provides the results, including the number of unique true and false claims extracted.

The average values of s_{sum} and $s_{\|\cdot\|}$ for the Twitter sample are particularly striking as they are extraordinarily high in comparison to all other results. This effect is a side-effect of the construction of $s_{\|\cdot\|}$ as already previously described in Section 4.2.3: The Twitter API returns the number of retweets of a tweet not only for the tweet of the initial author but also for the retweeted posts of all users retweeting the original post. Note, that this effect does not only appear in the Twitter sample but in all of the data. This suggests that there are significantly fewer highly popular posts being retweeted many times in the COVID-19-related domain.

Further, Figure 2 illustrates the obtained results graphically as a bar chart grouped by the model combination. The (extremely high) baselines of s_{sum} and $s_{\|\cdot\|}$ for the

Claim	s_{fav}	s_{sum}	$s_{\|\|\cdot\|\|}$
@USER @USER @USER you spelled it wrong — it's plandemic!! not pandemic lol wow … covid19 is merely the same symptoms as the influenza virus .. in case you missed it how many people did not catch the flu or was prescribed the tamiflu??	17	17	11.333
with vaccines, we should still be alert. no time for complacency. #covid19	0	0	0
great. there's a new covid strain that spreads germans now. nein. URL URL	5	6	2.001
have biden and harris eliminated systemic racism and white supremacy like they did covid19 deaths?	64	104	0.503
#coronavirus: 2,526 new cases from additional 175,033 tests reported in the uae • 1,107 recoveries • 17 deaths • 382,332 total recoveries • 394,050 total cases • 1,238 total deaths follow the latest #covid19 developments here: URL URL	5	9	0.001
@USER @USER masks reduce droplet transmission. covid-19 is transmitted by respiratory droplets (and aerosols) if you choose not to wear one (rather than can't wear one for whatever reason) you increase risk of transmission and community spread. be part of the solution not the problem	0	0	0
reality check @USER #lie : 5g mobile networks do not spread covid-19. #coronavirus #moronavirus	0	0	0
have you had the micro-chip ? take a look, where the jab was given, it's magnetic now ... . . [sc: the interwebs] #vaccine #billgatesbioterrorist #covid19 #nuremberg2 #plandemic #who #depopulationagenda #drfauci #vaccinepassports #wef URL	1	1	0.433
omg its offical guys coronavirus is over	166	176	0.904

Table 6: Sample tweets and their propagation scores, which have been classified as claims by the AutoML LightGBM model (removed URLs and emoticons, anonymized user mentions)

Model		Claims		s_{fav}		s_{sum}		$s_{\|\|\cdot\|\|}$	
Identifying claims	Veracity	#true	#false	true	false	true	false	true	false
Random Forest	BERT	19 934	19 173	11.529	13.277	14.304	16.428	1.214	1.473
	GloVe	24 838	21 351	8.328	8.519	10.820	11.122	1.303	1.459
SVM	BERT	20 173	19 430	7.538	9.476	9.784	12.271	2.674	1.122
	GloVe	24 964	21 418	11.780	9.450	14.828	12.264	1.902	1.215
AutoML Random Forest	BERT	19 025	18 479	11.220	8.039	13.901	10.338	1.332	1.052
	GloVe	24 439	20 814	9.358	10.892	11.806	13.412	0.965	1.203
AutoML LightGBM	BERT	19 738	18 924	10.054	9.259	13.090	11.969	1.639	1.415
	GloVe	24 786	21 134	9.064	12.252	11.635	15.815	1.584	2.172
Main (COVID-19) dataset		1 559 575		9.858		12.612		1.574	
Twitter sample		2 143 716		3.047		6 654.360		44 061.867	

Table 7: Average propagation scores for both classes for different claim detection and veracity estimation models, all values rounded to three digits

Twitter sample are omitted because they add no value to the comparison. In the following, we compare all the different propagation scores separately.

Favourites s_{fav} In three of the eight cases, the scores are significantly higher for the subset of true claims, while in four of the eight cases the scores for false claims are higher. For the combination of random forest with GloVe, the scores are balanced out. Comparing the score values to the main dataset baseline shows that overall the score of claims is slightly lower than on average, with true and false claims exceeding the dataset average of 9.858 evenly in only seven out of sixteen cases. The s_{fav} baseline of general Twitter, however, is greatly exceeded for every model.

Sum of Favourites and Retweets s_{sum} The distribution of s_{sum} for true and false claim subsets is similar to s_{fav} as to be expected regarding the computation of the two scores. Thus, three cases feature higher scores for true claims, while four cases feature higher scores for false claims and they are balanced once. Unsurprisingly, the dataset baseline for this score has a similar relationship to the dataset's s_{fav}

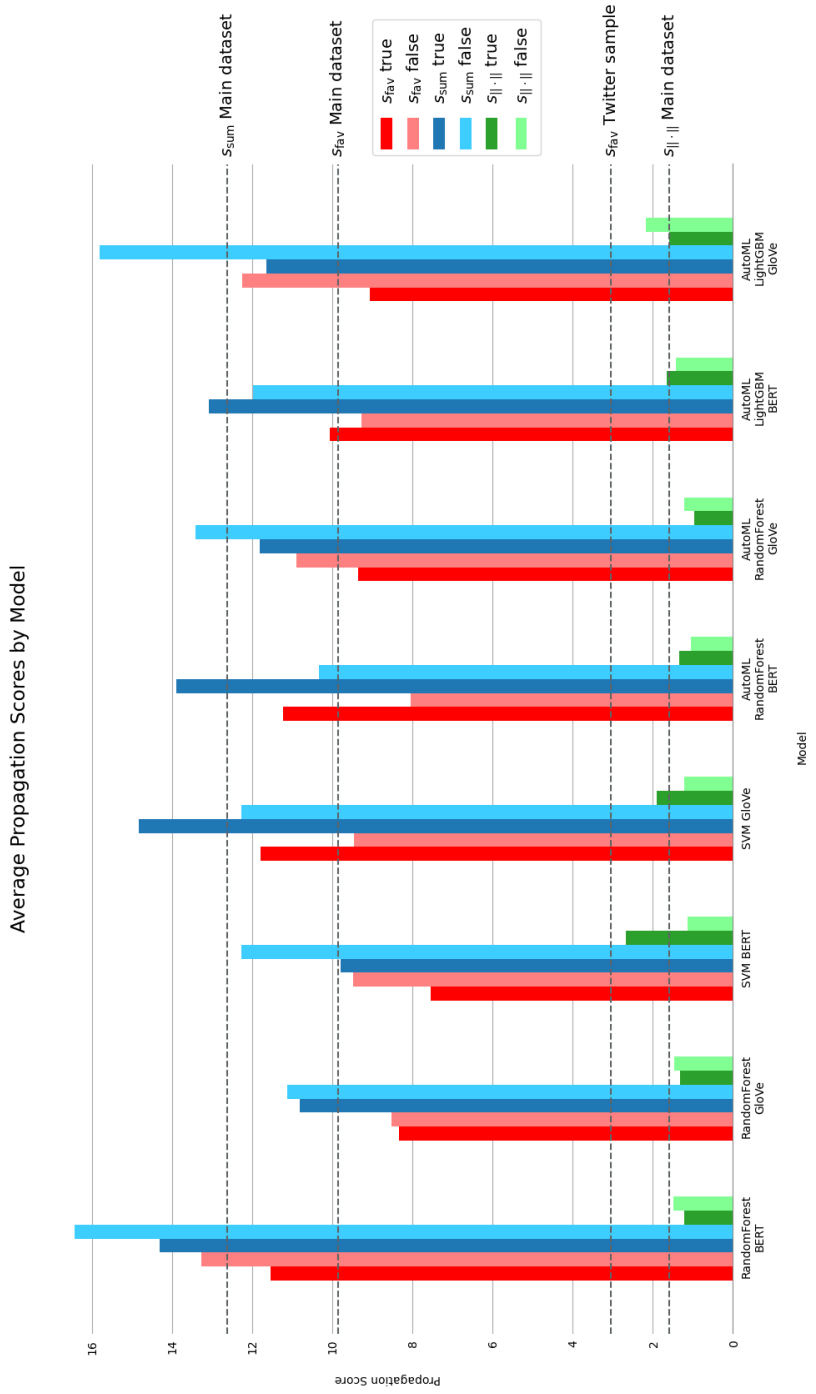

Figure 2: Average propagation scores with baselines for the COVID-19 themed main dataset and a general Twitter sample

baseline, with subsets exceeding it nearly half of the time.

Normalized $s_{\|\cdot\|}$ The normalized scores are more balanced overall. Significant differences between true and false are only seen for the two SVM models where true claims achieve higher scores and the AutoML LightGBM GloVe model where the opposite takes place. The majority of normalized scores for $s_{\|\cdot\|}$ are lower than the main dataset baseline, with only the significant differences exceeding it noticeably.

Regarding the grouped scores for the models, it stands out that the ratio of true to false claims remains consistent regardless of the underlying propagation scores. The only exception to this observation is made by $s_{\|\cdot\|}$ for the SVM BERT model combination.

Further, there cannot be observed any correlation between the model used for determining claim veracity and the obtained results. Just like BERT-based models, GloVe-based models provide higher scores for false claims in half of the cases. However, the choice between BERT and GloVe for the same claim identifying model always inverts the results, except when using the random forest model.

In conclusion, the observed results generally do *not* show false claims achieving higher propagation scores and therefore receiving more engagement than posts containing false claims.

5.3 Discussion

To exclude the possibility that the previously shown results contain artefacts, a manual qualitative analysis of the top 100 claims with the highest propagation scores was performed. Within the analysis, no major artefacts were found. However, a large portion (about 70%) of tweets were no longer accessible via the Twitter website. This observation has no impact on the results of the experiment as the metadata (text, favourites, retweets, followers of author) of the non-accessible posts could still be obtained through the Twitter API via their id. Further, in the qualitative analysis, several tweets with political context were observed. Tweets by politicians

and celebrities make up a solid share of the total. Tweets from dedicated news accounts are also common. Once again, the effect of popular retweets by users with few followers themselves can be seen, particularly for tweets analysed within the top 100 for $s_{||\cdot||}$, among those also one with artificially increased propagation, challenging the readers to balance favourites and retweets.

Overall, it is observed that claims extracted by the first module contain many news headlines or headline-like tweets. These claims often have duplicates, for example, resulting from a major news agency reporting a news headline that is being shared and retweeted among multiple other news accounts.

The obtained results might be noisy due to the relatively small length of tweets. Therefore, most tweets do not provide much context that the classification models could pick up, especially for the process of similarity classification to known true or false claims. Furthermore, because they are designed for larger documents, the BERT and GloVe document embeddings may produce poor results for such small document sizes.

Further, it is most likely that the similarity classifier for differentiation between true and false claims does not provide the best results due to overfitting. This is no surprise given the relatively small training set of known true and false claims.

The sample of tweets classified as claims in Table 6 strengthens the observation of the appearance of many headline-like posts. Furthermore, it shows, that posts with irony as well as sarcasm are (correctly) classified as claims. Therefore, these posts represent a problem for the module of veracity estimation as it cannot detect irony and sarcasm.

6 Future Work and Conclusion

6.1 Future Work

The results of the experiment open up numerous new approaches towards the analysis of the potential correlation between veracity and the propagation of posts. It is highly desirable to perform the analysis for different topic domains than COVID-19 and compare the results with those we found. Also, the exploration of the impact of different, possibly more advanced classification models, such as neural network-based systems, which proved quite successful in previous work, represents a clear task for a follow-up approach. Furthermore, expanding research on other social media channels besides Twitter, such as Telegram, is definitely worth investigating. This opens up research questions such as: Are false claims receiving more engagement on Telegram than on average on other social media platforms? Do fake news spread more successful on Facebook than on Twitter?

Further, small and simple additions could be made to the claim classification. Irony presents an impossible challenge to our approach to veracity determination, so an irony detector would be a helpful addition to the pipeline. An additional module to classify and omit news-like posts could potentially improve the results. However, the use of such a classifier can come with the cost of ignoring deliberately spread disinformation phrased in similar language. Therefore, this presents us with the research question: Do false social media posts phrased in news-like language receive more user engagement than false posts in non-news-like language? Furthermore, news posts do not necessarily always contain facts, considering corrupt, government-controlled, or censored media in some countries.

Most importantly, the process of dividing the claims into presumably true and false claims based on similarity to known claims can most likely be improved significantly by implementing a proper fact-checking mechanism. Hereby, a methodology to determine the check-worthiness of each post is most likely required. Alternatively, providing the classifier with more training data should counteract the current problem of overfitting.

Lastly, self-propagating social media bubbles represent a closely related task to the research hypothesis. Thus, diving into a deeper analysis of these bubbles poses an aspirational follow-up task, as especially right-wing extremist social bot bubbles pose a threat to modern society by spreading disinformation (Ferrara et al., 2020). This poses the research question: Do posts from self-propagating social media bubbles receive more user engagement than average Twitter posts?

6.2 Conclusion

We proposed a methodology to explore a potential correlation between the veracity of social media posts and their propagation. Therefore, a pipeline has been developed to identify claims within a set of posts and estimate their veracity based on similarity to known-veracity-claims. An analysis of a range of experimental hyperparameters has shown no evidence of false claims receiving a higher amount of engagement but poses multiple starting points for further research.

Bibliography

Firoj Alam, Fahim Dalvi, Shaden Shaar, Nadir Durrani, Hamdy Mubarak, Alex Nikolov, Giovanni Da San Martino, Ahmed Abdelali, Hassan Sajjad, Kareem Darwish, and Preslav Nakov. Fighting the COVID-19 infodemic in social media: A holistic perspective and a call to arms. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 15 of *ICWSM '21*, pages 913–922, May 2021a. URL https://ojs.aaai.org/index.php/ICWSM/article/view/18114.

Firoj Alam, Shaden Shaar, Fahim Dalvi, Hassan Sajjad, Alex Nikolov, Hamdy Mubarak, Giovanni Da San Martino, Ahmed Abdelali, Nadir Durrani, Kareem Darwish, Abdulaziz Al-Homaid, Wajdi Zaghouani, Tommaso Caselli, Gijs Danoe, Friso Stolk, Britt Bruntink, and Preslav Nakov. Fighting the COVID-19 infodemic: Modeling the perspective of journalists, fact-checkers, social media platforms, policy makers, and the society. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 611–649, Punta Cana, Dominican Republic, November 2021b. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-emnlp.56. URL https://aclanthology.org/2021.findings-emnlp.56.

Hunt Allcott, Matthew Gentzkow, and Chuan Yu. Trends in the diffusion of misinformation on social media. *Research & Politics*, 6(2):2053168019848554, 2019. doi: 10.1177/2053168019848554. URL https://doi.org/10.1177/2053168019848554.

G.W. Allport and L.J. Postman. *The Psychology of Rumor*. The Psychology of Rumor. Russell & Russell, 1965. URL https://books.google.de/books?id=N6O4AAAAIAAJ.

Juan M. Banda, Ramya Tekumalla, Guanyu Wang, Jingyuan Yu, Tuo Liu, Yuning Ding, Ekaterina Artemova, Elena Tutubalina, and Gerardo Chowell. A large-scale covid-19 twitter chatter dataset for open scientific research—an international collaboration. *Epidemiologia*, 2(3):315–324, 2021. doi: 10.3390/epidemiologia2030024. URL https://www.mdpi.com/2673-3986/2/3/24.

Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python: analyzing text with the natural language toolkit.* " O'Reilly Media, Inc.", 2009. doi: 10.1017/S1351324910000306. URL https://doi.org/10.1017/S1351324910000306.

Michael H. Birnbaum and Steven E. Stegner. Source credibility in social judgment: Bias, expertise, and the judge's point of view. *Journal of Personality and Social Psychology*, 37(1):48, 1979. URL https://psych.fullerton.edu/mbirnbaum/papers/Birnbaum_Stegner_JPSP_1979.pdf.

Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001. doi: 10.1023/A:1010933404324. URL https://doi.org/10.1023/A:1010933404324.

Leo Breiman, J.H. Friedman, R.A. Olshen, and C.J. Stone. Classification and regression trees. 1984. doi: 10.1201/9781315139470. URL https://doi.org/10.1201/9781315139470.

Kevin R. Canini, Bongwon Suh, and Peter L. Pirolli. Finding credible information sources in social networks based on content and social structure. In *2011 IEEE Third International Conference on Privacy, Security, Risk and Trust and 2011 IEEE Third International Conference on Social Computing*, pages 1–8. IEEE, 2011. doi: 10.1109/PASSAT/SocialCom.2011.91. URL https://doi.org/10.1109/PASSAT/SocialCom.2011.91.

Carlos Castillo, Marcelo Mendoza, and Barbara Poblete. Information credibility on twitter. pages 675–684, 01 2011. doi: 10.1145/1963405.1963500. URL https://doi.org/10.1145/1963405.1963500.

Leon Derczynski, Kalina Bontcheva, Maria Liakata, Rob Procter, Geraldine Wong Sak Hoi, and Arkaitz Zubiaga. SemEval-2017 task 8: RumourEval: Determining rumour veracity and support for rumours. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 69–76, Vancouver, Canada, August 2017. Association for Computational Linguistics. doi: 10.18653/v1/S17-2006. URL https://aclanthology.org/S17-2006.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423.

Keyan Ding, Kede Ma, and Shiqi Wang. Intrinsic image popularity assessment. In *Proceedings of the 27th ACM International Conference on Multimedia*, MM '19, page 1979–1987, New York, NY, USA, 2019a. Association for Computing Machinery. doi: 10.1145/3343031.3351007. URL https://doi.org/10.1145/3343031.3351007.

Keyan Ding, Ronggang Wang, and Shiqi Wang. Social media popularity prediction: A multiple feature fusion approach with deep neural networks. In *Proceedings of the 27th ACM International Conference on Multimedia*, MM '19, page 2682–2686, New York, NY, USA, 2019b. Association for Computing Machinery. doi: 10.1145/3343031.3356062. URL https://doi.org/10.1145/3343031.3356062.

Emilio Ferrara, Herbert Chang, Emily Chen, Goran Muric, and Jaimin Patel. Characterizing social media manipulation in the 2020 U.S. presidential election. *First Monday*, 25(11), Oct. 2020. doi: 10.5210/fm.v25i11.11431. URL https://journals.uic.edu/ojs/index.php/fm/article/view/11431.

Genevieve Gorrell, Elena Kochkina, Maria Liakata, Ahmet Aker, Arkaitz Zubiaga, Kalina Bontcheva, and Leon Derczynski. SemEval-2019 task 7: RumourEval,

determining rumour veracity and support for rumours. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 845–854, Minneapolis, Minnesota, USA, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/S19-2147. URL https://aclanthology.org/S19-2147.

Mark Graham and William H. Dutton. *Society and the internet: How networks of information and communication are changing our lives*. Oxford University Press, 2019. URL https://doi.org/10.1093/acprof:oso/9780199661992.001.0001.

Andrew M. Guess and Benjamin A. Lyons. *Misinformation, Disinformation, and Online Propaganda*, page 10–33. SSRC Anxieties of Democracy. Cambridge University Press, 2020. URL https://doi.org/10.1017/9781108890960.

Shintami Chusnul Hidayati, Yi-Ling Chen, Chao-Lung Yang, and Kai-Lung Hua. Popularity meter: An influence- and aesthetics-aware social media popularity predictor. In *Proceedings of the 25th ACM International Conference on Multimedia*, MM '17, page 1918–1923, New York, NY, USA, 2017. Association for Computing Machinery. doi: 10.1145/3123266.3127903. URL https://doi.org/10.1145/3123266.3127903.

Tamanna Hossain, Robert L. Logan IV, Arjuna Ugarte, Yoshitomo Matsubara, Sean Young, and Sameer Singh. COVIDLies: Detecting COVID-19 misinformation on social media. In *Proceedings of the 1st Workshop on NLP for COVID-19 (Part 2) at EMNLP 2020*, Online, December 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.nlpcovid19-2.11. URL https://aclanthology.org/2020.nlpcovid19-2.11.

Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. LightGBM: A highly efficient gradient boosting decision tree. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 3149–3157, Red Hook, NY, USA, 2017. Curran Associates Inc. URL https://dl.acm.org/doi/pdf/10.5555/3294996.3295074.

Elena Kochkina, Maria Liakata, and Arkaitz Zubiaga. All-in-one: Multi-task learning for rumour verification. In *Proceedings of the 27th International Conference on*

Computational Linguistics, pages 3402–3413, Santa Fe, New Mexico, USA, August 2018. Association for Computational Linguistics. URL https://aclanthology.org/C18-1288.

Ran Levy, Yonatan Bilu, Daniel Hershcovich, Ehud Aharoni, and Noam Slonim. Context dependent claim detection. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1489–1500, Dublin, Ireland, August 2014. Dublin City University and Association for Computational Linguistics. URL https://aclanthology.org/C14-1141.

Marco Lippi and Paolo Torroni. Context-independent claim detection for argument mining. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, page 185–191. AAAI Press, 2015. URL https://www.ijcai.org/Proceedings/15/Papers/033.pdf.

Jing Ma, Wei Gao, Prasenjit Mitra, Sejeong Kwon, Bernard J. Jansen, Kam-Fai Wong, and Meeyoung Cha. Detecting rumors from microblogs with recurrent neural networks. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, IJCAI'16, page 3818–3824. AAAI Press, 2016. URL https://www.ijcai.org/Proceedings/16/Papers/537.pdf.

Jing Ma, Wei Gao, and Kam-Fai Wong. Detect rumors on twitter by promoting information campaigns with generative adversarial learning. In *The World Wide Web Conference*, WWW '19, page 3049–3055, New York, NY, USA, 2019. Association for Computing Machinery. doi: 10.1145/3308558.3313741. URL https://doi.org/10.1145/3308558.3313741.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to information retrieval.* Cambridge: Cambridge University Press, 2008. doi: 10.1017/CBO9780511809071. URL https://doi.org/10.1017/CBO9780511809071.

Michael Mathioudakis and Nick Koudas. TwitterMonitor: Trend detection over the Twitter Stream. In *Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data*, SIGMOD '10, page 1155–1158, New York,

NY, USA, 2010. Association for Computing Machinery. doi: 10.1145/1807167.1807306. URL https://doi.org/10.1145/1807167.1807306.

Ulises A. Mejias and Nikolai E. Vokuev. Disinformation and the media: the case of russia and ukraine. *Media, culture & society*, 39(7):1027–1042, 2017. URL https://doi.org/10.1177%2F0163443716686672.

Miriam Muñoz-Expósito, M. Ángeles Oviedo-García, and Mario Castellanos-Verdugo. How to measure engagement in Twitter: advancing a metric. *Internet Research*, 2017. URL https://doi.org/10.1108/IntR-06-2016-0170.

Ashish Nargundkar and Y.S. Rao. Influencerank: A machine learning approach to measure influence of Twitter users. In *2016 International Conference on Recent Trends in Information Technology (ICRTIT)*, pages 1–6, 2016. doi: 10.1109/ICRTIT.2016.7569535. URL https://doi.org/10.1109/ICRTIT.2016.7569535.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. Scikit-learn: Machine learning in python. *J. Mach. Learn. Res.*, 12(null):2825–2830, nov 2011. URL https://dl.acm.org/doi/pdf/10.5555/1953048.2078195.

Radim Řehůřek and Petr Sojka. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta, May 2010. ELRA.

Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1410. URL https://aclanthology.org/D19-1410.

Ruty Rinott, Lena Dankin, Carlos Alzate Perez, Mitesh M. Khapra, Ehud Aharoni, and Noam Slonim. Show me your evidence - an automatic method for context dependent evidence detection. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 440–450, Lisbon, Portugal, September 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1050. URL https://aclanthology.org/D15-1050.

Ramya Tekumalla and Juan M. Banda. Social media mining toolkit (SMMT). *Genomics & informatics*, 18(2), 2020. URL https://doi.org/10.5808/GI.2020.18.2.e16.

Alper Kursat Uysal and Serkan Gunal. The impact of preprocessing on text classification. *Inf. Process. Manage.*, 50(1):104–112, jan 2014. doi: 10.1016/j.ipm.2013.08.006. URL https://doi.org/10.1016/j.ipm.2013.08.006.

William Yang Wang. "liar, liar pants on fire": A new benchmark dataset for fake news detection. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 422–426, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-2067. URL https://aclanthology.org/P17-2067.

Wes McKinney. Data Structures for Statistical Computing in Python. In Stéfan van der Walt and Jarrod Millman, editors, *Proceedings of the 9th Python in Science Conference*, pages 56 – 61, 2010. doi: 10.25080/Majora-92bf1922-00a. URL https://doi.org/10.25080/Majora-92bf1922-00a.

Arkaitz Zubiaga, Maria Liakata, Rob Procter, Geraldine Wong Sak Hoi, and Peter Tolmie. Analysing how people orient to and spread rumours in social media by looking at conversational threads. *PLOS ONE*, 11(3):e0150989, mar 2016. doi: 10.1371/journal.pone.0150989. URL https://doi.org/10.1371%2Fjournal.pone.0150989.

Arkaitz Zubiaga, Maria Liakata, and Rob Procter. Exploiting context for rumour detection in social media. In Giovanni Luca Ciampaglia, Afra Mashhadi, and

Taha Yasseri, editors, *Social Informatics*, pages 109–123, Cham, 2017. Springer International Publishing. URL https://doi.org/10.1007/978-3-319-67217-5_8.

Zusammenfassung

Motivation. Informationen verbreiten sich rasend schnell über soziale Medien, dabei stellt insbesondere die Verbreitung von Falschinformationen eine ernsthafte Bedrohung für die moderne, liberal-demokratische Gesellschaft dar. Daher ist es wichtig, zu erforschen, wie sich Falschinformation verbreitet. Wir stellen einen ersten Ansatz für dieses noch unerforschte Gebiet vor.

Forschungsfrage. Erhalten Beiträge, die falsche Behauptungen enthalten, mehr Nutzerinteraktion in den sozialen Medien als Beiträge mit wahren Behauptungen?

Methodik. Wir untersuchen eine potentielle Korrelation zwischen dem Wahrheitsgehalt und der Anzahl der erhaltenen Nutzerinteraktionen von Beiträgen in sozialen Medien. Dazu erkennen wir Behauptungen mithilfe eines binären Textklassifikators und schätzen deren Wahrheitsgehalt anhand ihrer Ähnlichkeit zu Behauptungen mit bekanntem Wahrheitsgehalt.

Ergebnis. Wir berichten von keiner feststellbaren, deutlichen Korrelation zwischen dem Wahrheitsgehalt und der Anzahl der erhaltenen Nutzerinteraktionen von Beiträgen. Dies gilt für alle betrachteten Klassifikationsmodelle und untersuchten Metriken für Interaktionen.

Schlussfolgerung. Wir können mit unseren verwendeten Mitteln keinen Einfluss des Wahrheitsgehalts von Beiträgen auf deren erhaltene Nutzerinteraktionen nachweisen. Daher schlussfolgern wir die Notwendigkeit weiterer Nachforschungen in diesem Forschungsbereich.

Erklärung

Ich versichere, diese Arbeit selbstständig verfasst zu haben. Ich habe keine anderen als die angegebenen Quellen benutzt und alle wörtlich oder sinngemäß aus anderen Werken übernommene Aussagen als solche gekennzeichnet. Weder diese Arbeit noch wesentliche Teile daraus waren bisher Gegenstand eines anderen Prüfungsverfahrens. Ich habe diese Arbeit bisher weder teilweise noch vollständig veröffentlicht. Das elektronische Exemplar stimmt mit allen eingereichten Druck-Exemplaren überein.

Datum und Unterschrift:

Declaration

I hereby declare that the work presented in this thesis is entirely my own. I did not use any other sources and references that the listed ones. I have marked all direct or indirect statements from other sources contained therein as quotations. Neither this work nor significant parts of it were part of another examination procedure. I have not published this work in whole or in part before. The electronic copy is consistent with all submitted hard copies.

Date and Signature: