

# Behavioral Forensics in Social Networks: Identifying Misinformation, Disinformation and Refutation Spreaders Using Machine Learning

Euna Mehnaz Khan<sup>\*,1</sup>, Ayush Ram<sup>2</sup>, Bhavtosh Rath<sup>3</sup>, Emily Vraga<sup>4</sup>, and Jaideep Srivastava<sup>5</sup>

## ABSTRACT

With the ever-increasing spread of misinformation on online social networks, it has become very important to identify the spreaders of misinformation (unintentional), disinformation (intentional), and misinformation refutation. It can help in educating the first, stopping the second, and soliciting the help of the third category, respectively, in the overall effort to counter misinformation spread. Existing research to identify spreaders is limited to binary classification (true vs false information spreaders). However, people's intention (whether naive or malicious) behind sharing misinformation can only be understood after observing their behavior after exposure to both the misinformation and its refutation which the existing literature lacks to consider. In this paper, we propose a labeling mechanism to label people as one of the five defined categories based on the behavioral actions they exhibit when exposed to misinformation and its refutation. However, everyone does not show behavioral actions but is part of a network. Therefore, we use their network features, extracted through deep learning-based graph embedding models, to train a machine learning model for the prediction of the classes. We name our approach *behavioral forensics* since it is an evidence-based investigation of suspicious behavior which is spreading misinformation and disinformation in our case. After evaluating our proposed model on a real-world Twitter dataset, we achieved 77.45% precision and 75.80% recall in detecting the *malicious* actors, who shared the misinformation even after receiving its refutation. Such behavior shows intention, and hence these actors can rightfully be called agents of disinformation spread.

## KEYWORDS

fake news mitigation, social network analysis, user behavior analysis, malicious spreader detection, machine learning

## 1 INTRODUCTION

In recent times, the ease of access to Online Social Networks and the extensive reliance on it for news has increased the dissemination of misinformation. The spread of misinformation has severe impacts on our lives which has also been witnessed during the COVID-19 pandemic [3]. Hence, it is important to detect misinformation along with its spreaders. It is worth to be noted that misinformation and disinformation are very related yet different terms. Misinformation is incorrect or misleading information [18] whereas disinformation is spread deliberately with the intention to deceive [17].

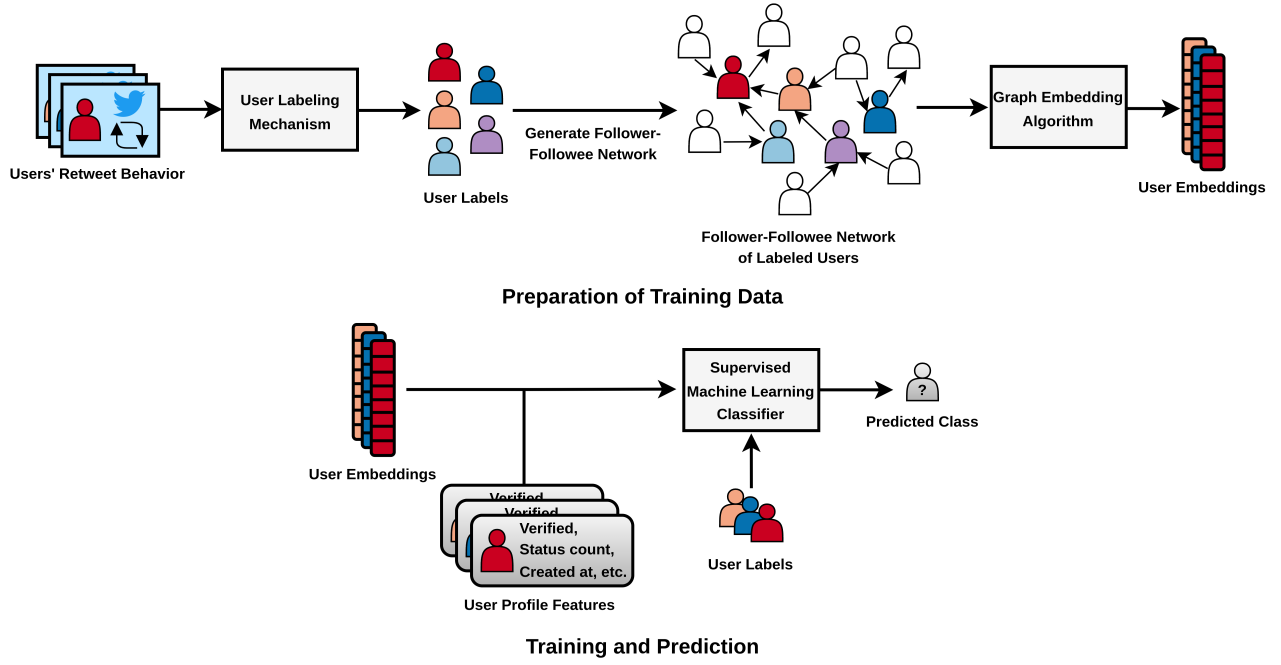
Fact-checking websites often debunk misinformation and publish its refutation. As a result, both the misinformation and its refutation can co-exist in the network and people can be exposed to them in different orders. At some point in time, they might get exposed to the misinformation, and retweet it. Later, they may get exposed to its refutation and retweet it. Since they have corrected their mistake, it can be inferred that they had spread the misinformation unintentionally. However, social media usually bans or flags accounts that they deem to be objectionable [14] without investigating the intention of those people sharing the misinformation. This results in many unfair bans of accounts that were simply deceived by the misinformation. On the other hand, some people may not correct their mistakes, or despite receiving the refutation they may choose to retweet the misinformation. These kinds of activities reveal their bad intention and hence these people can be considered as *malicious*. Again, some people might be smart enough to identify misinformation and choose to share refutations instead which indicates good intentions. Identifying these different groups of people will enable us to mitigate misinformation spread efficiently. For instance, Twitter may flag or ban the *malicious* people besides incentivizing the good people to spread the refutation. The followers of *malicious* people can also be treated with the refutation as a preventive measure [20]. Inoculating people against common misinformation and misleading tactics has shown promise, so targeting vulnerable groups more precisely offers great advantages [9]. In this paper, we build a model to identify these different groups of spreaders using their behavioral and network properties. The novelty of our approach is that it considers the possible series of actions (retweet or not retweet) a person might take when exposed to both misinformation and its refutation.

Most research on misinformation and refutation spreader detection utilizes users' profile features (i.e., gender, location, follower-follower count, linguistic features, sentiment extracted from content posted by them) [4, 32]. Some research [22, 28] has considered network features extracted through graph embedding algorithms to identify the spreaders. However, these works considered only two classes of spreaders (misinformation and refutation spreaders). In these works, people are labeled as misinformation spreaders if they retweet that information just once, despite the fact that given the refutation, they might have acted differently. On the other hand, if a person shares the refutation, they are considered true information spreaders whereas they might have shared the misinformation if received earlier, in which case the same model identifies them as misinformation spreaders. Thus, only part of the whole scenario has been considered in existing literature.

In this paper, we build a model to classify different kinds of spreaders. First, we label people into one of our five defined classes.

This work was first submitted to CIKM 2022.

<sup>\*</sup>Euna Mehnaz Khan is the corresponding author of this paper. All the authors are with the University of Minnesota, Minneapolis, MN. Email: {<sup>1</sup>khan0586, <sup>2</sup>ram00009, <sup>3</sup>rathx082, <sup>4</sup>ekvrage, <sup>5</sup>srivasta}@um.edu.



**Figure 1: Overview of the proposed *behavioral forensics* model which generates labels from people’s retweet behavior and utilizes both network and profile features to train machine learning classifiers.**

Our labeling mechanism is novel because we label people only after they have been exposed to both misinformation and its refutation aiming to understand their intention and consider multiple levels of good and bad behavior. However, it is not possible to classify everyone solely based on behavior because everyone does not have behavioral data. But everyone is part of the social network. Hence, features from their networks are used to predict the categories of people without behavioral histories. For that purpose, from the follower-followee network of these labeled people, we extract their network features using graph embedding models, like Large-scale Information Network Embedding (LINE) [27] and PyTorch-BigGraph [8]. Then, we use these learned network features along with the profile features of the labeled people to train a machine learning classification model and predict the labels. In this way, for users without past behavioral histories, we can predict the labels from their network and profile features. We have tested our model on a Twitter dataset, and have achieved 77.45% precision and 75.80% recall in detecting the *malicious* class (the extreme bad) where the accuracy of the model is 73.64% with a weighted F1 score of 72.22%, thus significantly outperforming the baseline models. The contributions of this paper are as follows:

- We observe people’s sequence of behaviors when they are exposed to false information and its refutation, and label them only after they have been exposed to both.
- We propose a nuanced labeling mechanism that uses behavioral data to label people into five categories. This is a granular level classification of misinformation, disinformation, and refutation spreaders which aims to identify the intent of people based on their history of actions.

- We leverage the network embedding features of users to predict their classes so that people without any behavioral property can also be categorized.

An overview of the proposed approach, which we name *behavioral forensics*, is demonstrated in Fig. 1.

## 2 RELATED WORK

Existing research in the area of misinformation detection mostly includes analyzing the content (i.e., linguistic, stylistic features, etc.) [7, 30], using deep learning-based models [12, 13], propagation structure of misinformation in the network [15], users’ profile features (account age, number of tweets and followers) [4], their history of sharing false and true information [26], etc. Research on identifying spreaders of misinformation and refutations is mostly centered around utilizing users’ profile features [32], linguistic and personality features from timeline tweets [6, 25], and sentiment and topics from past tweets [10]. Other works include identifying the misinformation source [35], finding minimal-size clusters of fake news spreaders [1] or refutation spreaders [19] who can influence the most people. Recently people’s interactions and network structures are also being considered while identifying the misinformation spreaders [22–24, 28]. However, none of the past works has considered the fact that a person can exhibit a series of behaviors when exposed to misinformation and its refutation. People’s perceptions of truth change as they get more exposed to the facts. They may retract their previous actions (retweeting the misinformation) by doing something opposite (retweeting the refutation) to account for their mistake, which implies good behavior. On the other hand, labeling them as *malicious* or bad people based on the

fact that they chose not to share a refutation and instead shared the misinformation gives more evidence than just relying on the fact that they shared the misinformation. In this paper, we have identified the multiple states that one can go through when exposed to both misinformation and its refutation, and classify them using their network properties.

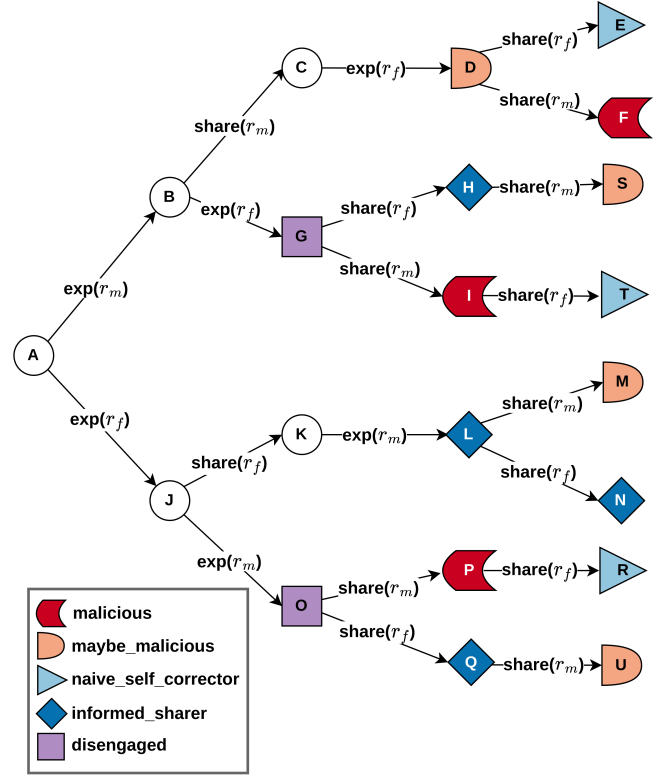
### 3 BEHAVIORAL FORENSICS IN SOCIAL NETWORKS

Every organization has some policy for identifying suspects based on their behavior. For example, many finance organizations will block a user after three failed password attempts, on the suspicion that the attempts are from a potential hacker. A similar approach is taken by organizations in identifying who might be originators and/or spreaders of misinformation, based on their behavior. Such approaches tend to be organization specific, and can be expressed as a set of labeling rules. *Our proposed research is not tied to any specific policy for identifying suspected perpetrators, but rather can take any policy and work with it. To illustrate our work, we pick a specific policy, which we believe is reasonable, but does not represent any specific organization. It is merely an illustrative example.* The overall approach is shown in Fig. 1, which requires a labeling policy, but not any specific one.

While labeling people, we consider only those who are exposed to at least one pair of misinformation and its refutation. Next, we observe their sequence of actions upon exposures to those messages and label them into one of the five categories using our proposed definitions described below. The second part (generating the embedding features using the follower-followee network of the labeled people and training machine learning models) is independent of the labeling mechanism (policies used to define spreaders) used. The categories can be predicted using embedding features which is our main contribution. However, the key novelty introduced regarding the labeling here is that people should be labeled by observing their actions only when they have been exposed to both misinformation and its refutation. The underlying policy to define the categories can differ from ours but has to be rational.

**Example policy:** A possible series of behavioral actions is depicted using a state diagram in Fig. 2. Here, misinformation and its refutation are denoted by  $r_m$  and  $r_f$ , respectively. Sharing or retweeting  $r_m$  is represented by  $\text{share}(r_m)$  whereas exposure to  $r_m$  is represented by  $\text{exp}(r_m)$ . Same applies for  $r_f$ . Self-loops are not shown in the diagram. For instance, if a person is at state G and they share an  $r_m$ , they go to state I. Now, they stay at state I if they repeatedly keep sharing the  $r_m$ . The self-loops are removed from the diagram for clarity of the figure. The definitions used for our proposed five classes are described below:

- (1) **malicious:** These are the people who spread misinformation knowingly. After being exposed to both  $r_m$  and  $r_f$ , they decide to spread  $r_m$ , not  $r_f$ . Since refutations are published by fact-checking websites, they are clear to identify as true and usually are not confused with  $r_m$ . Therefore, when a person shares the  $r_m$  even after getting the  $r_f$ , they can be considered to have malicious intent, hence categorized as *malicious*.



**Figure 2: State diagram for the proposed labeling mechanism demonstrating different sequences of behavior upon exposure,  $\text{exp}()$ , to a piece of misinformation ( $r_m$ ) and its refutation ( $r_f$ ).**

In Fig. 2, a person in state G or O is exposed to both  $r_m$  and  $r_f$  ( $A \rightarrow B \rightarrow G$ ,  $A \rightarrow J \rightarrow O$ ). Now, if they choose to share the  $r_m$ , they go to state I ( $G \rightarrow I$ ) and P ( $O \rightarrow P$ ), respectively. If they do not take further action, or repeatedly keep sharing the  $r_m$ , we label them as *malicious*. In another scenario, a person shares  $r_m$  after getting it ( $A \rightarrow B \rightarrow C$ ) and then they get the  $r_f$  ( $C \rightarrow D$ ). Now, if they still choose to share the  $r_m$  again ( $D \rightarrow F$ ), we label them as *malicious*. The *malicious* class of people is an extremely bad category in terms of misinformation spread and should be banned or flagged.

- (2) **maybe\_malicious:** This class refers to the following groups of people:
  - people who have shared the misinformation ( $A \rightarrow B \rightarrow C$ ), then have received its refutation ( $C \rightarrow D$ ) but did not share the refutation (they stay at state D). These people did not correct their mistake which may indicate malicious intent or it is possible that they got the  $r_f$  so late that the topic has become outdated. They are not necessarily the bad group of people. We can keep an eye on them and monitor their activities.
  - people who shared the  $r_m$  after sharing its  $r_f$  (perhaps they cannot distinguish between true and false information

although  $r_f$ s are usually clear to identify as true). The sequences  $A \rightarrow B \rightarrow G \rightarrow H \rightarrow S$ ,  $A \rightarrow J \rightarrow K \rightarrow L \rightarrow M$ ,  $A \rightarrow J \rightarrow O \rightarrow Q \rightarrow U$  demonstrate this kind of behavior.

The intent of these two subgroups of people is not clear from their behavior. But since they shared  $r_m$  as the latest action, we grouped them as a separate category called *maybe\_malicious*. These people are not as bad as the *malicious* class of people, however, they still contribute to misinformation spread. Less intense measures like providing refutations to their followers can be taken to account for the harm caused by them.

- (3) **naive\_self\_corrector**: These people got deceived by the  $r_m$  and shared it (naive behavior) but later corrected their mistake by sharing the  $r_f$  (self-correcting behavior). The sequences  $A \rightarrow B \rightarrow C \rightarrow D \rightarrow E$ ,  $A \rightarrow B \rightarrow G \rightarrow I \rightarrow T$ ,  $A \rightarrow J \rightarrow O \rightarrow P \rightarrow R$  fall into this category. These people can be provided  $r_f$  early to prevent them from naively believing and spreading  $r_m$  and be utilized to spread the true information.
- (4) **informed\_sharer**: This category includes two types of people:
  - People who shared only  $r_f$  after exposure to both  $r_m$  and  $r_f$  ( $A \rightarrow B \rightarrow G \rightarrow H$ ,  $A \rightarrow J \rightarrow O \rightarrow Q$ ).
  - People who shared  $r_f$  ( $A \rightarrow J \rightarrow K$ ), then after receiving  $r_m$  ( $K \rightarrow L$ ), did not share it (stay at state  $L$ ) or they shared the refutation again ( $L \rightarrow N$ ). Both the states  $L$  and  $N$  fall under this category.

This group of people are smart enough to distinguish between true and false information and are willing to fight misinformation spread by sharing the refutation. Therefore, they should be provided with refutations on the onset of misinformation dissemination to contain its spread.

- (5) **disengaged**: People who received both  $r_m$  and  $r_f$  but shared nothing are defined as *disengaged* people. This group of people does not incline to share any true or false information. People in state  $G$  (sequence  $A \rightarrow B \rightarrow G$ ) and state  $O$  (sequence  $A \rightarrow J \rightarrow O$ ) are *disengaged* people.

It should be noted that when a person is in state  $G$  and takes no further action, they are identified as *disengaged*. But if they share something at this point, then they make a transition to state  $H$  or  $I$  depending on what they share. For instance, if they share  $r_f$ , they go to state  $H$ . Now, if they stop here, then they are defined as *informed\_sharer*. However, if they share  $r_m$  here, they go to state  $S$  which indicates *maybe\_malicious* class.

## 4 PROPOSED METHOD

In this paper, our goal is to classify people into one of the five categories defined in Section 3. We follow the methodology described below to accomplish our goal:

### 4.1 Labeling Mechanism

First, we label the people who have been exposed to misinformation and its refutation. These labels will serve as gold standard labels [33] (as opposed to ground truth labels [34]) for training of our machine learning models.

In particular, we have considered multiple pairs of misinformation ( $r_m$ ) and corresponding refutations ( $r_f$ ) for this task (as discussed in Section 5). Ideally, we wanted to label someone as one class, i.e., *malicious* if they had shown that behavior multiple times across our many pairs of  $r_m$  and  $r_f$ . Although this would have been a more robust labeling of people, we observed during the experiment that a very small number of people showed behavior that falls into a class other than *disengaged*, and the number of people exhibiting that behavior multiple times was even less. To account for this problem, we labeled people according to our state diagram in Fig. 2 whenever they got exposed to at least one pair of  $r_m$ - $r_f$ . For users who got exposed to only one pair of  $r_m$  and  $r_f$ , received a single final label. On the other hand, people who got exposed to multiple pairs of  $r_m$  and  $r_f$ , got multiple labels. If a person receives multiple same labels, then we label them as the corresponding class. However, if they receive other label(s) with *disengaged* label(s), then we remove the *disengaged* label(s) and use the rest of the non *disengaged* label(s) to convert them into a single label. We do this because showing *disengaged* behavior is trivial since it means taking no action whereas the other behaviors require some sort of action which is essential to identify. Next, we use a Likert scale-like [11] representation to convert multiple labels for a user into a single label, as described below:

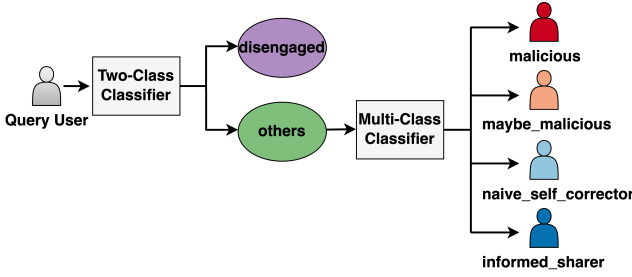
First, we represent the four different non *disengaged* classes (*malicious*, *maybe\_malicious*, *naive\_self\_corrector*, *informed\_sharer*) using the integers 1, 2, 3, and 4, respectively. Suppose, we have  $m$  labels for a person represented by the list  $l = [l_1, l_2, \dots, l_m]$ . After converting the labels to the integers, we get a list of integers  $[n_1, n_2, \dots, n_m]$ . Since we are studying behavior, and the distribution of labels for the same user was skewed, we take the median of these numbers, choosing the larger one in case of a tie to avoid false positives (considering the *malicious* as positive). However, if we had used mean (with rounding up to get an integer) instead of median, it would have changed the final label for only 8 users out of 218 multilabel users.

For example, if  $l = [\text{malicious}, \text{naive\_self\_corrector}, \text{informed\_sharer}]$ , then we get 3 as the median of  $[1, 3, 4]$ , which refers to the class *naive\_self\_corrector*.

### 4.2 Graph Embeddings

Since very few people exhibit behavioral actions as observed during our experiment, it would not be possible to label everyone using only their behavioral data. In contrast, everyone has network properties since they are part of the social network. Hence, we create a labeled dataset where labels come from the labeling mechanism stated in Section 3 and the features come from their network extracted using graph embedding models. This labeled dataset can be used to train a machine learning classifier to predict the category of a person given their network embedding features.

Graph embedding models are used to generate a low dimensional vector representation for each of the nodes in a network, preserving the network's topology and homophily of the nodes. Nodes with similar neighborhood structure should have similar vector representations. As we aim to utilize network properties of the people to distinguish between different classes, we apply



**Figure 3: Two step classification overview where we use a two-class classifier to predict the *disengaged* people and *others* and then use a multi-class classifier to further categorize the *others*.**

the existing graph embedding methods. In particular, after labeling people, we build a network using the followers and followees of the labeled users. Then, we use a graph embedding model to extract the network features of these users. Specifically, we have tried second order version of LINE [27] (as the network is directed) and PyTorch-BigGraph (PBG) [8] for this purpose. LINE algorithm captures the local and global network structure by considering the fact that similarity between two nodes is also dependent on the number of neighbors they share besides the existence of direct link between them. This is important in our problem because people from the same class may not be connected to each other but they might be connected to the same group of people, which the LINE algorithm still identifies as a similarity. For instance, people from the *malicious* class may or may not be connected to each other but their target people (whom they want to relay the misinformation to) might be the same. Again, nodes from the same class may form a cluster or community with lots of interconnections and common neighbors. LINE graph embedding technique is able to capture these aspects. On the other hand, PBG embedding system uses a graph partitioning scheme that allows it to train embeddings quickly and scale to networks with millions of nodes and trillions of edges. Its effectiveness and fast generation of embeddings for social network dataset (Twitter) is also shown in [8]. For this reason, we also try this embedding technique.

#### 4.3 Node Classification

In this step, we combine the profile features of the users with their learned embeddings to train different machine learning models (more details in Section 6.1.3). Finally, we use these models to make predictions. Due to the heavy imbalance between the *disengaged* class and other classes, we perform the classification in two steps as shown in Fig. 3. First, we classify people into the *disengaged* category and *others* category with undersampling of the *disengaged* class. Next, we classify the *others* category into the four defined classes.

### 5 DATASET

We have used the publicly available “False and refutation information network and historical behavioral data” [21] dataset for our experiment and model evaluation. It contains misinformation and

refutation related data for 10 news events (all on political topics), occurring on Twitter during 2019, identified through `altnews.in` [2], a popular fact-checking website. For each news event, the dataset includes the single original tweet (source tweet) information for a piece of misinformation and the list of people who retweeted that misinformation along with the timestamp of the retweets. It also contains the same information for its refutation tweet. As the time of retweet is missing for news events 1 and 9, we have used data for news events 2 through 8 and 10 (total 8 news events).

The dataset also includes the follower-followee network information for the retweeters of the misinformation and its refutation. Since people belonging to the *disengaged* category have retweeted neither of the true and false information, we had to collect their follower-followee network using Twitter API.

The following Twitter profile features of users in the follower-followee network is also included in the dataset:

- Follower Count
- Friend (Followee) Count
- Statuses Count (number of tweets or retweets issued by the user)
- Listed Count (number of public lists the user is a member of)
- Verified User (True/False)
- Protected Account (True/False)
- Account Creation Time

## 6 EXPERIMENTAL EVALUATION

### 6.1 Experiments

**6.1.1 Label Creation.** First, we have labeled the people who are exposed to at least one of the eight news events’ both misinformation ( $r_m$ ) and refutation tweets ( $r_f$ ) using the state diagram in Fig. 2. Now, a person is exposed to  $r_m$  or  $r_f$  when a person they follow has shared it and they see it. *Twitter’s data collection API does not let us collect the list of people who actually saw the tweet and the time when they saw it. This is why we assume that when a person shares  $r_m$  (or  $r_f$ ), all of their followers are exposed to it.* Similarly, we have used the following policy to set the exposure time<sup>1</sup>:

- We assume that when a person tweets or retweets a piece of information ( $r_m$  or  $r_f$ ), all of their followers get exposed to it. Hence, we set their exposure time equal to the (re)tweet time.
- If more than one followee of a person retweets the same message, then we set the exposure time of that person (to that message) to the earliest retweet time.
- If the user retweets the original tweet before any of the above events happens, we set the exposure time equal to the time of the user’s first retweet of that message.

Comparing the sequences of exposure time and retweet time, we have been able to label people into one of the five defined categories. After labeling, we have got 1,365,929 labeled users where 99.75% (1,362,510) of them fall into the *disengaged* category and 0.25% (3,419) of them are categorized as the other 4

<sup>1</sup>When Twitter or other social network platforms will use our tool, they can use the actual list of users who saw that  $r_m$  (or  $r_f$ ) as the set of exposed people along with the real exposure time.



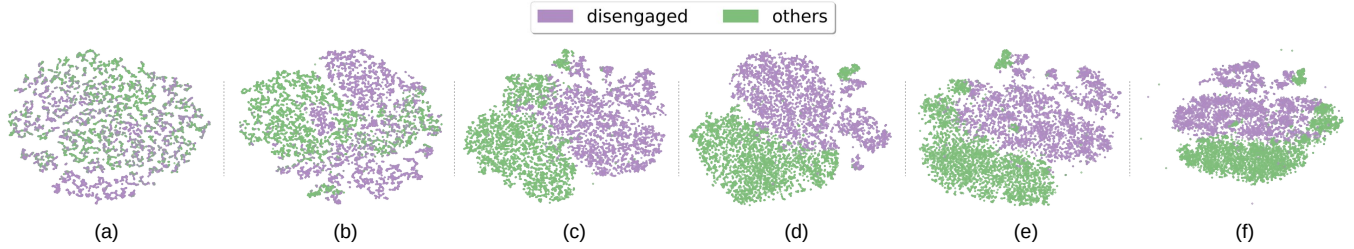


Figure 4: t-SNE plots of the learned LINE embeddings depicting two classes (*disengaged* and *others*) with varying feature dimensions: (a) 4d, (b) 8d, (c) 16d, (d) 32d, (e) 64d, and (f) 128d.

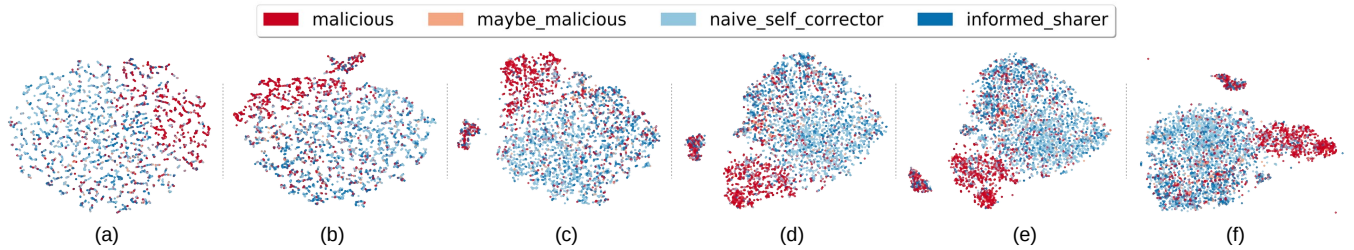


Figure 5: t-SNE plots of the learned LINE embeddings depicting the four non *disengaged* classes with varying feature dimensions: (a) 4d, (b) 8d, (c) 16d, (d) 32d, (e) 64d, and (f) 128d. (Best viewed at 300% zoom)

Table 1: Accuracy (%) of different machine learning models for the two-class classification step using 5-fold cross-validation.

Classifier	$k$ -NN	Logistic Regression	Naive Bayes	Decision Tree	Random Forest	SVM	Bagged Decision Tree
Accuracy	95.318	<b>99.251</b>	81.236	86.936	97.794	98.957	<b>99.157</b>

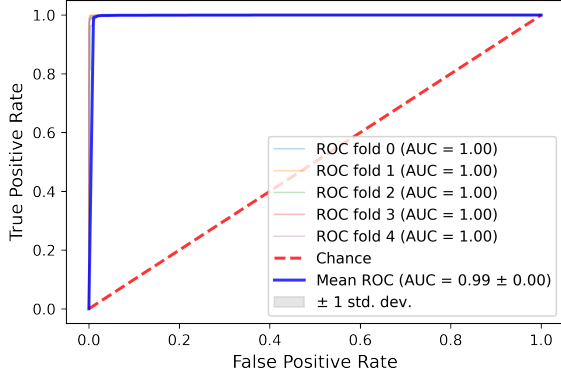
classes. The number of users in these classes are: *malicious*: 926, *maybe\_malicious*: 222, *naive\_self\_corrector*: 1,452, *informed\_sharer*: 819. We can see that most of the people (around 42%) are categorized as *naive\_self\_corrector*, which indicates that most of the people who transmit misinformation, do that mistakenly. Again, the number of people in the *malicious* and *informed\_sharer* categories implies that the number of people in the extreme good class is almost equal, if not more, to the number of people in the extreme bad class.

**6.1.2 Feature Creation.** Now, we extract the follower-followee information of these labeled people from the dataset and construct a network. Since 99.75% of the labeled people belong to the *disengaged* category, we randomly undersample the people belonging to this category and keep 4,059 of them for the analysis. After constructing the network, we have got 7.5M (7,548,934) nodes and 25M (25,037,335) edges. Then, we use graph embedding models LINE and PBG (as mentioned in Section 4.2) to extract their network features. We have generated embeddings of different dimensions (4d, 8d, 16d, 32d, 64d, and 128d).

Next, we have normalized the embedding features. We have used the embedding features directly on various two-class classifiers for the two-class classification step (*disengaged* and *others*). Since the embedding features have been able to get over 99% accuracy as discussed in Section 6.2, we have not included the profile features at this step. However, for the multi-class classification step, we

have concatenated the normalized profile features with the learned embeddings. The profile features listed in Section 5 were used in this step since authors in [26] have shown that these features exhibit a significant difference between true and false information spreader groups. The boolean (True/False) features (verified user, protected account) have been converted to integers (1/0) with the account creation time being converted to normalized account age (in days).

**6.1.3 Classification.** Since 99.75% of the labeled people belong to the *disengaged* category, and only 0.25% belong to the other four classes, we first focus on distinguishing the *disengaged* from the *others* to address the rare-class problem. Then, we further distinguish among those four classes. This is why the two-step classification is used (Fig. 3). We have tried different classifiers from different areas. For both the classification steps, we have used Logistic Regression as a linear classifier,  $k$ -Nearest Neighbors algorithm ( $k$ -NN), Support Vector Machine (SVM), Naive Bayes as non linear classifiers, Decision Tree as tree based classifier, Random Forest (with 100 trees), Bagged Decision Tree (with base estimator SVM) as ensemble classifiers. For  $k$ -NN,  $k=5$  has seemed to produce better results. One-vs-rest scheme has been used for Logistic Regression in the multi-class classification step. For the two-class classification step, the class distribution is almost balanced (4,059 *disengaged* and 3,419 *others*) after the undersampling of the *disengaged* users. However, for the multi-class classification step, the class distribution is found



**Figure 6: ROC curve for the two-class classification step using 5-fold cross-validation.**

to be imbalanced as mentioned in Section 6.1.1. To account for this problem, we have set the *class\_weights* parameter of the classifiers to ‘balanced’ when available which automatically adjusts weights inversely proportional to class frequencies in the input data. For classifiers that do not have this parameter, we have used Synthetic Minority Over-sampling Technique (SMOTE) [5] to balance the class distribution.

**6.1.4 Baselines Used for Comparison.** As the classifier built in the two-class classification step distinguishes between the disengaged people from the others, and *there is no past work which does the same*, a baseline model predicting random class (accuracy=50%) is considered for the evaluation of this step. Regarding the evaluation of the multi-class classification step for the detection of different kinds of spreaders, no existing work does such multi-class classification; all of them do binary classification to identify true and false information spreaders. Also, their definitions of false and true information spreaders to construct the labeled data are quite different from ours. Whenever a person spreads misinformation, they label them as misinformation spreader, despite the possibility of the fact that they might not have seen the refutation and got deceived by the misinformation. We have labeled our data using the labeling mechanism described in Section 3 where people are labeled only when they are exposed to a pair of misinformation and its refutation. *The results of this step are not comparable to the existing methods of detection of misinformation spreaders because of the difference in the number of classes and their definitions.* Therefore, for the multi-class classification step, two baseline models are considered: (1) Baseline 1, which predicts all samples as the majority class (*naive\_self\_corrector*), and (2) Baseline 2, which predicts random class. K-fold cross-validation with K=5 and K=10 for the two-class and multi-class classification respectively has been used for evaluation purposes.

## 6.2 Results

Both LINE and PBG embeddings show similar results in prediction. We are using LINE embedding method to report the results since it performed faster than PBG during our experiment. Fig. 4 shows

**Table 2: Accuracy (%) and weighted F1 score (%) of different machine learning models along with baseline models for the multi-class classification step using 10-fold cross-validation.**

Classifier	Accuracy	Weighted F1
Baseline 1	41.329	24.172
Baseline 2	25.108	26.988
<i>k</i> -NN	61.412	53.095
Logistic Regression	48.273	47.451
Naive Bayes	53.328	51.926
Decision Tree	41.758	41.247
Random Forest	53.546	45.285
SVM	52.239	50.783
Bagged Decision Tree	<b>73.637</b>	<b>72.215</b>

different t-distributed stochastic neighbor embedding (t-SNE) [29] plots for different dimensional LINE embeddings where separation between the *disengaged* class and *others* class is very clear as we increase the number of dimensions. For 4 and 8 dimensional LINE embeddings, (Fig. 4a and Fig. 4b), samples from both the groups are very scattered, but clusters start to appear from 16 dimension (Fig. 4c). Similarly, Fig. 5 shows the t-SNE plots for the LINE embeddings of different dimensions for the four non *disengaged* classes. As we increase the number of dimensions of the embeddings, clusters start forming, i.e., people from the *malicious* class have formed a cluster on the right and top part of Fig. 5f whereas people from the *informed\_sharer* class have formed a cluster on the bottom part of that Figure.

**6.2.1 Two-Class Classifier.** Table 1 shows the performance of the two-class classifiers using LINE embeddings with 128 dimensions. After using embeddings from different dimensions, we have observed that Logistic Regression and Bagged Decision Tree have consistently performed better (accuracy greater than 98%) than other classifiers when the number of dimensions is above 16. Logistic Regression achieves 99.251% accuracy for 128-dimensional LINE embeddings which outperforms the baseline model of accuracy = 50%. Fig. 6 depicts the Receiver Operating Characteristic (ROC) curve for 5-fold cross-validation which demonstrates the efficacy of the model.

**6.2.2 Multi-Class Classifier.** Table 2 reports the accuracy and the weighted F1 score of the multi-class classification step using 128 dimensional LINE embeddings whereas Fig. 7 depicts their precision, recall and F1 score using a bar plot. *k*-NN and bagged decision tree have consistently shown to produce better output than other classifiers where bagged decision tree has performed slightly better than *k*-NN. For example, the precision for the *malicious* class using *k*-NN is 75.812% and using bagged decision tree is 77.446%. The recall for this class is about the same (~75%) for both classifiers. The precision for the *maybe\_malicious*, *naive\_self\_corrector* and *informed\_sharer* classes are 92.078%, 64.246%, 69.073% using bagged decision tree, respectively and 54.57%, 59.893%, 60.327% using *k*-NN, respectively. Moreover, Bagged Decision Tree has achieved an accuracy score of 73.637% and a weighted F1 score of 72.215%. All of these numbers demonstrate the better prediction capability of these

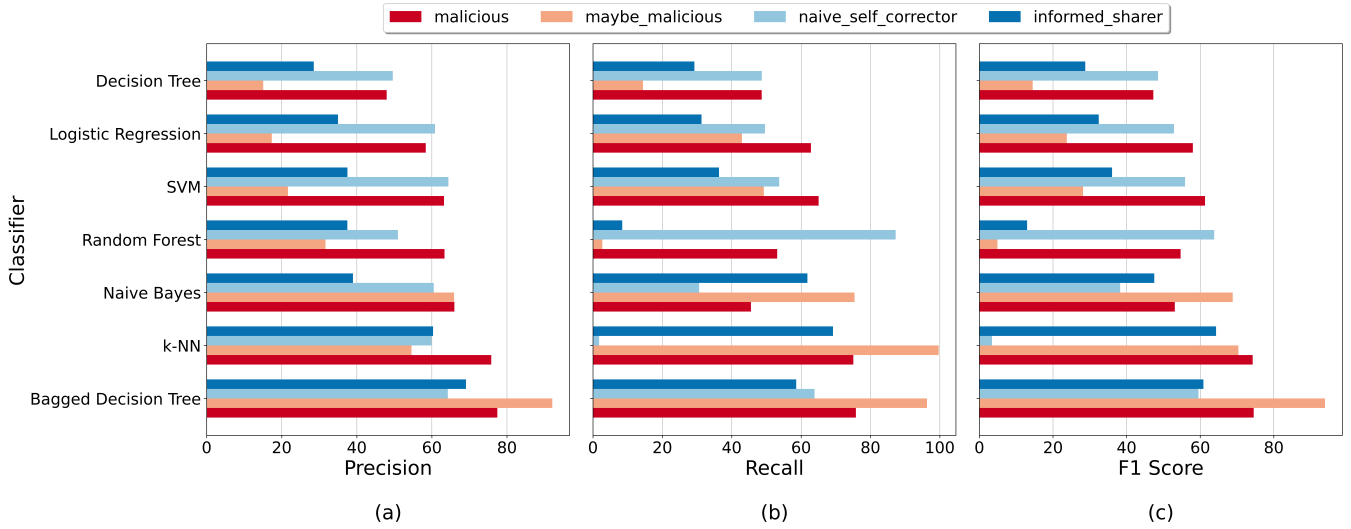


Figure 7: Horizontal bar plots showing the (a) precision, (b) recall, and (c) F1 score of four different non *disengaged* classes using different machine learning models for the multi-class classification step. All the values are in percentages.

Table 3: Precision, recall and F1 score of four different non *disengaged* classes using the baseline models for the multi-class classification step. All the values are in percentages. Note that while predicting all samples as the majority class, baseline 1 produces undefined precision and F1 score for the non-majority classes as expected, which are represented by ‘-’ in the table.

Class Categories	Baseline 1			Baseline 2		
	Precision	Recall	F1 Score	Precision	Recall	F1 Score
malicious	-	0	-	25.795	23.397	24.537
maybe_malicious	-	0	-	6.388	23.423	10.038
naive_self_corrector	41.329	100	58.486	42.430	26.229	32.418
informed_sharer	-	0	-	23.954	25.511	24.708

models than the baseline models shown in Table 3. After comparing the performance of the models using embeddings of different dimensions, we have observed improvement in the performance as we have increased the number of dimensions<sup>2</sup>.

## 7 DISCUSSION

This paper proposes a *behavioral forensics* model to identify different types of misinformation, disinformation and refutation spreaders. Our experimental results show the efficacy of our proposed model. We have used different dimensions of embeddings and observed that increasing the number of dimensions improves the performance of the model initially but this improvement slows down as we reach 64d. It is interesting to note from the t-SNE plots (Fig. 4 and Fig. 5) that people from different groups can form different clusters when only network embedding features are used. It signifies the importance of network structure in spreader detection and the classification results prove that network embedding features can be used for people’s behavioral aspect, such as spreading behavior prediction. However, the metric which should be used for model

selection and tuning depends on the mitigation techniques used to fight misinformation dissemination. For instance, if *malicious* people are decided to be banned, then we will want to put an emphasis on its precision since we do not want to ban any good account. On the other hand, if treating the followers of the *malicious* people with refutation is taken as a preventive measure, then we might want to focus on its recall, and select and tune the model based on that. If both measures are taken, then the F1 score has to be maximized. Again, if people from the *naive\_self\_corrector* and *informed\_sharer* classes are provided with refutation to facilitate the spread of true information, then their recall should be maximized. These peer corrections can reduce misinformation sharing [16] and mitigate misperceptions among the community seeing the interaction [31].

## 8 CONCLUSION & FUTURE WORK

This research proposes a novel way to define and classify different kinds of spreaders by introducing a multi-class behavior-based labeling mechanism. Exposure to both true and false information is required for labeling a person and their reactions to those messages

<sup>2</sup>The code used in this analysis is available at <https://github.com/eunakhan/behavioral-forensics>.



are observed to label them. It is also shown that people from different categories can form different clusters when network embedding features are used and those features can be used to train machine learning models to predict their labels. In this way, our proposed model is able to predict the category of a person without having behavioral history of actions. The results show the efficacy of our proposed model. Our model can be applied to any social network to fight misinformation spread. Possible future work includes updating people's labels when they exhibit a new or different behavior from the past. It will enable the model to be adaptive to people's behavior change and be more dynamic. Another possible extension to this work can be observing how the categories of people change when they are exposed to misinformation from different topics.

## ACKNOWLEDGEMENTS

We would like to acknowledge the Minnesota Supercomputing Institute (MSI) at the University of Minnesota for providing resources that contributed to the research results reported within this paper. URL: <http://www.msi.umn.edu>

## REFERENCES

- [1] Mustafa Alassad, Muhammad Nihal Hussain, and Nitin Agarwal. 2019. Finding Fake News Key Spreaders in Complex Social Networks by Using Bi-Level Decomposition Optimization Method. In *International Conference on Modelling and Simulation of Social-Behavioural Phenomena in Creative Societies*. Springer, 41–54.
- [2] Alt News. 2022. A fact-checking website. <https://www.altnews.in>.
- [3] Debanjan Banerjee and KS Meena. 2021. COVID-19 as an “infodemic” in public health: critical role of the social media. *Frontiers in Public Health* 9 (2021), 231.
- [4] Amirhosein Bodaghi and Jonice Oliveira. 2020. The characteristics of rumor spreaders on Twitter: A quantitative analysis on real data. *Computer Communications* 160 (2020), 674–687.
- [5] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. 2002. SMOTE: Synthetic Minority Over-sampling Technique. *Journal of artificial intelligence research* 16 (2002), 321–357.
- [6] Anastasia Giachanou, Bilal Ghanem, Esteban A Rissola, Paolo Rosso, Fabio Crestani, and Daniel Oberski. 2022. The impact of psycholinguistic patterns in discriminating between fake news spreaders and fact checkers. *Data & Knowledge Engineering* 138 (2022), 101960.
- [7] Benjamin Horne and Sibel Adali. 2017. This Just In: Fake News Packs a Lot in Title, Uses Simpler, Repetitive Content in Text Body, More Similar to Satire than Real News. In *Proceedings of the international AAAI conference on web and social media*, Vol. 11. 759–766.
- [8] Adam Lerer, Ledell Wu, Jiajun Shen, Timothee Lacroix, Luca Wehrstedt, Abhijit Bose, and Alex Peysakhovich. 2019. PYTORCH-BIGGRAPH: A LARGE-SCALE GRAPH EMBEDDING SYSTEM. *Proceedings of Machine Learning and Systems* 1 (2019), 120–131.
- [9] Stephan Lewandowsky and Sander Van Der Linden. 2021. Countering Misinformation and Fake News Through Inoculation and Prebunking. *European Review of Social Psychology* 32, 2 (2021), 348–384.
- [10] Zongmin Li, Qi Zhang, Yuhong Wang, and Shihang Wang. 2020. Social media rumor refuter feature analysis and crowd identification based on XGBoost and NLP. *Applied Sciences* 10, 14 (2020), 4711.
- [11] Rensis Likert. 1932. A technique for the measurement of attitudes. *Archives of psychology* (1932).
- [12] Yang Liu and Yi-Fang Wu. 2018. Early Detection of Fake News on Social Media Through Propagation Path Classification with Recurrent and Convolutional Networks. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 32.
- [13] Yi-Ju Lu and Cheng-Te Li. 2020. GCAN: Graph-aware Co-Attention Networks for Explainable Fake News Detection on Social Media. *arXiv preprint arXiv:2004.11648* (2020).
- [14] Michael Luca. 2021. Social Media Bans Are Really, Actually, Shockingly Common. <https://www.wired.com/story/opinion-social-media-bans-are-really-actually-shockingly-common>.
- [15] Jing Ma, Wei Gao, and Kam-Fai Wong. 2017. Detect rumors in microblog posts using propagation structure via kernel learning. *Association for Computational Linguistics*.
- [16] Drew B Margolin, Aniko Hannak, and Ingmar Weber. 2018. Political Fact-Checking on Twitter: When Do Corrections Have an Effect? *Political Communication* 35, 2 (2018), 196–219.
- [17] Merriam-Webster. 2022. disinformation. <https://www.merriam-webster.com/dictionary/disinformation>.
- [18] Merriam-Webster. 2022. misinformation. <https://www.merriam-webster.com/dictionary/misinformation>.
- [19] Nam P Nguyen, Guanhua Yan, My T Thai, and Stephan Eidenbenz. 2012. Containment of Misinformation Spread in Online Social Networks. In *Proceedings of the 4th Annual ACM Web Science Conference*. 213–222.
- [20] Pinar Ozturk, Huaye Li, and Yasuaki Sakamoto. 2015. Combating Rumor Spread on Social Media: The Effectiveness of Refutation and Warning. In *2015 48th Hawaii international conference on system sciences*. IEEE, 2406–2414.
- [21] Bhavtosh Rath. 2021. False and refutation information network and historical behavioral data. <https://doi.org/10.7910/DVN/GHAMOE>
- [22] Bhavtosh Rath, Aadesh Salecha, and Jaideep Srivastava. 2020. Detecting Fake News Spreaders in Social Networks using Inductive Representation Learning. In *2020 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*. 182–189. <https://doi.org/10.1109/ASONAM49781.2020.9381466>
- [23] Shakshi Sharma and Rajesh Sharma. 2021. Identifying Possible Rumor Spreaders on Twitter: A Weak Supervised Learning Approach. In *2021 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 1–8.
- [24] Anu Shrestha and Francesca Spezzano. 2022. Characterizing and predicting fake news spreaders in social networks. *International Journal of Data Science and Analytics* 13, 4 (2022), 385–398.
- [25] Anu Shrestha, Francesca Spezzano, and Abishai Joy. 2020. Detecting Fake News Spreaders in Social Networks via Linguistic and Personality Features. In *Working Notes of CLEF 2020-Conference and Labs of the Evaluation Forum*.
- [26] Kai Shu, Suhang Wang, and Huan Liu. 2018. Understanding User Profiles on Social Media for Fake News Detection. In *2018 IEEE conference on multimedia information processing and retrieval (MIPR)*. IEEE, 430–435.
- [27] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. 2015. LINE: Large-scale Information Network Embedding. In *Proceedings of the 24th international conference on world wide web*. 1067–1077.
- [28] Antonela Tommasel, Juan Manuel Rodriguez, and Filippo Menczer. 2022. Following the Trail of Fake News Spreaders in Social Media: A Deep Learning Model. In *Adjunct Proceedings of the 30th ACM Conference on User Modeling, Adaptation and Personalization*. 29–34.
- [29] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing Data using t-SNE. *Journal of machine learning research* 9, 11 (2008).
- [30] Svitlana Volkova and Jin Yea Jang. 2018. Misleading or Falsification: Inferring Deceptive Strategies and Types in Online News and Social Media. In *Companion Proceedings of the The Web Conference 2018*. 575–583.
- [31] Emily K Vraga and Leticia Bode. 2020. Correction as a Solution for Health Misinformation on Social Media. , S278–S280 pages.
- [32] Shihang Wang, Zongmin Li, Yuhong Wang, and Qi Zhang. 2019. Machine Learning Methods to Predict Social Media Disaster Rumor Refuters. *International journal of environmental research and public health* 16, 8 (2019), 1452.
- [33] Wikipedia contributors. 2022. Gold standard (test) – Wikipedia, The Free Encyclopedia. [https://en.wikipedia.org/w/index.php?title=Gold\\_standard\\_\(test\)&oldid=1110106256](https://en.wikipedia.org/w/index.php?title=Gold_standard_(test)&oldid=1110106256).
- [34] Wikipedia contributors. 2022. Ground truth – Wikipedia, The Free Encyclopedia. [https://en.wikipedia.org/w/index.php?title=Ground\\_truth&oldid=1092727524](https://en.wikipedia.org/w/index.php?title=Ground_truth&oldid=1092727524).
- [35] Kai Zhu and Lei Ying. 2014. Information Source Detection in the SIR Model: A Sample Path Based Approach. *IEEE/ACM Transactions on Networking* 24, 1 (2014), 408–421.