BullyNet: Unmasking Cyberbullies on Social Networks

Aparna Sankaran Srinath, Hannah Johnson, Gaby G. Dagher, and Min Long

***Abstract*— One of the most harmful consequences of social media is the rise of cyberbullying, which tends to be more sinister than traditional bullying, given that online records typically live on the Internet for quite a long time and are hard to control. In this article, we present a three-phase algorithm, called BullyNet, for detecting cyberbullies on Twitter social network. We exploit bullying tendencies by proposing a robust method for constructing a cyberbullying signed network (SN). We analyze tweets to determine their relation to cyberbullying while considering the context in which the tweets exist in order to optimize their bullying score. We also propose a centrality measure to detect cyberbullies from a cyberbullying SN and show that it outperforms other existing measures. We experiment on a data set of 5.6 million tweets, and our results show that the proposed approach can detect cyberbullies with high accuracy while being scalable with respect to the number of tweets.**

***Index Terms*— Cyberbullying, signed networks (SNs), social media mining.**

I. Introduction

THE Internet has created never before seen opportunities for human interaction and socialization. In the past decade, social media, in particular, has had a popularity explosion. From MySpace to Facebook, Twitter, Flickr, and Instagram, people are connecting and interacting in a way that was previously impossible. The widespread usage of social media across people from all ages created a vast amount of data for several research topics, including recommender systems [1], link predictions [2], visualization, and analysis of social networks [3].

While the growth of social media has created an excellent platform for communications and information sharing, it has also created a new platform for malicious activities, such as spamming [4], trolling [5], and cyberbullying [6]. According to the Cyberbullying Research Center (CRC) [7], cyberbullying occurs when someone uses the technology to send messages to harass, mistreat, or threaten a person or a group. Unlike traditional bullying where aggression is a short and temporary face-to-face occurrence, cyberbullying contains hurtful messages that are present online for a long time. These messages can be accessed worldwide and are often irrevocable. Laws about cyberbullying and how it is handled differ from one place to another. For example, in the United States, the majority of the states incorporate cyberbullying into their bullying laws, and cyberbullying is considered a criminal offense in most of them [8]. Popular social media platforms, such as Facebook and Twitter, are very vulnerable to cyberbullying due to the popularity of these social media sites and the anonymity that the Internet offers to the perpetrators. Although strict laws exist to punish cyberbullying, there are very less tools available to effectively combat cyberbullying. Social media platforms provide users with the option to self-report abusive behavior and content in addition to providing tools to deal with bullying. For example, Twitter has features that include locking accounts for a brief period of time or banning the accounts when the behavior becomes unacceptable. The body of work produced by the research community with regard to cyberbullying in social networks also needs to be expanded to get better insights and help develop effective tools and techniques to tackle the issue.

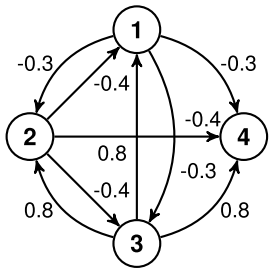


Fig. 1. Example of an SN.

To identify cyberbullies in social media, we first need to understand how social media can be modeled. The common way of modeling relationship in social psychology [9] is to represent it as a signed graph with positive edge that corresponds to the good intent and negative edge that corresponds to malicious intent between people. Using the signed graph, we model the Twitter social network as an SN to represent users' behavior [10] where nodes correspond to users and directed edges correspond to communications and/or relations between the users with assigned weight in the range $[-1, 1]$, as shown in Fig. 1.

Definition 1: A signed social network (SSN) is a directed, weighted graph $G = (V, E, W)$, where V is the set of users and $E \subseteq V \times V$ is the set of edges with an edge weight $w \in W$ in the range of $[-1,1]$.

Mining social media networks to determine cyberbullies imposes several challenges and concerns. First, it is typically hard to accurately interpret user's intentions and meanings

Manuscript received June 30, 2020; revised October 20, 2020 and December 15, 2020; accepted December 20, 2020. Date of publication January 18, 2021; date of current version April 1, 2021. *(Corresponding author: Aparna Sankaran Srinath.)*

The authors are with the Department of Computer Science, Boise State University, Boise, ID 83725 USA (e-mail: aparnasankaran@u.boisestate.edu).

Digital Object Identifier 10.1109/TCSS.2021.3049232

in social media based merely on their messages (e.g., posts, tweets, and comments), which are typically short, use slang languages, or may include multimedia contents such as pictures and videos. For example, Twitter limits its users' messages to 140 characters, which could be a mix of text, slangs, emojis, and gifs. As a result, it is hard to determine the opinion expressed in a message correctly. For this, we utilize a sentiment analysis (SA) [11], [12] to determine whether the user's attitude toward other users are positive, negative, or neutral. Second, bullying could be hard to detect whether the bully chooses to disguise it through techniques, such as sarcasm or passive aggression. In this situation, a single text (message) cannot determine the user's intention. Therefore, we collect the entire conversation between two or more users to identify the context in which the user attitude exists. Third, the large size and dynamic and complex structure of social media networks make it challenging to identify cyberbullies. For example, on Twitter, hundreds of millions of tweets are sent every day on the social network platform. In this case, we construct the social network as a graph and assign value based on the maliciousness of the user. Because the network analysis reduces the complex relationship between the users to the simple existence of nodes and edges [10]. There are several works in the literature concerning detecting malicious users from unsigned networks with positive edge weights, including community detection [13], node classification [14], and link prediction [2]. On the other hand, methods that analyze SSNs are scarce [15].

In this article, we study the problem of cyberbullying in social media in an attempt to answer the following research question: Can tweet contexts (conversations) help improve the detection of cyberbullying in Twitter? Our intuition is that each tweet should be evaluated not only based on its contents but also based on the context in which it exists. We call such a context a conversation, which is a set of tweets between two or more people exchanging information about a certain subject. Thus, our solution consists of three parts. First, for each conversation, a conversation graph is generated based on the sentiment and bullying words in the tweets. Second, we compute the bullying score for each pair of users in a conversation graph and then combine all graphs to create an SSN called bullying SN ($\mathcal{B}$). The inclusion of negative links can bring out information that would otherwise be missed with only positive links [16]. Finally, we propose a centrality measure called attitude and merit ($A\&M$) to detect bullying users from the SN $\mathcal{B}$.

Our main contributions are organized as follows.

1) Collected, preprocessed, and labeled the Twitter data set.
2) Proposed a novel efficient algorithm for detecting cyberbullies on Twitter.
    a) Built conversation.
    b) Constructed bullying SN.
    c) Proposed A&M centrality.
3) Experimented on 5.6 million tweets collected over six months. The results show that our approach can detect cyberbullies with high accuracy while being scalable with respect to the number of tweets.

II. Related Works

In this section, we review the literature on areas related to cyberbullying detection and SSNs.

A. Cyberbullying Detection

There is not a lot of works in the literature that utilizes SNs to detect cyberbullies. The papers [6] and [17] are aimed at detecting trolls in an SN. Wu *et al.* [17] proposed a method for ranking nodes to identify trolls without using a PageRank algorithm. Kumar *et al.* [6] proposed an iterative algorithm involving new decluttering operations and various centrality measures to detect trolls. Unlike the proposed method in this article, the authors begin their process with an already created SN.

A significant amount of work has been done over the past decade in the area of cyberbullying detection in general. There have been two broad methods in identifying bullies—one aims to detect bullying messages [18]–[21], whereas the other approach is to detect the cyberbullies responsible for the messages [22]–[25].

The first method of determining bullying messages was done using a combination of text-based analytics and a mix of text and user features. Zhao *et al.* [18] proposed a text-based embeddings enhanced bag-of-words (EBoW) model that utilizes a concatenation of bullying features, bag-of-words, and latent semantic features to obtain a final representation, which is then passed through a classifier to identify cyberbullies. Xu *et al.* [21] used textual information to identify emotions in bullying traces, as opposed to determining whether or not a message was bullying. Singh *et al.* [19] proposed a probabilistic sociotextual information fusion for cyberbullying detection. This fusion uses social network features derived from a 1.5 ego network and textual features, such as density of bad words and part-of-speech-tags. Hosseinmardi *et al.* [20] used images and text to detect cyberbullying incidents. The text and image features were gathered from media sessions containing images and the corresponding comments, which was then fed into various classifiers. Cheng *et al.* [25] proposed a novel method in identifying cyberbullies within a multimodal context. To understand cyberbullying, Kao *et al.* [26] proposed a framework by studying social role detection. By using words and comments, temporal characteristics, and social information of a session as well as peer influence Cheng *et al.* [27], [28] proposed frameworks for detecting cyberbullies.

The second method was aimed at identifying the person behind the cyberbullying incidents. Squicciarini *et al.* [22] used MySpace data to create a graph, which integrated user, textual, and network features. This graph was used to detect cyberbullies and predict the spreading of bullying behavior through node classification. Galán-García *et al.* [23] used supervised machine learning to detect the real users behind troll profiles on Twitter and demonstrated the technique in a real case of cyberbullying. In a recent paper on aggression and bullying in Twitter, Chatzakou *et al.* [24] found cyberbullies and aggressors using user, text, and network-based features.

TABLE I
COMPARATIVE EVALUATION OF THE MAIN FEATURES IN RELATED APPROACHES INCLUDING OUR PROPOSED APPROACH

Approach	Detect			Attributes based on				Signed Network		Dataset			
	Cyberbullying		Other	Content	Context	User	Network	Yes	No	Twitter	YouTube	Slashdot	Instagram
	Message	User											
Zhao *et al.* [18]	●			●					●	●			
Xu *et al.* [21]	●			●					●	●			
Hosseinmardi *et al.*, [20]	●			●					●				●
Dadvar *et al.*, [35]	●			●					●		●		
Dinakar *et al.*, [36]	●			●		●			●		●		
Squicciarini *et al.* [22]		●		●	●	●	●		●	●			
Chen *et al.* [37]		●		●					●		●		
Galán-García *et al.* [23]		●		●		●			●	●			
Chatzakou *et al.* [24]		●		●		●	●		●	●			
Mishra & Bhattacharya [34]			●				●	●				●	
Kumar *et. al* [6]			●				●	●				●	
Wu *et. al* [17]			●				●	●				●	
Ortega *et al.* [38]			●				●	●				●	
Our proposed protocol		●		●	●		●	●		●			

From the above methods, we determined that these approaches focus on how offensive the content of the message is based on that they identify cyberbullies but does not consider why the message was offensive, i.e., the above papers do not analyze the context of the entire conversation just the content of the message. Our approach utilizes the bag-of-words with the text to identify curse words and use SA to determine the emotions or attitude of the sender, and finally, we analyze the entire context in which the sender and receiver communicate. These overlooked factors could significantly or completely change the results of cyberbullying detection.

B. SSNs

This section reviews the previous work done on SNs [6], [10], [15], [17], [29]. The idea of SNs is not new, but its application and analysis of them were only developed in recent years. We extended its application to establish node classification in our model. Previously, in 2010, Leskovec *et al.* [10] reviewed the balance and status theory and their relation to social media and proposed a modified status theory that better reflects patterns found in SNs in social media. Tang *et al.* [15], [29] have done a broad survey of SNs in social media and proposed a new framework for node classification in SSNs. The authors incorporated negative links in the SN and proposed an approach to mathematically model both independent and dependent information from the links.

Over the last few years, a number of methods have been designed for SN analysis with both positive and negative links [30]–[33]. Most of these methods are based on simple modifications of the PageRank or eigenvector centrality that accounts for negative weights on the links. However, some of these measures do not consider how the incoming edges of a node depend on the outgoing edges from the same node and vice versa, i.e., interactions between incoming and outgoing links in SNs. Mishra and Bhattacharya [34] employed this scenario and proposed bias and deserve (BAD) measures. The deserve of a node depends on the opinions of other nodes, whereas the trustworthiness of a node depends on how a node gives a correct opinion about other nodes. From the experiment in Section VI-D, we can find that the BAD measures are not effective for identifying bullies in the network.

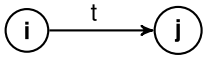


Fig. 2. Example of a tweet.

TABLE II
TWEET FEATURES

$\mathcal{SID}$	$\mathcal{DID}$	$\mathcal{UID}$	$\mathcal{RID}$	$\mathcal{MID}$	Text
101	3001	UserI	UserJ	UserX,UserY	@UserX @UserY Lets meet at the central park

Table I provides a comparative evaluation of main features in related approaches including our proposed approach.

III. PROBLEM FORMULATION

In this section, the Twitter social network is represented as a directed, weighted graph $G = (U, E)$ with U being the set of users (represented as nodes) and E being the set of tweets T sent between the users(represented as edges). Each user $u \in U$ has a set of features, including an ID, the number of followers, the number of friends, and the number of the tweets that they sent.

Each tweet $t \in T$ is associated with certain features: source ID ($\mathcal{SID}$), destination ID ($\mathcal{DID}$), the date of creation, a user ID ($\mathcal{UID}$), a reply ID ($\mathcal{RID}$), and mentions ($\mathcal{MID}$). If the tweet includes mentions (i.e., if a given @username is included in a tweet anywhere else but at the very start), then Twitter interprets this as a mention and the user gets a notification that someone has mentioned them.

As shown in Fig. 2, the notation e_{ij} represents a tweet t directed edge from node (user) i to node (user) j. The existence of an edge e_{ij} denotes an interaction from node i to node j which is t. Each tweet has a set of features, as shown in Table II. ($\mathcal{SID}$) is assigned when a new tweet is created; in this case, it is 101. ($\mathcal{DID}$) is an ID to which this current tweet is in response to where the destination ID is 3001. ($\mathcal{UID}$), ($\mathcal{RID}$), and ($\mathcal{MID}$) correspond to IDs of that particular users/nodes. Finally, the text is the content of the tweet sent from node i to node j.

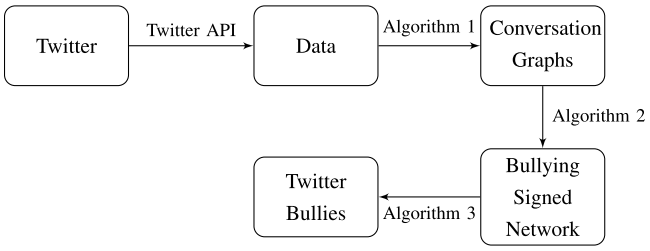


Fig. 3. Protocol flowchart of BullyNet.

From the above Twitter data, we extract conversations and build a directed weighted graph for each conversation $C = \{c_1, c_2, \ldots c_{|C|}\}$. In our model, each c_i is a set of two or more tweets between two or more users.

Definition 2: A conversation c is a set of time-ordered tweets $c = \{t_1, t_2, \ldots t_{|c|}\}$ such that the following holds.

1) The first tweet t_1 is the initiator tweet that starts the conversation and can be one of the two following types.
   a) $\mathcal{DID}(t_1)$ = NULL, and either $\mathcal{MID}(t_1)$ or $\mathcal{RID}(t_1)$ is not null.
   b) $\mathcal{DID}(t_1) \neq$ NULL, and $\forall t \subseteq T : \mathcal{SID}(t) \neq \mathcal{DID}(t_1)$.
2) All tweets in c satisfy the following: $\mathcal{SID}(t_i) = \mathcal{DID}(t_{i+1}) : 1 \leq i \leq |c| - 1$.

Our model will analyze the nodes and conversions and will output a list as results for detecting cyberbullies on the Twitter social network

$$L = \{(u_1, s_1), (u_2, s_2), \ldots, (u_{|L|}, s_{|L|})\}$$

where u_i is a user (node) and s_i is a confidence value for the likelihood of user u_i being a bully.

IV. Our Solution: BullyNet Algorithm

In this section, we first present an overview of the proposed three-phase bully finding algorithm (BFA) and elaborate the steps in each phase.

The objective of our solution is to identify the bullies from raw Twitter data based on the context as well as the contents in which the tweets exist. Given a set of tweets T containing Twitter features such as user ID, reply ID, and so on, the proposed approach consists of three algorithms: 1) conversation graph generation algorithm; 2) bullying SN generation algorithm; and 3) BFA. The first algorithm constructs a directed weighted conversation graph G_c by efficiently reconstructing the conversations from raw Twitter data while enabling a more accurate model of human interactions. The second algorithm constructs a bullying SN $\mathcal{B}$ to analyze the behavior of users in social media. The third algorithm consists of our proposed A&M centrality measures to identify bullies from $\mathcal{B}$. Fig. 3 shows the process flow of BullyNet where the raw data are extracted from Twitter using Twitter API from which the conversation graph is constructed for each conversation using Algorithm 1. Then, from the conversation graphs, a bullying SN is generated using Algorithm 2. Finally, the bullies from Twitter are identified by applying Algorithm 3.

Algorithm 1 Conversation Graph Generation

Input: Set of tweets, $T = \{t_1, \ldots, t_n\}$
Output: Conversation graphs $G_c = \{g_{c_1}, \ldots, g_{c_m}\}$

1) Sort all tweets in T in reverse-chronological order based on date of creation.
2) For each tweet t_i in T, where $1 \leq i \leq |T|$:
   a) If t_i does not belong to a conversation, then create a new conversation $c \in C$ and associate t_i with c.
   b) If there is a tweet $t' \in \{t_i, t_{i+1}, \ldots, t_{|T|}\}$ where $\mathcal{DID}(t_i) = \mathcal{SID}(t')$ then associate t' with all t_i's conversations.
3) For each conversation $c_i \in C$:
   a) Construct a conversation graph $g_{c_i} \in G_c$, where users are represented as nodes and tweets as edges.
   b) For each edge $e = (u, v)$ in g_{c_i}:
      i) Compute the sentiment of the tweet (SA).
      ii) Compute the cosine similarity (CS) of the tweet with bullying bag of words (CS).
      iii) Calculate the bullying indicator I_{t_i} (weight) of the edge as follows: $I_{uv} = \beta * SA + \gamma * CS$
4) Return G_c

A. Algorithm 1—Conversation Graph Generation

The conversation graph generation in Algorithm 1 is constructed from a set of tweets T to generate directed weighted conversation graphs G_c for each conversation. The weights between the nodes or users are determined by analyzing the sentiment behind the text of a tweet and examining for curse words. We then provide a score based on the expression the text represents. For each tweet t_i in T, the conversations are built by doing a binary search $\mathcal{DID}(t_i)$ with the $\mathcal{SID}$ of the remaining tweets. If a match is found as t', then it is appended with t_i to form a new conversation. If a binary search match is found with an already existing tweet in a conversation c_i, then t_i is appended to tweets in c_i. The graphs are represented as $G_c = (V, E, I)$, where V is the set of users involved in the conversation, E is the set of edges representing the tweets in the conversation, and each edge is assigned a bullying indicator value I as the edge weight which is in the range of $[-1, +1]$. When $I_{ij} = -1$, it indicates a negative interaction by i toward j, and when $I_{ij} = 1$, it indicates a positive interaction. The bullying indicator for each tweet is computed as $I = \beta * SA + \gamma * CS$, based on SA [Valence Aware Dictionary and sEntiment Reasoner (VADER)] and cosine similarities (CSs) with a list of commonly used insulting words. The factors of β and γ are 0.9 and 0.1, respectively, which are determined by the experiment (see Section VI-C).

Example 1: Fig. 4 shows the conversation extracted from the set of tweets $T = \{t_1, \ldots, t_7\}$. First, the tweets are sorted in descending order, i.e., $t_7, t_6, \ldots, t_1$. Next, $\mathcal{DID}(t_7)$ is searched with the $\mathcal{SID}$ of the remaining tweets (t_6 through t_1). A match is found in t_3 and conversation c_1 is formed. This process is repeated for each tweet. The conversations c_2 and c_3 are

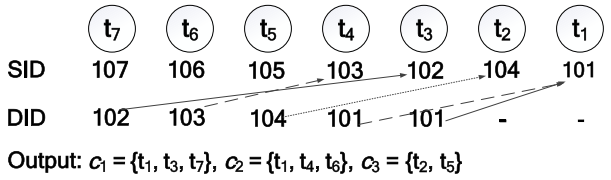


Fig. 4. Matching tweets based on $\mathcal{DID}$ and $\mathcal{SID}$ to construct conversations. Given tweets $\{t_1, \ldots, t_7\}$, the output is three conversations: c_1–c_3.

created with tweets $\{t_6, t_4\}$ and $\{t_5, t_2\}$, respectively. Since $\mathcal{DID}(t_4)$ and $\mathcal{DID}(t_3)$ match with the $\mathcal{SID}(t_1)$, the tweet t_1 is appended with t_4 and t_3. Therefore, the final conversations are $c_1 = \{t_7, t_3, t_1\}$, $c_2 = \{t_6, t_4, t_1\}$, and $c_3 = \{t_5, t_2\}$. This process can be represented by Algorithm 1.

In step 3 of Algorithm 1, a directed, weighted graph $g_{c_i} = (V, E)$ is constructed for every conversation c_i where nodes V, represented as the users, and the edges E, represented as the tweets, are directed from one user to another in a conversation. For every edge e, an edge weight is calculated as $I = \beta * SA + \gamma * CS$. This is known as the bullying indicator which is in range of $[-1, +1]$. The SA and CS are computed on the tweet (edge) to evaluate the emotion and behavior of the user. β and γ are constants, which will be determined by the experiment (see Section VI-C). In step 4, the algorithm outputs the conversation graphs G_c.

SA is the process of analyzing the sentiment of a message based on the user's opinion, attitude, and emotion toward an individual. Depending on the analysis, the polarity of the text is classified into positive, negative, or neutral. The sentiment reflects feeling or emotion, while emotion reflects attitude. There are different libraries or tools available to determine the sentiment of the content, which includes sarcasm, emoji, images, and so on. Some of them are: VADER, TextBlob, Python NLTK, and so on. We use VADER [39], which is a lexicon and rule-based SA tool that is specifically attuned to sentiments expressed in social media. It performs well with emojis, emoticons, slangs, and acronyms in a sentence. CS [40] measures the similarity between two vectors using their inner product. In Twitter, some tweets may contain curse or insulting words that are reasonable indications of the existence of bullying. Thus, we select a reference list of insulting words commonly used in Twitter and some external linguistic resources for insulting analysis seeds. This list contains words indicating curse or negative emotions such as nigga, bitch, and slut and are compared with individual tweets with CS to compute a score. In this context, each tweet and insulting seeds are represented as vectors, where each vector has the word frequencies.

Example 2: Fig. 5(a) shows a sample conversation of tweets. From Algorithm 1, the conversation graphs are constructed, as shown in Fig. 5(b). It contains two conversation graphs shown with dashed blue edges and with solid red edges. The rounded number on the edges indicates the tweet order of that particular conversation. Fig. 5(c) and (d) shows the two conversation graphs as g_{c_1} and g_{c_2} with the bullying indicator as the edge weight. With β and γ values as 0.9 and 0.1, respectively, which was determined experimentally (see Section VI-C), the edge weight I_{31}, i.e., the edge from $P3$ to $P1$, is calculated as -0.23. Similarly, the score of the other edges is calculated, as shown in Fig. 5(c) and (d).

B. Algorithm 2—Bullying SN Generation

In many real-world social systems, the relation between two nodes can be represented as SNs with positive and negative links. Since this research focuses on identifying the bullying nodes in the network, Algorithm 2 is designed to determine the final outgoing edge weight w_{ij} for the users in the conversation graphs G_c.

Algorithm 2 Bullying SN Generation

Input: Set of conversation graphs, G_c
Output: Bullying Signed Network $\mathcal{B}$

1) For each conversation graph g_{c_i} in G_c:
   a) For each set of edges with the same order, sorted ascendingly, compute the bullying score of source node u toward target node v for each edge $e = (u, v)$ as follows:
   $S_{uv} = I_{uv} + \alpha(I_{uv} - S_{vu})$.
   and then determine the average score of node u for the same set of edges.
   b) Compute the overall bullying score S of each node in g_{c_i} as follows:
      i) If the node is the *root* node, then: $S = \frac{\sum S}{1+2.2(n-1)}$
      ii) Otherwise: $S = \frac{\sum S}{2.2(n)}$
2) Construct the bullying SN graph $\mathcal{B}$ by merging all the conversation graphs together.
3) Return $\mathcal{B}$.

In step 1(a) of Algorithm 2, for every conversation graph g_{c_i}, a bullying score S is calculated based on how a node/user interacts with other nodes/users in the graph based on the tweet order (sorted in ascending order), i.e., tweets are arranged based on the conversation. For an edge $e = (u, v)$, the bullying score $S_{uv} \equiv I_{uv}$ if the edge toward v is not a reply from u. Otherwise, the bullying score S_{uv} is calculated as $I_{uv} + \alpha(I_{uv} - S_{vu})$, where α is a constant determined by the experiment as 0.6. Here, α is used to calculate how much percent of the difference between the sender and receiver should be taken to determine the bullying score S. I_{uv} is the bullying indicator between the nodes u to v and S_{vu} is the bullying score between the nodes v to u. The difference between I_{uv} and S_{uv} is that I_{uv} computes a score for the content on a tweet based on the SA and CS, whereas S_{uv} computes a score based on the entire conversation between u and v, i.e., the context in which opinion of u toward v. If there are more than one edge for a user with the same order, an average bullying score is computed for the same set of orders after the bullying score is evaluated.

Example 3: Table III shows the bullying score calculation for the conversation graph g_{c_1} in Fig. 5(c). In order 1, the bullying score $S_{21} = I_{21} = 0$ since the edge from $P2$ to $P1$

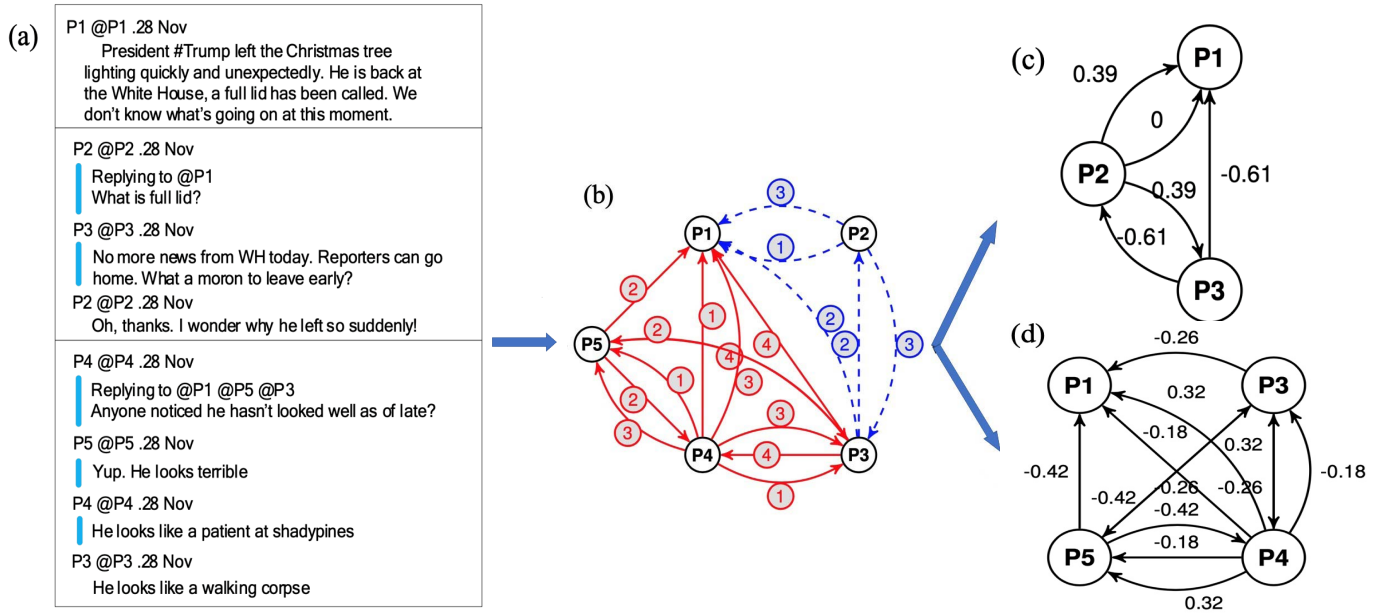


Fig. 5. Conversation graph generation. (a) Sample conversation of tweets. (b) Conversations graph. Blue and red subgraphs represent two different conversations. (c) and (d) Bullying indicators are added as edge weights to the conversation graphs.

TABLE III
BULLYING SCORE TABLE FOR g_{c_1}

Tweet #	P1	P2	P3
1	-	0 (P1)	
2	-		-0.61 (P1,P2)
3	-	0.99 (P3) 0.39 (P1)	
Total	-	0.69	-0.61

TABLE IV
BULLYING SCORE TABLE FOR g_{c_2}

Tweet #	P1	P4	P5	P3
1	-	-0.18 (P1,P3,P5)		
2	-		-0.56(P4) -0.42(P1,P3)	
3	-	0.84(P5) 0.32(P1,P3)		
4	-			-0.60(P4) -0.16(P5) -0.26(P1)
Total	-	0.4	-0.49	-0.34

is not a reply edge. The user in the parenthesis represents to whom the edge responds. In order 2, there are two edges from $P3$ to $P1$ and $P3$ to $P2$ and the bullying score $S_{31} = -61$ and $S_{32} = -61$ is the same as I_{31} and I_{32}, respectively. The order 3 also has two edges, $P2$ to $P3$ and $P2$ to $P1$. Since the edge $P2$ to $P3$ is a reply to the edge $P3$ to $P2$, the bullying score is calculated as $S_{23} = I_{23} + \alpha(I_{23} - S_{32}) = 0.99$ where $\alpha = 0.6$ was determined by the experiment. Next, the average of the score for the same order of the user is computed, i.e., order 2 of the user $P3$ is -0.61 and order 3 of the user $P2$ is 0.69. Following a similar approach, the bully score S is calculated in Table IV for the second conversation graph g_{c_2} in Fig. 5(d).

In step 1(b), the bullying score that was computed in the previous step for the users in every conversation graph $g_{c_i 1}$ is normalized in $[-1, 1]$. The normalization is performed in two ways, i.e., for the user that initiated the conversation, known as root nodes, and the users that are involved in the conversation. For the first type of users, the normalization is computed as $\sum S/(1 + 2.2(n-1))$, and for the second type of users, the normalization is computed as $\sum S/2.2(n)$ where n is the number of times the user occurs in the order and the value 2.2 is computed using $1 + (\text{Maxdiff})(\alpha)$ in which Maxdiff is the range, i.e., 2. This normalized score of the users becomes the edge weight to the other users in g_{c_i}.

In step 2, the bullying SN graph $\mathcal{B}$ is constructed by merging all the conversation graphs G_c. If there is more than one edge, i.e., $e = (u, v)$, then a single edge weight is calculated by taking the difference between average and standard deviation of all w_{uv}. Step 4 outputs the bullying SN graph $\mathcal{B}$.

Example 4: Fig. 6(c) shows the bullying SN by merging the two normalized conversation graphs in Fig. 6(a) and (b). From Fig. 6, it can be seen that there are two different edges from the user $P3$ to $P1$ (-0.27 and -0.15). Therefore, the difference between the average and the standard deviation of the two edges is calculated as -0.15 that is the final edge weight of $P3$ to $P1$ in the bullying SN.

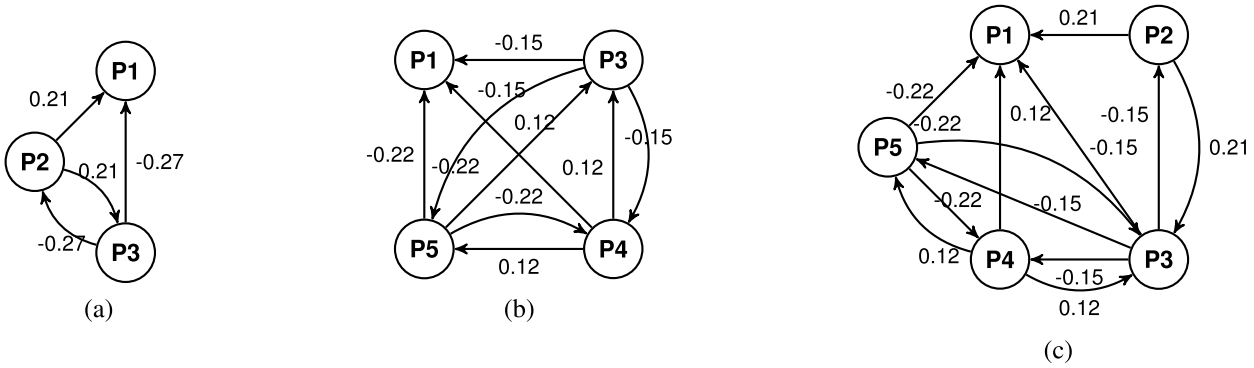


Fig. 6. Normalized conversation graphs. (a) g_{c_1}, (b) g_{c_2}, and (c) bullying SN.

C. Algorithm 3—Bully Finding

This work is to identify bullies from $\mathcal{B}$ using centrality measures. Since this article is about social networks, the importance is defined as the behavior. Among several centrality measures, we consider BAD by Mishra and Bhattacharya [34] a state-of-the-art method that handles SN because their measure is computed on how the outgoing edge from a node/user depends on the incoming edges from other nodes/users. However, BAD is modeled on a trust-based network, i.e., the users that have a propensity to trust/distrust other users. Also, the edge weight denotes the trust score rather than the bullying score as in this research.

Therefore, we proposed a centrality measure $A\&M$, similar to that of BAD to identify bullies from our proposed SN $\mathcal{B}$. Merit is a measure of the opinion (good or bad) that the other nodes have toward a particular node and Attitude is a measure of the behavior of a node toward the other node. However, in a given bullying singed network, the attitude or likes or dislike of a node toward other nodes in the network is not known. Therefore, the expressions to compute the Merit and Attitude metrics in a mutually recursive manner

$$M^{n+1}(j) = \frac{1}{2|\text{in}(j)|} \sum_{k \in \text{in}(j)} \big(w_{kj}\big)\big(A^n(i)\big) \tag{1}$$

$$A^{n+1}(i) = \frac{1}{2|\text{out}(i)|} \sum_{j \in \text{out}(i)} \big(w_{ij} + X_{ij}\big)$$

$$X_{ij} = \begin{cases} M(j), & \text{if } (w_{ij} \times M(j)) > 0 \\ -M(j), & \text{otherwise.} \end{cases} \tag{2}$$

Let in(j) denote the set of all incoming edges to node j and out(i) denote the set of all outgoing edges from node i. Normalization is done to maintain the value in the range of $[-1, 1]$. An auxiliary variable X_{ij} is introduced to measure the effect of the merit score of a node j on its incoming edge to node i. Since merit is about whether the node is considered good or bad, it is calculated to be the sum of all its incoming edges from other nodes. Likewise, since attitude is about the particular node's view of others, it is calculated using the outgoing edges of a node toward others and its corresponding merit score in the network. Although we use two metrics similar to BAD, the calculation of the incoming and the outgoing edges of a node differs. Since Bias in BAD is about how truly it rates other nodes, it is calculated by the difference in the edge weight and the real trust of a node (deserve). The explanation of the proposed metric follows.

TABLE V

EXAMPLE SHOWING THE VALUES OF THE GRAPH [SEE FIG. 6(c)] AFTER EACH ITERATION. A DENOTES ATTITUDE AND M DENOTES MERIT

No.	P1		P2		P3		P4		P5	
	M	A	M	A	M	A	M	A	M	A
0	-1	-	-1	-1	-1	-1	-1	-1	-1	-1
1	0.02	-	0.01	0.11	-0.01	-0.13	0.09	0.06	0.01	-0.13
2	0.01	-	0.02	0.11	0.1	-0.11	0.01	0.06	0.0	-0.11
3	0.01	-	0.01	0.11	0.00	-0.11	0.01	0.06	0.0	-0.11
4	0.01	-	0.01	0.11	0.00	-0.11	0.01	0.06	0.0	-0.11

From the above expression, it can be seen that if the outgoing edge weight from node i to node j has a positive value and the merit score of node j is also positive, then the attitude of node i to j is calculated by the sum of both values. If the outgoing edge weight from node i to j is negative and the merit score of node j is positive or vice versa, then the attitude of node i to j is calculated by subtracting the merit score from the edge weight, which means that if a node has a positive edge weight toward a benign merit node, then the attitude score increases. Similarly, holding a negative edge weight toward a benign merit node decreases that node's attitude score. However, if a node has a positive edge weight toward a negative merit node, the attitude of a node decreases.

From (1) and (2), the attitude of a node depends on the merit of its neighbors and vice versa. A fixed-point iteration method is used to obtain the solution. The Merit and Attitude of node i at iteration n are denoted by $A^n(i)$ and $M^n(i)$, respectively. The proposed Algorithm 3 is designed to compute merit and attitude scores for each node in the network. Initially, we start with a Merit and Attitude score of -1 (i.e., the first iteration) in step 1. In step 2a, the merit scores for each node are updated using the attitude scores from the previous iteration. In step 2b, the attitude scores are updated using the newly updated Merit scores in the same iteration. Both Merit and Attitude scores are mutually recursive and are updated until both the scores converge in step 3. The scores of Merit and Attitude from the last iteration are the final scores. In the final step 4, all the nodes whose attitude score is less than zero are added the list L along with the user's attitude score.

Algorithm 3 BFA

Input: Bullying Signed Network $G_s = (V, E, W)$

Output: List of bullies and its attitude score $L = [(u_1, s_1), (u_2, s_2), \ldots, (u_{|L|}, s_{|L|})]$

1) Initialize $M^0(v) = -1$ and $A^0(v) = -1$, $\forall v \in V$.
2) Set iteration index i = 1
   a) For each $v \in V$ compute merit score $M^i(v) = \frac{1}{2|in(v)|} \sum_{u \in in(v)} (w_{uv})(A^{i-1}(u))$ where $|in(v)|$ is the number of incoming edges to the node v
   b) For each $u \in V$ compute attitude score $A^i(u) = \frac{1}{2|out(u)|} \sum_{v \in out(u)} (w_{uv} + X_{uv})$ where $|out(u)|$ is the number of outgoing edges from the node u
3) If there exist atleast one $v \in V : M^i(v) \neq M^{i-1}(v)$ or $A^i(v) \neq A^{i-1}(v)$
   a) Increase the iteration index $i = i + 1$
   b) Repeat step 2a & 2b for each iteration
4) For each $v \in V$ add the node and its corresponding attitude score value greater than 0 to the list L
5) Return L

Example 5: Table V shows the value of A&M that are updated after each iteration by applying Algorithm 3 to Fig. 6(c). The Attitude column of node $P1$ is blank because there are no outgoing edges from $P1$. The last iteration shows the final A&M score of the nodes. It can be seen that nodes $P3$ and $P5$ are bullies with a confidence score of 0.11 and 0.11, respectively.

V. Algorithm Analysis

In this section, we show the proof of convergence of the centrality measure and perform a complex analysis of our proposed approach.

A. Convergence of Centrality Measure

We start the convergence proof by showing the difference between the attitude of a node at any iteration and the infinite iteration is bounded, which then leads to convergence by proving the error bound $\epsilon, \ll 1$.

After a certain iteration t, the attitude score of that iteration becomes close to A^∞. Since the merit of a node can be expressed in terms of attitude of other nodes, this implies that merit values exhibit similar properties.

Proposition 1: A&M of a node at any iteration n and the infinite iteration is bounded by an inverse exponential function of n.

Proof: We prove this in Appendix.

B. Complexity Analysis

Proposition 2: The overall complexity of our proposed approach in the average case is $\mathcal{O}(k \times l + \log n)n$.

Proof: We can determine the time complexity of the proposed approach in three phases: constructing conversation graph, constructing bullying SN, and bully finding.

1) Constructing Conversation Graphs Phase: In the constructing conversation phase, the runtime complexity is the time taken to construct m conversations from n tweets and then generate graphs from the constructed conversations.

Initial sorting of tweets uses merge sort, which takes a computational time of $\mathcal{O}(n \log n)$. The conversation is constructed by doing a binary search on $\mathcal{DID}$ and $\mathcal{SID}$ of the conversation tweet and the current tweet, respectively, leading to m conversations with a computational time of $\mathcal{O}(n \log n)$. The cost for generating graph from the conversations is $\mathcal{O}(m)$. Therefore the average computational cost to construct conversation graphs is $\mathcal{O}(n \log n + n \log n + m) = \mathcal{O}(n \log n + m)$.

2) Constructing Bullying SN Phase: In the constructing bullying SN phase, we traverse through each conversation graph where the bullying score is calculated for each node with respect to the edges with the same order. For each conversation graph m, the maximum number of nodes in the worst case is k. Therefore, the total computational cost is $\mathcal{O}(n \cdot k + m \cdot k)$.

3) Bully Finding Phase: In the bully finding phase, the runtime is the time taken to detect the bullying users using A&M centrality. For each l number of iterations, A&M centrality touches each edge atmost twice. Therefore, the average case in detecting bullies in each iteration is $\mathcal{O}(2n \cdot k)$, and for the given l iteration, it is $\mathcal{O}(n \cdot k \cdot l)$.

Therefore, the overall complexity of our proposed approach in the average case is:

$\mathcal{O}((k \cdot l + \log n)n + k \cdot m) = \mathcal{O}(k \cdot l + \log n)n$ since $m,\ k \ll n$.

VI. Experimental Evaluation

In this section, we evaluate the performance of the proposed algorithms. First, we present the data used in our evaluation. Second, we discuss the implementation details and the way we process it to build ground truth. Finally, we present the experimental results that include determining the coefficients α, β, and γ, utility, and scalability.

A. Data Set

In this article, we rely on Twitter's Streaming API, which provides free access to 1% of all tweets. The API returns each tweet in a JSON format, with the content of the tweet, metadata (e.g., creation time, source ID, destination ID, and reply/retweet), as well as information about the poster (e.g., username, followers, and friends). To prevent our own bias, we first randomly chose 5000 interconnected users and collected all the tweets in JSON format totaling 5.6 M within a six-month time frame between May and October 2017. We then extracted features, such as username, text, replyname,

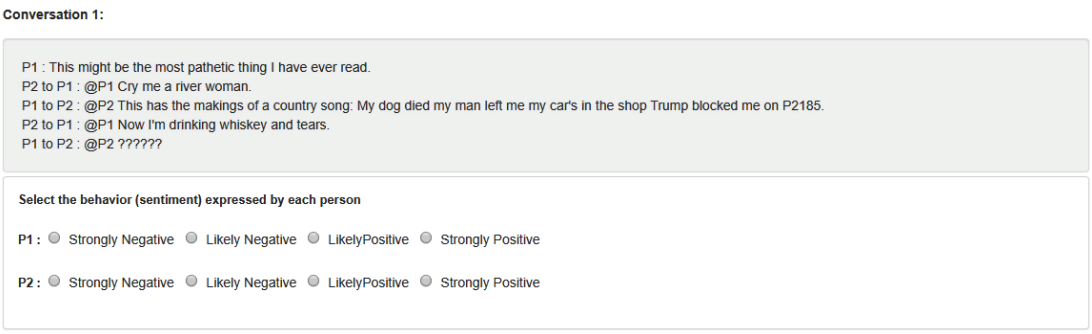


Fig. 7. Sample user interface of Amazon Mechanical Turk survey (positive: appropriate behavior and negative: inappropriate behavior).

and mentions, and network-based features, such as source ID and destination ID from the Twitter JSON. There were about 2% of the tweets that were in languages other than English. When examining the users, about 90% of their geographical location were in USA, 6% of the users' location were in U.K. and the remaining 4% were from Ecuador, Japan, and China.

B. Implementation and Setup

We implemented our algorithm in Java, and our experiments were conducted on a machine equipped with an Intel Core i7-8550U CPU at 2.00-GHz processor and 16.0-GB RAM, running Windows 10 64-bit operating system.

We employed Amazon *Mechanical Turk* (MTurk) workers to respond to an online survey that we developed. We provided 2700 surveys with each survey consisting of ten conversations. Each survey was assigned to three workers to classify the bullying behavior of the users in the conversations according to four predefined labels (strongly positive, likely positive, likely negative, and strongly negative) to avoid biased interpretation of bullies. Overall, the workers rated 27 000 conversations containing 1700 users, which were extracted from the set of raw Twitter data by using Algorithm 1. The MTurk UI enables requesters to create and publish surveys (HITs) in a batch when processing many HITs of the same type thus saving time. For our study, we created a csv file that contained 2700 HITs. MTurk automatically created a separate HIT for each set of conversation in the csv file, as shown in Fig. 7. The results to rate each users involved in the set of conversations were obtained from the workers. A significant share of the participants for the survey came from USA, Canada, Europe, and India. There was not a marked variation in the rating provided by the workers. There were about 7978 strongly negative, 47 426 likely negative, 56 704 likely positive, and 23 762 strongly positive user interactions. Some of these users appear in few conversations, and therefore, we collect these ratings based on the users and number of workers and compute using a metric to identify 569 users as bullies. Finally, the results are normalized to form the ground truth. We analyze and compute the ground truth in a metric, which results in bullying and nonbullying users. Having the computed results as ground truth, we evaluated the performance measure by experimenting the proposed algorithms results with respect to the number of users increasing linearly from 500 to 1700.

C. Determining Optimal Values for Coefficients α, β, and γ

Recall that $I_{uv} = \beta \times SA + \gamma \times CS$ in Algorithm 1 and $S_{uv} = I_{uv} + \alpha(I_{uv} - S_{vu})$ in Algorithm 2. To determine the coefficient β and γ for bullying indicator I and α for the bullying score S, we generate input tweets of varying length and performed experiment for different values of α, β, and γ, that is, with 5.7 million tweets data set, we did experiment for the tweets ranging from 1M, 2M, 3M, 4M, and 5M for different α, β, and γ values. After experimenting with different values, we found that the coefficient values of $\beta \geq 0.6$, $\gamma \leq 0.4$, and $\alpha \leq 0.6$ to provide the greatest accuracy. The accuracy was measured with $\beta \geq 0.6$ and $\gamma \leq 0.4$ for every $\alpha \leq 0.6$ with respect to the ground truth, using the F1 Measure [41].

Fig. 8 shows the optimal values for the coefficients α, β, and γ with respect to the β and γ values, which are set from 60 to 90 and 40 to 10, respectively. We use three different α values for every bullying indicator coefficients β and γ, which varies from 0.4 to 0.6. In our approach, we observe that the F1 measure increases linearly when the coefficients β increase and γ decrease. We also observe that when we increase the α value, the F1 measure increases in all the cases, indicating that the SA has more impact on the bullying indicator than the CS. This is because SA analyzes not only the text but also emojis, emoticons, and determining the CS alone hurts the performance. Hence, we take advantage of both SA and Cosine. Similarly, the response to a tweet has a direct effect on the bullying score.

D. Utility

We briefly introduce our evaluation metrics that will be used to determine the accuracy of our approach.

1) *AccuracyCM [42]:* The accuracy measure is the ratio of the number of bully users detected to the total number of bullies. It does not perform well with imbalanced data sets

$$\text{AccuracyCM} = \frac{\#\text{ of detected bullies}}{\text{total number of bullies}}.$$

2) *Precision and Recall [43]:* Precision and recall are evaluation metrics used in binary classification tasks. Precision is the measure of exactness and recall is the

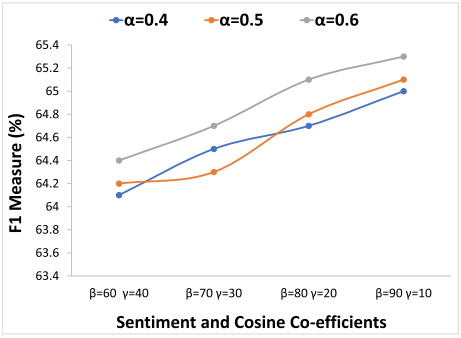


Fig. 8. Determining optimal values for coefficients α, β, and γ.

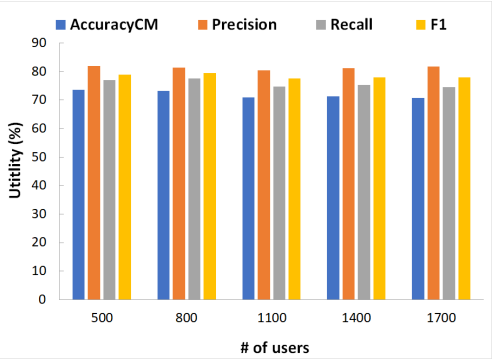


Fig. 9. Utility with respect to the number of users.

measure of completeness. They are defined as follows:

$$\text{Precision} = \frac{\#\text{ of true bullies detected}}{\text{total number of detected users}}$$

$$\text{Recall} = \frac{\#\text{ of true bullies detected}}{\text{total number of true bullies}}.$$

In simple terms, high precision means that an algorithm returned substantially more bully users, whereas high recall means that an algorithm returned most of the bullies.

3) *F1 Measure [41]:* F1 Measure is the harmonic mean between precision and recall. The range for F1 is [0, 1]. It measures how many bullies are identified correctly and how robust it is. Mathematically, it can be expressed as

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}.$$

F1 Measure attempts to find a balance between precision and recall. The greater the F1 Measure, the better is the performance of our approach.

To determine the accuracy of our proposed centrality measure, A&M, we compare all the evaluation metrics discussed above with respect to the number of users increasing linearly from 500 to 1700 users. Fig. 9 shows the utility values of the metrics (accuracyCM, precision, recall, and F1 Measure) with respect to the number of users generated from Algorithm 2 as the input.

For the number of users ranging from 500 to 1700, we observed that the AccuracyCM metric ranged between 70.8% and 73.6% and can be biased in the case of unbalanced data sets; however, it produces better results when false positives (is an error in bullies detection in which a detection result improperly indicates that a user is bully when in reality the user is not a bully) and false negatives (is an error in which a test detection improperly indicates that a user is not bully when in reality the user is a bully) are almost even. In this case of uneven distribution of data, we measure the accuracy with F1 Measure, which ranges from 77.5% to 79.4%, whereas the precision and recall center around 81% and 76%, respectively. Therefore, from Fig. 9, it can be seen that the precision outperforms other metrics, i.e., higher the precision means that our algorithm identifies more bullies precisely among the total number of users. The percentages mentioned above for all the metrics remained almost consistent even with the increase in the number of users.

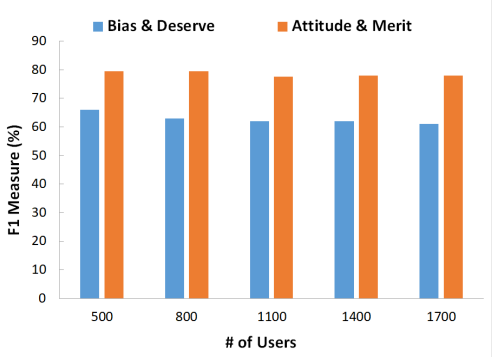


Fig. 10. Comparative evaluation of the proposed centrality measure A&M with BAD.

Next, we compare the performance of our proposed centrality measure A&M with the research work done by Mishra and Bhattacharya [34]—BAD which is explained in Section IV-C. We compare the F1-score in terms of accuracy achieved with respect to the number of users generated from Algorithm 2 as the input. Fig. 10 shows the comparison of the centrality measures with respect to the number of users increasing linearly from 500 to 1700 users.

In our approach, we observed that *A&M* has an accuracy of about 80%. Also, our centrality measures outperform BAD in all the cases, i.e., number of users. As the number of users increased from 500 to 1700, the accuracy of BAD decreased from 65% to 60%, whereas the proposed centrality measures A&M stays consistent. There can be multiple reasons behind it. First of all, the bias score of a node with highly positive bias decreases when it has an outgoing edge with positive weight whereas in *A&M*, the Attitude score increases when a positive node has an outgoing edge with positive weight. Next, when calculating the deserve for a node, the bias value is taken in range of $[0, 1]$, whereas in *A&M*, merit is calculated with the attitude value in the range of $[-1, 1]$. Furthermore, BAD does not perform well when a node has fewer outgoing and incoming edges. Nevertheless, it is still outperformed by the *A&M* centrality.

We also compare the accuracy of our BullyNet algorithm with Chatzakou *et al.* [24], Zhao *et al.* [18], and Singh *et al.* [19]. As shown in Table VI, the *F1 score* of BullyNet outperforms all the other methods. However, Precision and Recall of BullyNet are outperformed in [18] and [19], respectively.

TABLE VI
PERFORMANCE COMPARISONS OF DIFFERENT METHODS

	Precision	Recall	F1 Score
Chatzakou *et al.* [24]	75	53	79
Zhao *et al.* [18]	76.8	79.4	78.0
Singh *et al.* [19]	82	53	64
BullyNet	**81.3**	**77.6**	**79.4**

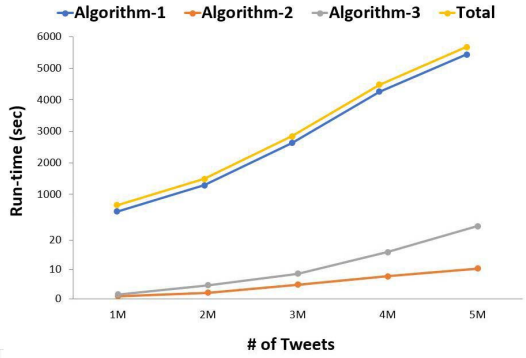


Fig. 11. Scalability with respect to the number of tweets.

E. Scalability

We measure the scalability of BullyNet with respect to the number of tweets and observe the run times of our three algorithms: conversation graphs generation, bullying SN generation, and bully finding with optimal values for coefficients α, β, and γ set at 0.6, 0.9, and 0.1, respectively.

We observed that running a data set with 1M records takes up to 8 min for the BullyNet algorithm and the runtime increases linearly as the record size increases linearly from 1M to 5M. Fig. 11 shows the runtime for the records size from 1M to 5M for each data set. We also observed that the most dominant algorithm of our experiment is conversation graphs generation which took the majority of run time, i.e., approximately 70% of total execution time of the three algorithms. This is due to the fact that the conversation graphs have to calculate SA and CS for each tweet and then calculate the corresponding bullying indicator I as the edge weight for each conversation graph.

We observed that there is a linear increase in total runtime with an increase in a number of tweets. However, we also observed that the bullying SN generation algorithm (Algorithm 2) runtime did not grow linearly with the increase in records, rather it tends to remain constant. This is because there are k number of nodes in m conversation graphs. Therefore, to calculate the bullying score for each graph, it takes $\mathcal{O}(k)$ and does not affect the runtime with the growth in a number of tweets. We can observe that similar to the first algorithm, the runtime of the third algorithm also increases linearly with record size. The variation is attributed to the increase in the number of users in each tweet resulting in corresponding increase in computation time for centrality measures.

VII. CONCLUSION AND FUTURE WORK

Although the digital revolution and the rise of social media enabled great advances in communication platforms and social interactions, a wider proliferation of harmful behavior known as bullying has also emerged. This article presents a novel framework of BullyNet to identify bully users from the Twitter social network. We performed extensive research on mining SNs for better understanding of the relationships between users in social media, to build an SN based on bullying tendencies. We observed that by constructing conversations based on the context as well as content, we could effectively identify the emotions and the behavior behind bullying. In our experimental study, the evaluation of our proposed centrality measures to detect bullies from SN, and we achieved around 80% accuracy with 81% precision in identifying bullies for various cases.

There are still several open questions deserving further investigation. First, our approach focuses on extracting emotions and behavior from texts and emojis in tweets. However, it would be interesting to investigate images and videos, given that many users use them to bully others. Second, it does not distinguish between bully and aggressive users. Devising new algorithms or techniques to distinguish bullies from aggressors would prove critical in better identification of cyberbullies. Another topic of interest would be to study the relationship between conversation graph dynamics and geographic location and how these dynamics are affected by the geographic dispersion of the users? Are the proximity increase the bullying behavior?

APPENDIX
CONVERGENCE OF CENTRALITY MEASURE

Proposition 1: A&M of a node at any iteration n and the infinite iteration is bounded by inverse exponential function of n.

Proof: By using mathematical induction, we prove the convergence of attitude. Given its definition, the attitude score $A^{\infty}(i)$ and $A^{t+1}(i)$ can be written as

$$A^{\infty}(i) = \left| \frac{1}{2|\text{out}(i)|} \sum_{j \in \text{out}(i)} \left\{ w_{ij} \pm \frac{1}{2|\text{in}(j)|} \sum_{k \in \text{in}(j)} \left(w_{kj} \times A^{\infty}(k) \right) \right\} \right|$$

$$A^{n+1}(i) = \left| \frac{1}{2|\text{out}(i)|} \sum_{j \in \text{out}(i)} \left\{ w_{ij} \pm \frac{1}{2|\text{in}(j)|} \sum_{k \in \text{in}(j)} \left(w_{kj} \times A^{n}(k) \right) \right\} \right|.$$

Base Case: For n = 1, we have

$$= \left| \frac{1}{2|\text{out}(i)|} \sum_{j \in \text{out}(i)} \left\{ w_{ij} \pm \frac{1}{2|\text{in}(j)|} \sum_{k \in \text{in}(j)} w_{kj} \left(A^{\infty}(k) \right) - A^{0}(k) \right\} \right|$$

$$\leq \frac{1}{2|\text{out}(i)|}\sum_{j\in\text{out}(i)}\left\{|w_{ij}| \pm \frac{1}{2|\text{in}(j)|}\sum_{k\in\text{in}(j)}|w_{kj}|\left|\left(A^{\infty}(k)\right)-A^{0}(k)\right|\right\}$$

$$\left[\because |x.y| \leq |x||y| \ |w_{ij}| \text{ and } |w_{kj}| \leq 1\right]$$

$$\leq \frac{1}{2|\text{out}(i)|}\sum_{j\in\text{out}(i)}\left\{\frac{1}{2|\text{in}(j)|}\sum_{k\in\text{in}(j)}\left|\left(A^{\infty}(k)\right)-A^{0}(k)\right|\right\}$$

$$\leq \frac{1}{2|\text{out}(i)|}\sum_{j\in\text{out}(i)}\left\{\frac{1}{2|\text{in}(j)|}\sum_{k\in\text{in}(j)}2\right\}.$$

Since $A(k)\in[-1,+1]$, we have $|A^{\infty}(k)-A^{0}(k)|\leq 2$

$$\leq \frac{1}{2|\text{out}(i)|}\sum_{j\in\text{out}(i)}\left\{\frac{1}{2|\text{in}(j)|}2|in(j)|\right\}=\frac{1}{2}.$$

Induction Step: We assume the bound to be true for $A^{n}(i)$ so, by the hypothesis $|A^{\infty}(i)-A^{n}(i)|\leq 1/(2^{n+2})$. In the $(n+1)$th iteration

$$\left|A^{\infty}(i)-A^{n}(i)\right| = \left|\frac{1}{2|\text{out}(i)|}\sum_{j\in\text{out}(i)}\left\{w_{ij}\pm\frac{1}{2|\text{in}(j)|}\sum_{k\in\text{in}(j)}w_{kj}\left(A^{\infty}(k)\right)-A^{n}(k)\right\}\right|$$

$$\leq \frac{1}{2|\text{out}(i)|}\sum_{j\in\text{out}(i)}\left\{\frac{1}{2|\text{in}(j)|}\sum_{k\in\text{in}(j)}\left|\left(A^{\infty}(k)\right)-A^{n}(k)\right|\right\}$$

$$\leq \frac{1}{2|\text{out}(i)|}\sum_{j\in\text{out}(i)}\left\{\frac{1}{2|\text{in}(j)|}\sum_{k\in\text{in}(j)}\frac{1}{2^{n}}\right\}=\frac{1}{2^{n+2}}.$$

Therefore, the error is bounded by an inverse exponential function. Thus, we conclude that a convergence has been achieved in determining the measures "attitude" and "merit." □

REFERENCES

[1] J. Tang, C. Aggarwal, and H. Liu, "Recommendations in signed social networks," in *Proc. 25th Int. Conf. World Wide Web*, Apr. 2016, pp. 31–40.

[2] D. Liben-Nowell and J. Kleinberg, "The link-prediction problem for social networks," *J. Amer. Soc. Inf. Sci. Technol.*, vol. 58, no. 7, pp. 1019–1031, 2007.

[3] U. Brandes and D. Wagner, "Analysis and visualization of social networks," in *Graph Drawing Software*. Amsterdam, The Netherlands: Elsevier, 2004, pp. 321–340.

[4] X. Hu, J. Tang, H. Gao, and H. Liu, "Social spammer detection with sentiment information," in *Proc. IEEE Int. Conf. Data Mining*, Dec. 2014, pp. 180–189.

[5] E. E. Buckels, P. D. Trapnell, and D. L. Paulhus, *Trolls Just Want to Have Fun*. Springer, 2014, pp. 67:97–102.

[6] S. Kumar, F. Spezzano, and V. S. Subrahmanian, "Accurately detecting trolls in slashdot zoo via decluttering," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining (ASONAM)*, Aug. 2014, pp. 188–195.

[7] J. W. Patchin and S. Hinduja, "2016 cyberbullying data," Cyberbullying Res. Center, Tech. Rep. 2016, 2017.

[8] Cyberbullying Research Center. *State Bullying Laws in America*. Accessed: Jul. 1, 2020. [Online]. Available: https://cyberbullying.org/bullying-laws

[9] D. Cartwright and F. Harary, "Structural balance: A generalization of Heider's theory," *Psychol. Rev.*, vol. 63, no. 5, p. 277, Sep. 1956.

[10] J. Leskovec, D. Huttenlocher, and J. Kleinberg, "Signed networks in social media," in *Proc. 28th Int. Conf. Hum. Factors Comput. Syst. (CHI)*, 2010, pp. 1361–1370.

[11] R. Plutchik, "A general psychoevolutionary theory of emotion," in *Theories of Emotion*. 1980, pp. 3–33.

[12] W. Medhat, A. Hassan, and H. Korashy, "Sentiment analysis algorithms and applications: A survey," *Ain Shams Eng. J.*, vol. 5, no. 4, pp. 1093–1113, Dec. 2014.

[13] L. Tang and H. Liu, "Community detection and mining in social media," *Synth. Lectures Data Mining Knowl. Discovery*, vol. 2, no. 1, pp. 1–137, Jan. 2010.

[14] S. Bhagat, G. Cormode, and S. Muthukrishnan, "Node classification in social networks," in *Social Network Data Analytics*. 2011, pp. 115–148.

[15] J. Tang, Y. Chang, C. Aggarwal, and H. Liu, "A survey of signed network mining in social media," in *Proc. ACM Comput. Surv.*, vol. 3, 2016, pp. 42:1–42:37.

[16] J. Kunegis, J. Preusse, and F. Schwagereit, "What is the added value of negative links in online social networks?" in *Proc. 22nd Int. Conf. World Wide Web (WWW)*, 2013, pp. 727–736.

[17] Z. Wu, C. C. Aggarwal, and J. Sun, "The troll-trust model for ranking in signed networks," in *Proc. 9th ACM Int. Conf. Web Search Data Mining*, Feb. 2016, pp. 447–456.

[18] R. Zhao, A. Zhou, and K. Mao, "Automatic detection of cyberbullying on social networks based on bullying features," in *Proc. 17th Int. Conf. Distrib. Comput. Netw.*, Jan. 2016, pp. 1–6.

[19] V. K. Singh, Q. Huang, and P. K. Atrey, "Cyberbullying detection using probabilistic socio-textual information fusion," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining (ASONAM)*, Aug. 2016, pp. 884–887.

[20] H. Hosseinmardi, S. A. Mattson, R. I. Rafiq, R. Han, Q. Lv, and S. Mishra, "Detection of cyberbullying incidents on the Instagram social network," *CoRR*, vol. 1503.03909, 2015.

[21] J.-M. Xu, X. Zhu, and A. Bellmore, "Fast learning for sentiment analysis on bullying," in *Proc. 1st Int. WISDOM*, 2012, pp. 10:1–10:6.

[22] A. Squicciarini, S. Rajtmajer, Y. Liu, and C. Griffin, "Identification and characterization of cyberbullying dynamics in an online social network," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining*, Aug. 2015, pp. 280–285.

[23] P. Galán-García, J. G. De La Puerta, C. L. Gómez, I. Santos, and P. G. Bringas, "Supervised machine learning for the detection of troll profiles in Twitter social network: Application to a real case of cyberbullying," *Logic J. IGPL*, vol. 24, no. 1, pp. 42–53, 2015.

[24] D. Chatzakou, N. Kourtellis, J. Blackburn, E. De Cristofaro, G. Stringhini, and A. Vakali, "Mean birds: Detecting aggression and bullying on Twitter," in *Proc. ACM Web Sci. Conf.*, Jun. 2017, pp. 13–22.

[25] L. Cheng, J. Li, Y. N. Silva, D. L. Hall, and H. Liu, "XBully: Cyberbullying detection within a multi-modal context," in *Proc. 12th ACM Int. Conf. Web Search Data Mining*, Jan. 2019, pp. 339–347.

[26] H.-T. Kao, S. Yan, D. Huang, N. Bartley, H. Hosseinmardi, and E. Ferrara, "Understanding cyberbullying on Instagram and Ask.Fm via social role detection," in *Proc. Companion Proc. World Wide Web Conf.*, May 2019, pp. 183–188.

[27] L. Cheng, R. Guo, Y. Silva, D. Hall, and H. Liu, "Hierarchical attention networks for cyberbullying detection on the Instagram social network," in *Proc. SIAM Int. Conf. Data Mining*. Philadelphia, PA, USA: SIAM, 2019, pp. 235–243.

[28] L. Cheng, J. Li, Y. Silva, D. Hall, and H. Liu, "PI-bully: Personalized cyberbullying detection with peer influence," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Aug. 2019, pp. 5829–5835.

[29] J. Tang, C. Aggarwal, and H. Liu, "Node classification in signed social networks," in *Proc. SIAM Int. Conf. Data Mining*, Jun. 2016, pp. 54–62.

[30] S. Brin and L. Page, "The anatomy of a large-scale hypertextual Web search engine," *Comput. Netw. ISDN Syst.*, vol. 30, nos. 1–7, pp. 107–117, Apr. 1998.

[31] M. Shahriari and M. Jalili, "Ranking nodes in signed social networks," *Social Netw. Anal. Mining*, vol. 4, no. 1, p. 172, Dec. 2014.

[32] C. D. Kerchove and P. V. Dooren, "The PageTrust algorithm: How to rank Web pages when negative links are allowed?" in *Proc. SIAM Int. Conf. Data Mining*, Apr. 2008, pp. 346–352.

[33] P. Bonacich and P. Lloyd, "Calculating status with negative relations," *Social Netw.*, vol. 26, no. 4, pp. 331–338, Oct. 2004.

[34] A. Mishra and A. Bhattacharya, "Finding the bias and prestige of nodes in networks based on trust scores," in *Proc. 20th Int. Conf. World Wide Web (WWW)*, 2011, pp. 567–576.
[35] M. Dadvar, D. Trieschnigg, R. Ordelman, and F. de Jong, "Improving cyberbullying detection with user context," in *Proc. Eur. Conf. IR*, 2013, pp. 693–696.
[36] K. Dinakar, R. Reichart, and H. Lieberman, "Modeling the detection of textual cyberbullying," in *Proc. Int. AAAI WSM*, vol. 5, no. 1, 2011.
[37] Y. Chen, Y. Zhou, S. Zhu, and H. Xu, "Detecting offensive language in social media to protect adolescent online safety," in *Proc. Int. Conf. Privacy, Secur., Risk Trust Int. Conf. Social Comput.*, Sep. 2012, pp. 71–80.
[38] F. J. Ortega, J. A. Troyano, F. L. Cruz, C. G. Vallejo, and F. Enríquez, "Propagation of trust and distrust for the detection of trolls in a social network," *Comput. Netw.*, vol. 56, no. 12, pp. 2884–2895, Aug. 2012.
[39] C. J. Hutto and E. Gilbert, "Vader: A parsimonious rule-based model for sentiment analysis of social media text," in *Proc. Int. AAAI WSM*, vol. 8, no. 1, 2014.
[40] J. Han, J. Pei, and M. Kamber, *Data Mining: Concepts and Techniques*. Amsterdam, The Netherlands: Elsevier, 2011.
[41] Y. Sasaki *et al.*, "The truth of the f-measure," *Teach Tutor Mater*, pp. 1–5, 2007.
[42] C. E. Metz, "Basic principles of ROC analysis," *Seminars Nucl. Med.*, vol. 8, no. 4, pp. 283–298, 1978.
[43] J. W. Perry, K. Allen, and M. M. Berry, "Machine literature searching X. Machine language; factors underlying its design and development," *Amer. Documentation (Pre-1986)*, vol. 6, no. 4, p. 242, 1955.

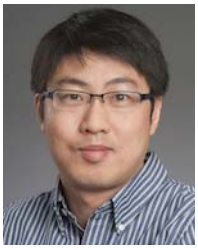
Aparna Sankaran Srinath received the master's degree in computer science from Boise State University, Boise, ID, USA, in December 2019, where she is currently pursuing the Ph.D. degree with the Computer Science Department, specializing in cybersecurity.

Her research interests include lightweight cryptography, cryptography applications in machine learning, social media, and data mining.

Hannah Johnson is currently pursuing the bachelor's degree in computer science with Boise State University, Boise, ID, USA.

While studying at Boise State University, she was a member of the Information Security, Privacy and Mining (ISPM) research lab. Her work focused on social media data mining and designing a social network for real-life cybersecurity applications. She currently works for Fast Enterprises as a Conversion Team Member.

Gaby G. Dagher is an Assistant Professor of computer science at Boise State University, Boise, ID, USA. He is the Director of the Information Security, Privacy & Mining (ISPM) research lab, and the Associate Director of the Idaho Election Cybersecurity Center (INSURE). His work focuses on designing secure protocols and developing tools for solving real-life problems related to cybersecurity and applied cryptography, including blockchain and cryptocurrencies, cyberforensics, and cloud computing.



Min Long received the Ph.D. degree from Cornell University, in 2008.

He is an Assistant Professor with the Department of Computer Science, Boise State University, Boise, ID, USA. He has extensive experience in scientific computing, numerical methods, data structures and algorithms, data analytics, and high-performance computing. He is a DOE Visiting Faculty with host scientists across multiple projects at Idaho National Laboratory, Idaho Falls, ID, USA. He later worked as a Post-Doctoral Fellow in computational science with the University of Illinois, Champaign, IL, USA, and The University of Chicago, Chicago, IL, USA, before joining Boise State University. His primary interest is in application of computer science and computational science to fundamental sciences, such as physics and materials science.