RESEARCH-ARTICLE

A Compression-Based Approach to Detecting Automated and Coordinated Behavior on Social Media

EDOARDO LORU, Sapienza University of Rome, Rome, RM, Italy

NICCOLÒ DI MARCO, University of Tuscia Viterbo, Viterbo, VT, Italy

MATTEO CINELLI, Sapienza University of Rome, Rome, RM, Italy

WALTER QUATTROCIOCCHI, Sapienza University of Rome, Rome, RM, Italy

Open Access Support provided by:

University of Tuscia Viterbo

Sapienza University of Rome

Published: 13 January 2026
Online AM: 30 November 2025
Accepted: 20 November 2025
Revised: 21 October 2025
Received: 14 January 2025

A Compression-Based Approach to Detecting Automated and Coordinated Behavior on Social Media

EDOARDO LORU, Department of Computer, Control and Management Engineering, University of Rome La Sapienza, Rome, Italy
NICCOLÒ DI MARCO, Department of Legal, Social, and Educational Sciences, Tuscia University, Viterbo, Italy
MATTEO CINELLI and WALTER QUATTROCIOCCHI, Department of Computer Science, University of Rome La Sapienza, Rome, Italy

Social media platforms are frequently targeted by entities engaging in automated or coordinated behavior, aiming to manipulate public opinion or conduct information operations without revealing their synthetic or managed nature. Research on detecting such actors faces the challenge of developing scalable, versatile methods that allow for consistent comparisons across diverse datasets. The challenge is even made more pressing by evidence of these actors on platforms beyond the extensively studied X (formerly Twitter), as well as the emergence of new platforms. We fill this gap by introducing a novel compression-based detection methodology, in addition to a new sparse method for network reconstruction that scales linearly under reasonable parameter choice. Being independent of the social media platform and the behavioral trace under study, our approach marks a departure from traditional methods that rely on multiple criteria or measures to assess user similarity. We evaluate our technique on multiple benchmark and real-world datasets, including widely known datasets related to political campaigns and emerging misinformation scenarios. We show that our approach provides a flexible unsupervised framework that effectively identifies both automated and coordinated activities across various behavioral traces, ensuring broad applicability.

CCS Concepts: • **Information systems** → **Social networks**; • **Mathematics of computing** → *Graph theory*; • **Computing methodologies** → Machine learning;

Additional Key Words and Phrases: Social Media, Coordinated Behavior, Bot Detection

Associate Editor: Yu Yang

The work is supported by IRIS Infodemic Coalition (UK government, grant no. SCH-00001-3391), SERICS (PE00000014) under the NRRP MUR program funded by the European Union—NextGenerationEU, project CRESP from the Italian Ministry of Health under the program CCM 2022, PON project "Ricerca e Innovazione" 2014-2020, and PRIN Project MUSMA for Italian Ministry of University and Research (MUR) through the PRIN 2022. This work was supported by the PRIN 2022 "MUSMA"—CUP G53D23002930006—Funded by EU—Next-Generation EU—M4 C2 I1.1.
Authors' Contact Information: Edoardo Loru (corresponding author), Department of Computer, Control and Management Engineering, University of Rome La Sapienza, Rome, Italy; e-mail: edoardo.loru@uniroma1.it; Niccolò Di Marco, Department of Legal, Social, and Educational Sciences, Tuscia University, Viterbo, Italy; e-mail: niccolo.dimarco@unitus.it; Matteo Cinelli, Department of Computer Science, University of Rome La Sapienza, Rome, Italy; e-mail: matteo.cinelli@uniroma1.it; Walter Quattrociocchi, Department of Computer Science, University of Rome La Sapienza, Rome, Italy; e-mail: walter.quattrociocchi@uniroma1.it.

ACM 1556-472X/2026/1-ART24
https://doi.org/10.1145/3778356

ACM Reference format:
Edoardo Loru, Niccolò Di Marco, Matteo Cinelli, and Walter Quattrociocchi. 2026. A Compression-Based Approach to Detecting Automated and Coordinated Behavior on Social Media. *ACM Trans. Knowl. Discov. Data.* 20, 2, Article 24 (January 2026), 25 pages.
https://doi.org/10.1145/3778356

1 Introduction

Social media platforms have reshaped the landscape of information exchange and public engagement [18]. Initially conceived for personal interaction and entertainment, these platforms now serve as critical arenas for disseminating news and discussing societal issues [5, 29, 57, 82, 96]. Consequently, substantial research has focused on their potential influence on discourse and collective dynamics. While some studies suggest a minimal impact on user attitudes [2, 5, 31, 42, 75], a significant body of work highlights issues such as misinformation spreading [10, 27], polarization [6, 93], and the presence of hateful content [60]. In this context, malicious actors may attempt to manipulate public opinion [35, 80, 100], provoke specific social responses [56, 87, 92], and orchestrate **Information Operations (IOs)** to amplify precise agendas [51, 99], thereby raising concerns about their potential role in the spreading of misinformation and propaganda [16, 28, 45, 86, 95].

Acknowledging the pressing issues posed by social media manipulation, researchers have developed a range of methods to detect social bots [35, 65, 88] and coordinated inauthentic behaviors [80, 86, 99]. These methods have evolved from simple classifiers that analyze individual accounts' features to elaborate techniques capable of encoding complex behavioral patterns and user activity. Despite these advancements, the increasing sophistication of automated accounts [22, 24] and the multifaceted nature of coordinated behavior online [76, 89] continue to pose considerable challenges. In response, this work introduces a novel compression-based method to detect and characterize suspicious behavior, including automated and coordinated activity.

Building upon prior research on the detection of suspicious behavior on social media platforms, this study introduces a novel technique that begins with encoding user activity into strings of text. These strings represent user actions based on arbitrary behavioral traces, such as co-retweeting or textual content, similar to methodologies in previous studies [21, 39, 67]. Inspired by its effectiveness in text classification [50], we measure user similarity using the **Normalized Compression Distance (NCD)** [55], a metric that quantifies the information distance between two compressed strings, to construct a similarity network of users. While previous applications of compression to assess user behavior have focused primarily on analyzing individual accounts [54, 67], our method extends its utility to group-level analyses, providing a more comprehensive perspective on coordinated actions. To mitigate the computational demands of calculating pairwise user similarity in large datasets, we introduce a novel heuristic that significantly reduces the number of calculations needed to construct an approximate similarity network. Our experimental results confirm that this approach not only enhances computational efficiency but also effectively preserves the underlying community structure of the data.

Our novel approach enables the identification of clusters of highly similar users with a uniform procedure across all kinds of behavioral traces, thus enhancing its flexibility and applicability to different social media platforms. In fact, although X (formerly Twitter) is historically the most analyzed platform, the presence of automated or coordinated behavior across other mainstream platforms such as Instagram [1], Reddit [48], and YouTube [68], as well as the emergence of new platforms like Koo [64] and Bluesky [78], highlights the need for versatile and platform-agnostic methods. Additionally, being simply based on a compression algorithm, our approach does not need to rely on a language model to evaluate text similarity, making it also suitable for non-English

documents (e.g., tweets or hashtags) or even low-resource languages [50]. We assess the validity of our technique using well-known benchmark datasets and real-world case studies, to confirm its effectiveness across different contexts and its ability to deliver reliable quantitative performance when ground-truth labels are available.

2 Related Works

Research into suspicious behavior on social media platforms primarily focuses on characterizing users on the extent of their automation and coordination.

The activity of social bots has been documented since the rise in popularity of online social networks. Despite lacking a single precise definition of what a social bot is [19], scholars generally refer to these accounts as completely or partially automated users (the latter also known as *cyborgs* [12]) who produce content and interact with legitimate accounts while disguising as humans [35, 44]. Early approaches relied on the assumption that humans and social bots display significantly different behavioral patterns and account features [19], hence they typically focused on the study of individual accounts and the detection of spammers and content polluters [12, 23, 54, 91, 101], suspicious content re-sharers [22, 37, 62], or fake followers [20, 49, 101]. Notably, Botometer (previously BotOrNot) [25] is a classifier made publicly available in 2014 that yields a *bot score* in $[0, 1]$ and has been widely used in research to assess the probability of automated behavior in users [34, 52, 53, 85]. However, concerns have been raised regarding the limitations of supervised approaches and automatic detection [7, 19]. These concerns mainly come down to the absence of a real ground truth that can be used to train and evaluate these classifiers [19], human annotators failing to reliably identify sophisticated bots [22, 53], and potentially misleading outputs [34, 81]. Further, it has been shown that social bots can "evolve" to evade detection systems [24, 102], with the risk of rendering annotated datasets outdated.

Despite these limitations, advances in Graph Neural Networks and other Deep Learning architectures have led to notable results that should not be overlooked. Among these, BotRGCN [33] employs relational graph convolutional networks on heterogeneous user graphs to learn embeddings that expose deceptive bots mimicking human-like behavior. BotDCGC [97] introduces a deep contrastive graph clustering framework that learns node embeddings through attention mechanisms and uses contrastive objectives to uncover bot communities without labels. RoSGAS [103] integrates reinforcement learning with self-supervised graph embedding to optimize model architectures for each user's ego-network, improving the detection of coordinated botnets. Dynamic methods such as BotDGT [43] leverage dynamic graph transformers to learn time-evolving node embeddings, which are particularly effective for identifying bots exhibiting shifting behavior patterns. Recent work has proposed a framework that combines heterogeneous graph attention with similarity-based features to improve bot detection by jointly learning from multi-relational structures and similarity-enhanced node representations [47].

Following studies that have exposed the coordinated activity of (not necessarily automated) users engaging in political astroturfing or IOs [51, 83, 86], a significant effort has been devolved to the development of techniques that can be applied "in the wild" to study group-level suspicious behavior. Among the proposed methodologies, network-based approaches are the most commonly found. These approaches usually rely on building networks of users where connections may indicate suspicious similarity, to be evaluated across different behavioral traces treated separately or jointly. Traces commonly found in the literature include the kind of actions a user performs [74], the targets of such actions [49, 51, 72, 83], unusual temporal activity [70, 74, 76], the textual content of posts [4, 26], the URLs that are shared [38, 41], or the hashtags that are used [41, 76, 98]. Among recent notable studies, Luceri et al. [59] and Graham et al. [41] create similarity networks where multiple behavioral traces are simultaneously considered and the edges are weighted based on the similarity

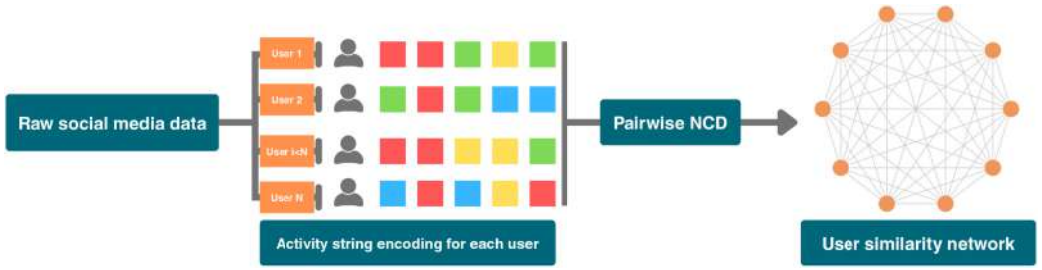

Fig. 1. *Pipeline of our proposed method.* Starting from social media data, we assign to each user a string of text. This string encodes the user's activity trace by concatenating their actions, according to arbitrary rules. For instance, it can be created by concatenating their reshared post or account IDs, shared URLs, or posts' textual content. Next, we measure the Normalized Compression Distance $\mathrm{NCD}(x_1, x_2)$ between each pair of user strings (x_1, x_2). Finally, we create a user similarity network, where the edge connecting two users is given a weight equal to $1 - \mathrm{NCD}(x_1, x_2)$. The network is subsequently analyzed to identify suspiciously high similarities, indicative of automated or coordinated behavior.

between users, whereas Nwala et al. [74] propose an alphabet that can be used to encode user actions and characterize their behavior. Shifting from previous methods focusing on static network analyses, Tardelli et al. [90] provide a dynamic analysis of coordinated behavior by studying the evolution of coordinated communities over time, revealing different behavioral patterns across communities. To address the limitations of bot detection models related to their complexity and the data required to train them effectively, Ng and Carley [71] propose an ensemble method capable of detecting and analyzing bot activity across multiple social media platforms. Luceri et al. [58] present a method to detect influence campaigns based on prompting a Large Language Model, obtaining promising results. Minici et al. [66] propose an architecture for the detection of IOs that combine Graph Neural Networks with information derived from users' sharing activities, including the textual content they share.

Our proposed method advances existing research by offering a platform-agnostic, flexible, and unsupervised framework that effectively identifies both automated and coordinated activities. Unlike traditional methods, our proposal is not tied to a specific platform or behavioral trace. The procedure we outline can be easily adapted to investigate suspicious interactions, coordinated URL or hashtag sharing, spam and content pollution, or even unusual temporal activity. Further, because it does not rely on features that are characteristic of specific social media platforms, it can be as easily implemented to analyze user activity on niche or understudied platforms. The overall pipeline of our method is presented in Figure 1.

3 Methods

3.1 Data

We employ a multitude of datasets collected from two social media platforms, in different contexts, and within different time frames (see Table 1 for a complete breakdown).

UK2019. This dataset is a large collection of approximately 11 million tweets by 1 million distinct users collected in the month before the UK2019 general election (from 12 November 2019 to 12 December 2019). All tweets featuring at least one among a predefined set of election-related hashtags have been gathered, in addition to all tweets produced by the official accounts of the two main leaders and the two main parties, and all retweets and replies they have received. This dataset has already been used in previous research on coordinated behavior online [17, 45, 72, 90], and

Table 1. Data Breakdown

Dataset	Time frame (yyyy-mm-dd)	Users	Posts	Annotated	Source/Reference
X—UK2019	2019-11-12 to 2019-12-12	11,264,280	1,179,659	Partially	[72, 73]
X—Honduras IO	2019-09-10 to 2020-01-08	224,685	1,262,830	Yes	[15]
X—cresci-2015	2007-01-02 to 2013-06-06	5,148	2,827,757	Partially	[20]
X—cresci-rtbust-2019	2018-06-18 to 2018-07-01	47,700	5,121,131	Partially	[61, 62]
X—COP26	2021-06-01 to 2021-11-14	2,080,280	10,240,966	No	[30, 32]
YouTube US Top Videos	2022-10-22 to 2023-01-01	2,167,460	3,030,260	No	–

A description of each dataset is reported in Section 3.1.

both the collected tweet IDs and the detected coordinated communities have been made publicly accessible by the authors [73].

Honduras IO. Starting in 2018, the **Twitter Moderation Research Consortium (TMRC)**[1] maintained a publicly available archive of datasets of users involved in known state-backed IOs. While these data are an outstanding resource for studying the patterns of activity of these users, the lack of a set of control users makes it difficult to employ them effectively in the setting of a detection task. In this work, we use the data made available in [15], which enriches the original TMRC dataset about IO in Honduras with tweets by genuine users concerning the same topics as those of the IO. The data span 4 months (from 11 September 2019 to 8 January 2020) and include approximately 1.2 million total tweets.

cresci-2015. Introduced in [20], this dataset has been extensively used in the literature as a benchmark for social bot detection techniques. It is an aggregation of five different datasets featuring either genuine users or so-called *fake followers*, accounts that have been purposefully bought to inflate the number of followers of a given user. It includes 3,900 annotated accounts, with half being automated accounts, and almost 3 million tweets mostly posted between 2012 and 2013. The data are publicly available on the Bot Repository.[2]

cresci-rtbust-2019. Similarly to *cresci-2015*, this publicly available dataset [61] is also commonly utilized as a benchmark for bot detection methods. It includes approximately 5 million retweets by 48,000 users posted within a 2-week time frame (from 18 June 2018 to 1 July 2018) within the Italian X. A set of approximately 700 accounts has been manually annotated as either a bot or genuine user (approximately 51/49%), and these labels are available on the Bot Repository.[3]

COP26. This dataset amasses a large collection of tweets posted within the context of the COP26 debate on X and was collected using the "`cop26`" search query. It includes approximately 8 million English tweets posted by 1 million unique users between 1 June 2021 and 14 November 2021. Unlike the previously discussed datasets, this particular one does not include any ground-truth labels regarding the authenticity or automation extent of the accounts within. Hence, we treat it as a real-world case study. The list of collected tweet IDs has been made publicly available by its authors [30].

YouTube. We collect approximately 3 million comments by 2 million users between November and December 2022, posted below the most popular videos in the United States across all YouTube's categories (e.g., Music, Entertainment, Sports). Despite the limited time frame, the videos' popularity on the platform during this period offers a relevant real-world scenario. It highlights how users

[1] https://web.archive.org/web/20240913160929/https://transparency.x.com/en/reports/moderation-research.
[2] https://botometer.osome.iu.edu/bot-repository/datasets/cresci-2015/cresci-2015.csv.tar.gz.
[3] https://botometer.osome.iu.edu/bot-repository/datasets/cresci-rtbust-2019/cresci-rtbust-2019.tar.gz.

may engage in coordinated or repetitive behavior to artificially boost a video's visibility or interact with a large audience, potentially to mislead or even scam them.

3.2 Extracting User Features with Compression

Grounded in information theory, lossless compression algorithms are designed to encode data using fewer bits by eliminating the statistical redundancies in the original representation. An illustrative example of a lossless compressor is Run-length encoding, where consecutive identical values are stored as a single value alongside its number of consecutive occurrences. Inspired by this foundation, our study leverages a compression algorithm to detect suspicious behavior on social media. Unusual regularities often indicate coordination or automation, and compression techniques have been shown to effectively capture their repetitiveness [21, 39, 62]. Specifically, we employ the widely used *gzip* compression algorithm, which is particularly effective in revealing such patterns.

3.2.1 Behavioral Traces. In this section, we describe how to encode user activity as a string to obtain an *activity string*. The selection of the string generation method is arbitrary and uniquely follows from the research question being addressed or the dataset under study. As the method requires no training data, this choice can be made ahead of time to study a particular kind of user activity and can later be refined in light of the results.

Here, we provide examples for three generic kinds of commonly studied behavioral traces: similarity of interactions, text similarity, and time-based similarity. Encoding rules for other traces or other platforms can intuitively follow as variations on these.

Interactions. A user's interactions can be encoded as the concatenation of the IDs of users or posts it has interacted with. For instance, if we want to study the retweeting activity of a user u on X, we can represent it as a string A_u:

$$A_u = \sum_{i=1}^{N_u} \texttt{id}_i^u, \tag{1}$$

where $\texttt{id}_i^u$ is the ID of the i-th (timestamp-wise) retweeted tweet and the sum operator $\sum$ indicates the concatenation of strings, with N_u being the number of retweets by the user within the time frame under study. Generalizing to other interactions (e.g., replies or quotes), we can also take into account the type of interaction that has occurred between user u and the authors of tweets $i = 1, \ldots, N_u$ by modifying Equation (1) as:

$$A_u = \sum_{i=1}^{N_u} (\texttt{interaction\_kind}_i^u + \texttt{id}_i^u), \tag{2}$$

where $\texttt{id}_i^u$ and $\texttt{interaction\_kind}_i^u \in \{\texttt{retweet}, \texttt{quote}, \texttt{reply}\}$ are, respectively, the ID of the i-th (timestamp-wise) tweet and the type of interaction user u had. Conversely, if we want to encode the targeted user (e.g., retweeted) rather than the kind of interaction, we can modify Equation (2) as:

$$A_u = \sum_{i=1}^{N_u} (\texttt{user\_id}_i^u + \texttt{id}_i^u), \tag{3}$$

using an analogous notation. We also underscore that Equations (1)–(3) can easily be combined to create a single activity string that includes all three items (post ID, author ID, and type of interaction).

Even if other social media platforms may not feature the same types of interactions as X's, the string encoding process will be similar. For instance, to investigate users repeatedly commenting YouTube videos in a seemingly automated or coordinated manner, one could build user activity strings made of the concatenations of the targeted video or channel IDs. Conversely, to investigate

content boosting via liking/upvoting of posts, each user's activity strings may be obtained by concatenating the IDs of the liked posts or post owners.

Since the user or post IDs and the interaction kind prefixes are represented as strings of different lengths, we can compute their MD5 hashes to standardize their length. We specifically choose the MD5 hashing algorithm because it is widely known and efficient. However, any alternative that serves the same purpose would likely produce similar results. Finally, by concatenating the resulting hashes, we ensure that their original string lengths do not affect the information distance between two activity strings.

Text. Analogously to previous works [54], evaluating the similarity in document production from the standpoint of textual content can be simply implemented by concatenating all posts (e.g., tweets or YouTube comments) by a user:

$$A_u = \sum_{i=1}^{N_u} \texttt{post\_text}_i^u, \tag{4}$$

where $\texttt{post\_text}_i^u$ is the textual content of the i-th (timestamp-wise) post and N_u is the number of posts by the user within the time frame under study. We underline that this approach allows text similarity to be evaluated regardless of the document's language, as shown in [50]. This is especially relevant for low-resource languages.

Additionally, while Equation (4) shows how the history of posts by two users can be compared, this is only an example of how this kind of similarity can be applied. For instance, the same generic procedure can be implemented to investigate coordinated URL or hashtag sharing, or user targeting via mentions. This can be achieved by simply concatenating those elements, rather than the whole text. Further, rather than user activity, this approach can also be employed to evaluate the similarity of other text documents, such as user profile descriptions on X or YouTube.

Time-Based. Multiple criteria have been used to capture the similarity between users in terms of their activity over time [11, 62, 74]. In this work, we focus on the time interval between two successive actions by the same user and evaluate the similarity between users from this particular standpoint. This is motivated by previous works showing that producing or sharing content in fast succession or with characteristic patterns can be a sign of suspicious behavior [62]. As our method requires the behavioral traces to be encoded as strings, the time interval between actions must first be binned to obtain a finite number of symbols to use. To this end, one could either arbitrarily define a set of bins, or partition the time delay distribution according to its quantiles or by using methods such as that developed by [40]. Then, similar to the encoding rules presented earlier, we concatenate in chronological order all binned time delays for each user into a single string:

$$A_u = \sum_{i=1}^{N_u} \texttt{delay\_bin}_i^u, \tag{5}$$

where $\texttt{delay\_bin}_i^u \in \{1, \ldots, N_b\}$ is the i-th chronologically ordered interval between two successive actions, in case of N_b bins. We note that the actual characters used to encode the bins have no impact on the results as long as their length in bytes is the same, meaning that by using an ordered sequence of characters as symbols we are not weighing short intervals any differently than long intervals. However, using an encoding such as that of the example above can aid in interpreting the results coming from our procedure.

Other time-based traces, such as the time between a retweet and its referenced tweet, or between the date of upload of a YouTube video and the timestamp of one of its comments, can be implemented similarly.

3.3 Constructing the Similarity Network

3.3.1 User Similarity. After creating a user's activity string with the methods detailed in the previous section, we compare it to those of other users using the NCD measure. This approach is inspired by prior work [50], which shows that the NCD, when paired with the gzip compression algorithm, is highly effective for classifying textual documents into distinct categories based solely on their pairwise distances. Similarly, in our framework, we intend to classify users based on their online behavior. This can be captured by the NCD among users' activity strings, which are, fundamentally, textual documents.

The NCD is a function that takes in input two strings x_1 and x_2 and outputs their information distance (i.e., degree of dissimilarity):

$$\mathrm{NCD}(x_1, x_2) = \frac{C(x_1x_2) - \min\{C(x_1), C(x_2)\}}{\max\{C(x_1), C(x_2)\}}, \tag{6}$$

where $C(\cdot)$ is the compressed length of the string, and x_1x_2 is the concatenation of the strings. Despite its intuitive definition, two relevant considerations must be made. First, while it is referred to as *distance*, the NCD cannot be formally considered as such because it is not symmetric ($\mathrm{NCD}(x_1, x_2) \neq \mathrm{NCD}(x_2, x_1)$). However, the difference is typically small enough to be negligible [13] and could be addressed by replacing $C(x_1x_2)$ in Equation (6) with $(C(x_1x_2) + C(x_2x_1))/2$, although this would incur additional computational cost. Second, its output is not guaranteed to be normalized in $[0, 1]$, as there exists some x_1 and x_2 such that $\mathrm{NCD}(x_1, x_2) \gtrapprox 1$; in such cases, we round down the NCD to 1. Computing all pairwise distances yields a weighted adjacency matrix that can be used to build a similarity network whose nodes are users. Following the previously discussed considerations about the NCD, to build the network we only consider the upper (or lower) triangular part of the matrix and we assign to each edge (u, v) a weight $w_{uv} = 1 - \mathrm{NCD}(u, v)$. In the case of $\mathrm{NCD}(u, v) = 1$, we set $w_{uv} = 0.001$, to retain the edge (u, v) while simultaneously minimizing its weight.

3.3.2 Node Pruning. As some of the datasets we employ do not include the accounts' ground-truth labels, in such cases we filter the similarity network by only keeping nodes that show substantial similarity with at least another one, motivated by previous research showing the effectiveness of node pruning [59].

Analogously to the coordination detection algorithm proposed by [72], we assign to each user a score in $[0, 1]$ that quantifies its extent of similarity with its neighbors. In detail, the score of node v is obtained by computing the percentile rank of weight $\max_{u \in N(v)} W[u, v]$ with respect to the edge weight distribution of the similarity network. Subsequently, we remove from the network all nodes below a certain score and focus our analysis on the rest.

We note that, while more sophisticated filtering techniques exist, such as backbone extraction algorithms [84], these are usually intended for multiscale weighted networks, rather than similarity networks with weights in $(0, 1]$.

3.4 Similarity Network Approximation

3.4.1 Algorithm. The construction of similarity networks of N users requires $O(N^2 \cdot C(N))$ computations, where $C(N)$ represents the computational cost of the similarity measure. Consequently, this process can incur substantial computational costs, particularly when dealing with a large user base. In this section, we present a heuristic that can be used to manage the required number of computations, while simultaneously retaining the potential underlying community structure of the network. Following the intuition that computing all pairwise distances between N users can be viewed as traveling along all edges of a complete graph with N nodes, our method

Algorithm 1: Similarity Network Approximation

Input: V : set of nodes (i.e., activity strings)
η : number of nodes sampled at each step
μ : number of times the procedure is repeated
Output: $G(V, E, W)$: Undirected weighted similarity network
Let $\epsilon > 0$; // minimum possible edge weight
$\{x, y\} \leftarrow \text{RandomSample}(V, n = 2)$;
$E \leftarrow \{(x, y)\}$;
$W[x, y], W[y, x] \leftarrow \max\{\epsilon, 1 - \text{NCD}(x, y)\}$;
while $\mu \geq 1$ **do**
  for $u \in V$ **do**
    if $u \in \{x, y\}$ ***and*** $\mu = 1$ **then**
      skip iteration;
    $V_c \leftarrow \{v \in V \setminus \{u\} : \text{degree}(v) > 0\}$;
    $V_s \leftarrow \text{RandomSample}(V_c, n = \min\{\eta, |V_c|\})$;
    $v_s \leftarrow \text{RandomSample}(\arg\min_{v \in V_s} \text{NCD}(u, v), n = 1)$;
    $V'_s \leftarrow N[v_s] \setminus N(u)$;
    if $V'_s \neq \emptyset$ **then**
      $V'_s \leftarrow \text{RandomSample}(V'_s, n = \min\{\eta, |V'_s|\})$;
      $v \leftarrow \text{RandomSample}(\arg\min_{v' \in V'_s} \text{NCD}(u, v'), n = 1)$;
      $E \leftarrow E \cup \{(u, v)\}$;
      $W[u, v], W[v, u] \leftarrow \max\{\epsilon, 1 - \text{NCD}(u, v)\}$;
  $\mu \leftarrow \mu - 1$;

greedily constructs the similarity network in a way similar to approximate nearest neighbor search techniques [3]. The algorithm, whose pseudocode is provided in Algorithm 1, requires setting two parameters: $\eta \in \{1, \ldots, N\}$ and $\mu \geq 1$.

The procedure can be summarized as follows: We first connect a randomly chosen pair of nodes x, y and compute their similarity $1 - NCD(x, y)$. Then, for each other node u, we randomly select η nodes already belonging to the connected component (i.e., having degree greater than 0) and we select the node v_s for which $\text{NCD}(u, v_s)$ is minimum. Then, we select η random nodes from $V'_s = N[v_s] \setminus N(u)$, where $N[\cdot]$ and $N(\cdot)$ denote the closed and open neighborhood of a node, respectively, and add an edge between u and $v \in V'_s$ such that $\text{NCD}(u, v)$ is minimized. These steps are applied, for each node, μ times. Therefore, the η parameter can be interpreted as a quantity proportional to the likelihood of sampling the node showing the highest similarity to the target. In contrast, the μ parameter can be used to build denser graphs, which can aid in subsequent network-based analyses. By memorizing all the computed similarities without further re-computation, Algorithm 1 needs $O(\eta\mu N \cdot C(N))$ steps to terminate and give the (connected) similarity network. Therefore, by caching all intermediate similarities computed during its execution, our approach allows tuning η and μ ($\ll N$) to reduce the original $O(N^2 \cdot C(N))$ complexity and, at the same time, it promises to maintain the community structure of the underlying complete network.

3.4.2 Robustness. Algorithm 1 is particularly suitable for studying user activity online, as the focus of these analyses does not typically entail an assessment of the similarity of each user to all others, but rather the identification and characterization of groups of users who share unusually similar behavior. To test its performance in this task, we derive a toy dataset of 1,074 users out of the *UK2019* X dataset, by randomly sampling 10% of users from each of the coordinated communities identified by the authors. As the communities are imbalanced in size, with such sampling we can reduce the number of users in the network while retaining the community structure of the full network. Using the approach detailed in Section 3.2, we obtain for each user an activity string of interactions as defined in Equation (2), build the complete similarity network, and run the Louvain

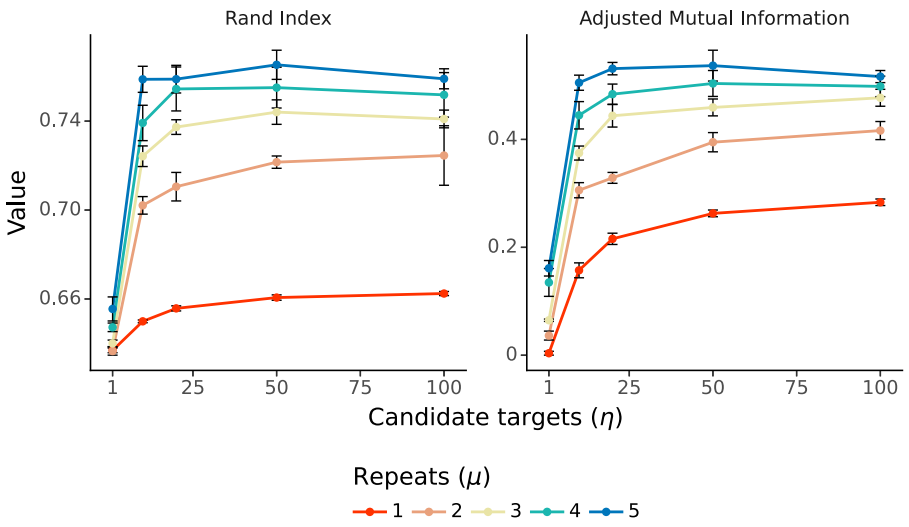

Fig. 2. *Clustering agreement of the approximated network with the complete similarity network of our toy dataset of 1,074 users.* For each pair of η and μ, we compute the average Rand Index and Adjusted Mutual Information between the complete network's clusters and those detected on five approximated similarity networks. The error bars correspond to the sample standard deviation.

clustering algorithm with resolution $\gamma = 1$. For each pair of parameters $\eta \in \{1, 10, 20, 50, 100\}$ and $\mu \in \{1, 2, 3, 4, 5\}$, we generate $k = 5$ similarity networks with Algorithm 1, and like for the complete network, we perform on each one community detection with Louvain and $\gamma = 1$. To compare the resulting communities for each pair (η, μ), we measure the Rand Index [79] and the **Adjusted Mutual Information (AMI)** [94] between the clustering obtained for the complete network and that of each of the $k = 5$ generated networks, and average the resulting values. The results, which we report in Figure 2, show that by randomly sampling even a small number of candidate targets ($\eta = 25$ corresponds to $\approx 3\%$ of the total number of users), we can achieve a promising agreement with the clustering on the complete network. Additionally, we can observe that increasing the number of times μ the algorithm is repeated leads to larger values of the two measures, as denser networks better approximate the complete case. These results indicate that running the algorithm with computationally affordable parameters yields results that can already provide a good indication of any underlying community structure in the similarity network. However, we also note that the exact values of the Rand Index and the AMI depicted in Figure 2 are bound to change when different networks or community detection algorithms are employed, meaning that they should not be taken as benchmarks. Further discussion on the performance of the heuristic is provided in the Appendix A, including an analysis of its running time compared to building the complete similarity network.

While the applications that we showcase in this work are confined to similarity networks of social media users based on their activity, and to use the NCD between two activity strings as a dissimilarity measure, we underscore that the algorithm can easily be generalized to different metrics and applications.

3.5 Node Embeddings

Node embeddings are commonly used to represent networks graphically or to derive feature vectors that can be utilized to train Machine Learning models for node classification tasks. For instance, they can be derived by computing the spectral decomposition of the Laplacian matrix of the network [8].

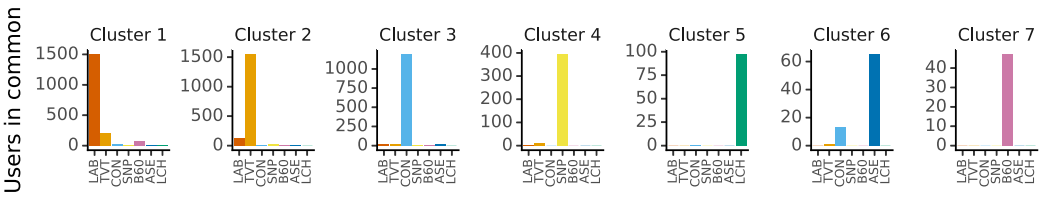

Fig. 3. *Clustering agreement with* [72] *for the UK2019 dataset.* Each panel corresponds to the best-matching cluster on our similarity network with the clusters previously identified by [72] (on the x-axis, the name assigned by the authors of the dataset), numbered according to the number of users within (largest to smallest among these). In Appendix Figure A3, we report a heatmap showing the distribution for all clusters detected on our similarity network.

The symmetrically normalized Laplacian matrix $\mathbf{L}$ of a network with $N = |V|$ nodes is defined as:

$$\mathbf{L} = \mathbf{I} - \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}, \tag{7}$$

where $\mathbf{I}$ is the identity matrix of size N, $\mathbf{D}$ is the diagonal strength matrix with elements the sum of incident edge weights $D_{ii} = \sum_{j \in N(i)} w_{ij}$, and $\mathbf{A}$ is the adjacency matrix. First, we compute the eigendecomposition of $\mathbf{L}$, obtaining a $N \times N$ matrix of N eigenvectors $\mathbf{v}_i$, each associated with an eigenvalue λ_i. Then, we select the k smallest non-null eigenvalues and the corresponding eigenvectors, which results in a $N \times k$ matrix $\mathbf{E}$ whose rows are $1 \times k$ vectors associated with each node. We associate to node i, the ith row of $\mathbf{E}$, normalized by its Euclidean norm. As a result, these embeddings can be used for node classification tasks or with dimensionality reduction techniques such as UMAP for visualization.

4 Results

In this section, we report the performance of our technique in detecting and clustering users based on the behavioral traces described in Section 3.2. Additional evaluations of our method's performance, including a comparison against existing baselines, are provided in the Appendix A.

For community detection, we use either the Louvain algorithm [9] or **Stochastic Block Modeling (SBM)** [77]: the former for well-known annotated datasets and the latter for real-world case studies. This choice allows performing consistent comparisons with prior works, which often relied on the Louvain algorithm despite its limitations [36], in contrast to the increased robustness that SBM can provide when ground-truth labels are not available. To further assess the resulting networks, we also generate node embeddings as detailed in Section 3.5 for visualization and classification purposes.

4.1 Similarity of Interactions

We evaluate the performance of our method in identifying users with similar interaction patterns on three different datasets: *UK2019* [73], *Honduras IO* [15], and *cresci-rtbust-2019* [61].

4.1.1 UK2019. We focus on the 10,782 "superspreaders" (i.e., the top 1% retweeters) considered by the authors of this dataset [72] and apply our method to assess whether the clusters we detect on our similarity network coincide with the seven communities originally found. To this end, we build the similarity network as described in Section 3.4, using Equation (2) to obtain the activity string for each user. In detail, we construct the network using parameters $\eta = 500$ and $\mu = 2$ and then apply the Louvain clustering algorithm with $\gamma = 1$ to identify communities. In Figure 3, we show that each of the original communities has a large number of users in common with at least one of those detected with our method. The overlap with all clusters identified in our similarity network is available in Appendix Figure A3.

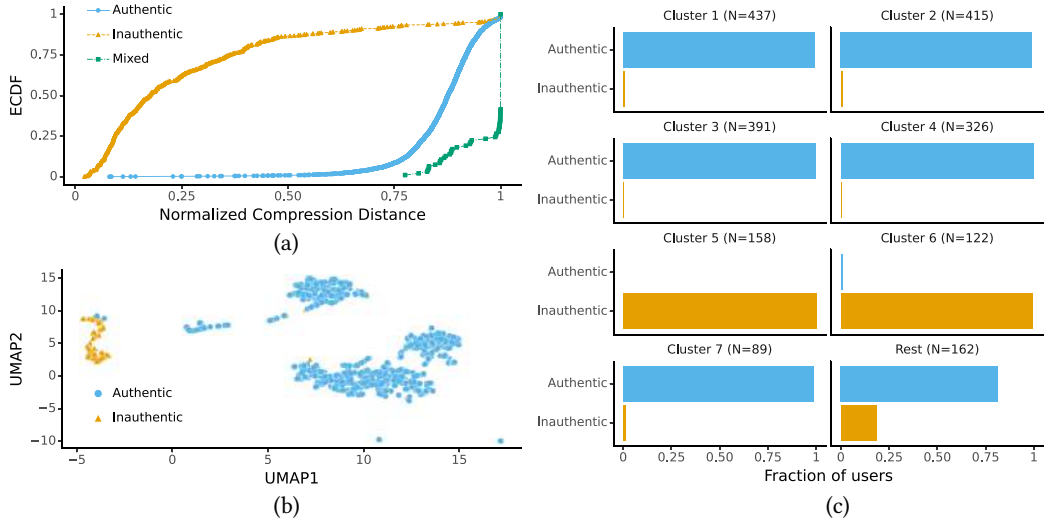

Fig. 4. *Similarity network based on co-retweeting in the Honduras IO dataset.* (a) Empirical Cumulative Distribution Function of the NCD between nodes. (b) UMAP projection of 128-dimensional Laplacian embeddings. (c) Distributions of authentic and inauthentic users in the clusters identified via SBM, with "Rest" including all nodes not in the first seven largest clusters.

4.1.2 Honduras IO. Similarly to *UK2019*, we focus on the top 1% retweeters ($N = 2{,}100$) and their retweeting activity, inspired by previous works showing the salience of co-retweeting in coordination detection [51, 59, 72]. Notably, prior work has specifically used the 1% threshold to identify users potentially engaging in coordinated retweeting [72, 89, 90]. Therefore, we use Equation (1) to construct an activity string for each user and then build the network with $\eta = 100$ and $\mu = 2$.

Figure 4 highlights the effectiveness of our method in detecting users displaying authentic and inauthentic behavior (i.e., organic users and IO drivers, respectively). In detail, Figure 4(a) shows that inauthentic accounts are generally characterized by much smaller NCD (i.e., large edge weights), indicating highly similar retweeting activity. This difference between users is further highlighted in Figure 4(b), which we obtain by projecting with UMAP [63] onto a 2D plane the Laplacian embeddings of the similarity network, where inauthentic accounts appear well separated from the genuine ones, despite the latter being themselves organized in further sub-communities of similar users. In Figure 4(c), we report the fraction of authentic and inauthentic accounts in each cluster detected via SBM, obtaining well-separated communities of authentic or inauthentic users. We note that in this case, we employ SBM rather than Louvain because further tuning of its resolution parameter would be required to obtain similar results, which is a known limitation of modularity optimization [36].

Because this dataset includes ground-truth labels regarding accounts' authenticity, we can evaluate the performance of our method at classifying genuine and inauthentic accounts. Similarly to previous works [76], we can employ an unsupervised approach that is based on edge filtering and classifies as "inauthentic" all users with at least one edge e such that $NCD_e < t$, where t is a fixed threshold. To find the best-performing threshold, we evaluate the classification performance over a range of values of $t \in [0.1, 0.9]$, obtaining with $t = 0.3$ a promising F1-score of 0.87 despite the naivety of this classification criterion. We note that using any threshold in $[0.2, 0.6]$ yields F1-score > 0.82 (see Appendix Figure A2), indicating that this result is not overly sensitive to the

specific threshold chosen. We can also evaluate the classification performance in a supervised manner, by using the 128-dimensional node embeddings displayed with UMAP in Figure 4(b) as feature vectors to train a Random Forest model, similarly to [59]. We stress that with this approach we are not aiming to build a general classification model that can reliably discern authentic and inauthentic users. Rather, we are showing that node embeddings derived from our similarity network may also be employed with efficacy as feature vectors to classify users in a supervised framework. By using 10-fold cross-validation, we evaluate this approach and obtain an average F1-score of 0.968 (with standard deviation $\hat{\sigma} = 0.019$).

4.1.3 cresci-rtbust-2019. We also evaluate the effectiveness of our technique in *cresci-rtbust-2019*, an additional dataset used as a benchmark for bot detection tasks. Inspired by the study that introduced this dataset [62], we focus on retweeting activity and define each user's activity string A_u as the concatenation of the retweeted tweet IDs and the corresponding retweeted user ID, using Equation (3). Then, we construct the similarity network with parameters $\eta = 50$ and $\mu = 2$ and run the Louvain clustering algorithm with resolution $\gamma = 1$.

Figure 5 shows that our method can effectively separate bots and humans by their retweeting patterns, despite solely employing the IDs of the retweeted accounts and tweets. Similarly to the case of coordinated accounts observed in *Honduras IO*, Figure 5(a) highlights that automated accounts tend to be characterized by much higher retweeting similarity. Additionally, Figure 5(b) and (c) shows that bots and humans form communities that are overall well separated from each other, with Cluster 2 being a notable exception. Based on prior work [62] that modeled user activity to capture similarities in temporal patterns, this result suggests that bots and humans in this cluster may retweet similar sets of accounts or tweets, but exhibit different temporal activity. This observation also helps explain the lower clustering performance observed in comparison to Honduras IO, shown in Figure 4. In the case of Honduras IO, we capture behavior that we knew in advance to be suspicious (i.e., coordinated retweeting) based on the dataset's source. In contrast, here we focus on retweeting activity despite the original study's focus on temporal activity. As our method does not offer a straightforward approach to modeling temporal patterns, we discuss potential strategies to capture them within our framework in the Limitations section.

As in Section 4.1.2, we evaluate the classification performance in both an unsupervised and a supervised manner. For the former, we obtain a maximum F1-score of 0.84 using a threshold $t = 0.7$ (see Appendix Figure A2), which is comparable to that obtained with the method proposed by the authors of the dataset [62]. For the latter, we train a Random Forest model with the 128-dimensional Laplacian embeddings of the similarity network, obtaining an average F1-score of 0.83 ($\hat{\sigma} = 0.08$) with 10-fold cross-validation.

4.2 Text Similarity

In this section, we evaluate the similarity of users concerning the textual content of their posts, concatenated as in Equation (4). In detail, we focus on *cresci-2015* [20] and a *YouTube* dataset we collected ourselves (see Section 3.1 for details). For the former, as it includes *fake followers* that may have been programmed to produce certain predefined posts to appear more genuine, we expect text similarity to capture such manifestations. Similarly for the latter, text similarity may be able to capture accounts conducting scam campaigns on YouTube [68] or polluting popular comment sections to boost their visibility.

4.2.1 cresci-2015. For this bot detection task, we create the similarity network with parameters $\eta = 100$ and $\mu = 2$ and then run the Louvain clustering algorithm with $\gamma = 1$. Figure 6(a) shows a clear separation between automated and human accounts, which is further confirmed by the classification performance we achieve with the same unsupervised and supervised approaches

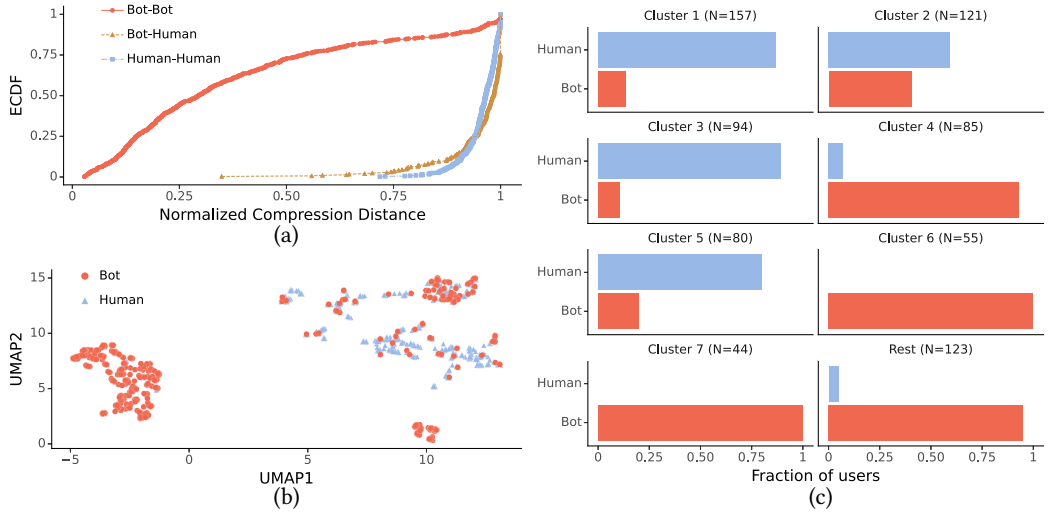

Fig. 5. *Similarity network based on user interactions in cresci-rtbust-2019.* (a) Empirical Cumulative Distribution Function of the NCD between nodes. (b) UMAP projection of 128-dimensional Laplacian embeddings. (c) Distributions of bots and humans in the clusters identified via the Louvain algorithm, with "Rest" including all nodes not in the first seven largest clusters.

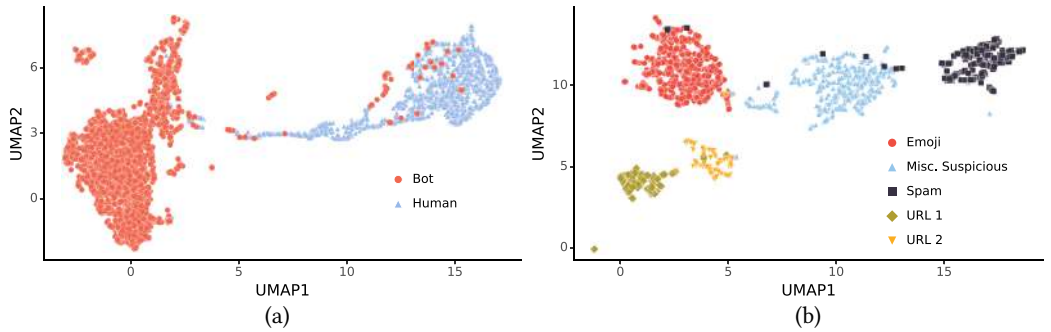

Fig. 6. *Similarity networks based on text written by users.* (a) Similarity of tweets by users in the cresci-2015 dataset. (b) Similarity of comments by users below top videos on YouTube, first filtered to only retain nodes with a score > 0.9 (see Section 3.3.2), and successively fitted with a hierarchical SBM to identify communities. For both networks, we project their 128-dimensional Laplacian embeddings on a 2D plane using UMAP.

employed in Section 4.1: for the former an F1-score of 0.89 with threshold $t = 0.95$ (see Appendix Figure A2), and for the latter an average F1-score of 0.982 ($\hat{\sigma} = 0.021$) with 10-fold cross-validation.

4.2.2 YouTube Top Videos. Applying our method to the YouTube dataset, which does not include ground-truth labels on the accounts' nature, reveals multiple communities of users who post highly similar comments. For this analysis, we start by focusing on the top 1% commenters. As they represent highly active users, they are ideal candidates for investigating content pollution. Then, we apply an additional threshold on the number of distinct commented videos, focusing on users who have commented on a number of distinct videos above the median. This ensures that we concentrate on users who exhibit this behavior across a variety of videos, rather than those who may be spamming multiple comments on a single video or a limited subset of videos. While we

believe that this particular set of users is the most indicative of potential coordinated activity, the specific threshold applied is arbitrary and can be adjusted based on the specific dataset or context being studied.

We create the similarity network using parameters $\eta = 500$ and $\mu = 2$ and then apply the method in Section 3.3.2 to only retain nodes with a score > 0.9. Finally, we use SBM to identify the five clusters whose embeddings are displayed with UMAP in Figure 6(b). Upon visual inspection, these clusters consist of users who can generally be considered content polluters [54]. For instance, the communities we name "URL 1" and "URL 2" both contain users who repeatedly spam other YouTube links (mostly of other videos), whereas the "Emoji" community is formed by users who exclusively post multiple messages of just emojis. The users in the "Spam" and "Misc. Suspicious" clusters do not exhibit as clear behavioral patterns as the rest, with the former mostly made up of users who write identical or highly similar low-quality comments below the same or multiple videos, and the latter also including scammers who promise gifts to those who contact them and users who advertise their channels. These results show that our procedure can be reliably applied "in the wild" to identify users posting suspiciously similar messages, regardless of their automation or coordination extent.

4.3 Time-Based Similarity

We now evaluate user similarity concerning temporal activity on the *COP26* dataset [30], focusing on the set of users ($N = 21{,}990$) obtained by the union of the top 1% retweeters and the top 1% of producers of original tweets, following the rationale outlined for the previous experiments. As we describe in Section 3.2.1, each user's activity string is obtained by first computing the time interval between two successive actions (regardless of the kind of action, e.g., a retweet or a reply) and by then binning these values according to the time interval distribution. Specifically, we consider the corresponding quantile function $Q(p)$ and values $\mathbf{p} = (0.25, 0.50, 0.75)$, and label as "1" all time intervals $\Delta t < Q(0.25)$, as "2" those in $Q(0.25) \leq \Delta t < Q(0.50)$, as "3" those in $Q(0.50) \leq \Delta t < Q(0.75)$, and as "4" the rest. Then, we build the similarity network using parameters $\eta = 100$ and $\mu = 2$, we filter it as described in Section 3.3.2 by removing nodes with a score ≤ 0.95, and we finally apply SBM for community detection. We underscore that high similarity in this examination does not necessarily imply coordination or alikeness of shared content: it is only an indication that the activity signals of some users show similar patterns of temporal gaps between successive posts.

In Figure 7(a) and (b), we display the two clusters resulting from our procedure and the respective time interval distributions of the users therein. While the time interval distribution for the users in Cluster 1 suggests that the similarity between these users might be non-suspicious, interestingly that of Cluster 2 shows that the majority of time intervals belong to the first quantile (i.e., approximately 90 seconds or shorter). Upon manual inspection of Cluster 2 ($N = 371$), we do in fact identify actors who target certain accounts (using mentions or retweets) or hashtags with repeated and repetitive posting, indicating suspicious behavior aimed at boosting the popularity of certain content or seeking the attention of certain accounts. These results show that our approach can also effectively handle and highlight the similarity between numerically encoded strings, such as the binned time intervals used in this case.

5 Limitations

Although it yields promising results across several datasets and contexts, our method still presents some limitations. First, while quantitatively evaluated on datasets with ground-truth labels, these datasets might be outdated or may have been collected or assembled in such a way that the differences between legitimate and suspicious users are more pronounced than "in the wild."

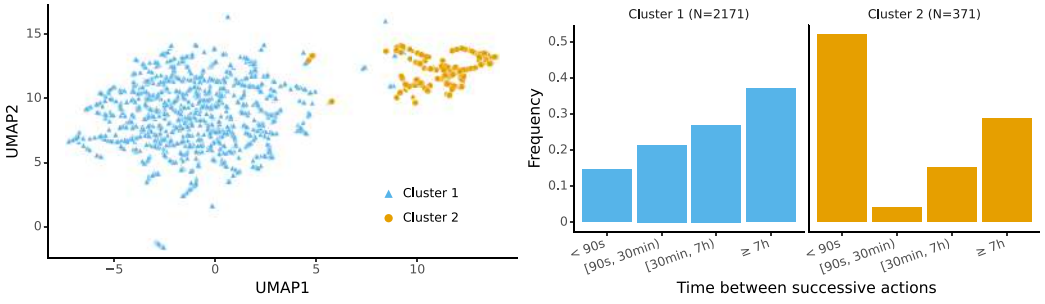

Fig. 7. *Similarity network based on the delay between successive actions in the COP26 dataset.* The network is filtered by only retaining nodes with a score > 0.95 (see Section 3.3.2); then, an SBM is fitted to the network to identify the two communities shown in the figure. (a) UMAP projection of the 128-dimensional Laplacian embeddings of the similarity network. (b) Frequency in the two clusters of the four time intervals between actions employed to encode user activity.

However, we have also been able to further check the goodness of our results on real-world datasets.

Second, while our method is designed to be platform-agnostic, we acknowledge that our analysis primarily focuses on X datasets and only features one additional platform, namely YouTube. With our dataset choice, we intended to strike a reasonable tradeoff between datasets with ground-truth labels and real-world case studies, while still providing a comprehensive overview of our procedure's output. Therefore, due to the scarcity of annotated datasets, in addition to the historical focus on X, most of the data employed comes from one platform. Nonetheless, as the activity concatenation procedure can be defined arbitrarily, we expect our method to easily adapt to other platforms and their specific interactions, too. For instance, this could include coordinated URL sharing on platforms such as Facebook or Instagram, or upvoting on Reddit or Koo. This is the main benefit of building a similarity network over an actual interaction network: it standardizes the procedure across different behavioral traces and social media platforms. This approach also enables cross-platform studies without requiring alterations to the procedure, an extremely relevant task that should be explored in future work.

Third, similar to previous works, our procedure relies on setting arbitrary values to construct the similarity network and a threshold to produce the final classification labels, meaning that manual inspection of the detected clusters of users is still advised for a better interpretation of the results. Future work should explore the development of methods that minimize reliance on ex-post interpretation, while still guaranteeing explainability and wide applicability.

Finally, we acknowledge that while our method offers flexibility, it may not allow a straightforward encoding of more complex behavioral patterns that may characterize specific influence campaigns. In such instances, implementing more sophisticated and ad-hoc techniques could provide a more effective detection strategy. For instance, capturing certain temporal patterns may require procedures that go beyond simple string concatenation. Although not explored in this study, one potential strategy could involve constructing separate time windows and applying our technique to each separately, similar to prior work [14, 90]. The resulting similarity networks could then be analyzed independently or aggregated before further analysis. This would allow adding a temporal constraint to the evaluated user similarity, rather than considering the entirety of a user's activity.

6 Conclusions

In this work, we present a novel framework that can be used to detect and characterize suspicious behavior on social media platforms by leveraging the *gzip* compression algorithm. Inspired by existing network-based approaches that connect and analyze users based on their behavioral similarity, our approach constitutes a shift from prior works by simplifying the encoding of user actions and their comparison with others. In fact, by encoding behavioral traces as simple strings of text that are compared with the NCD measure, our procedure does not need to rely on multiple similarity measures or criteria to capture each behavioral trace. This flexibility allows its application for different purposes, including social bot and coordination detection, and on different platforms, thus going beyond the historical focus on Twitter (now X). Additionally, to overcome the computational challenge of building similarity networks of a large number of users, we introduce a novel algorithm that can construct the network and retain its underlying community structure, while simultaneously being computationally affordable. Beyond showcasing the methodology, we prove its effectiveness on a multitude of datasets, chosen among well-known benchmarks commonly found in the literature and real-world case studies. The results show that our procedure can reliably identify suspicious groups of users—whether automated or coordinated accounts—across several behavioral traces and contexts, including IOs and more generic content pollution. These findings highlight the possibility of designing simple and versatile detection methods that are nonetheless effective, thus providing a viable alternative to overly complex or task-specific techniques. Future research on the detection of suspicious behavior should focus on developing flexible and explainable methods suitable for the continuously evolving social media landscape.

References

[1] Fatih Cagatay Akyon and M. Esat Kalfaoglu. 2019. Instagram fake and automated account detection. In *2019 Innovations in Intelligent Systems and Applications Conference (ASYU)*. IEEE, 1–7. DOI: https://doi.org/10.1109/asyu48272.2019.8946437

[2] Hunt Allcott, Matthew Gentzkow, Winter Mason, Arjun Wilkins, Pablo Barberá, Taylor Brown, Juan Carlos Cisneros, Adriana Crespo-Tenorio, Drew Dimmery, Deen Freelon, et al. 2024. The effects of facebook and instagram on the 2020 election: A deactivation experiment. *Proceedings of the National Academy of Sciences* 121, 21 (May 2024). DOI: https://doi.org/10.1073/pnas.2321584121

[3] Sunil Arya and David M. Mount. 1993. Approximate nearest neighbor queries in fixed dimensions. In *Proceedings of the 4th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA '93)*. Society for Industrial and Applied Mathematics, USA, 271–280.

[4] Dennis Assenmacher, Lena Clever, Janina Susanne Pohl, Heike Trautmann, and Christian Grimme. 2020. *A Two-Phase Framework for Detecting Manipulation Campaigns in Social Media*. Springer International Publishing, 201–214. DOI: https://doi.org/10.1007/978-3-030-49570-1_14

[5] Michele Avalle, Niccolò Di Marco, Gabriele Etta, Emanuele Sangiorgio, Shayan Alipour, Anita Bonetti, Lorenzo Alvisi, Antonio Scala, Andrea Baronchelli, Matteo Cinelli, et al. 2024. Persistent interaction patterns across social media platforms and over time. *Nature* 628, 8008 (Mar. 2024), 582–589. DOI: https://doi.org/10.1038/s41586-024-07229-y

[6] Christopher A. Bail, Lisa P. Argyle, Taylor W. Brown, John P. Bumpus, Haohan Chen, M. B. Fallin Hunzaker, Jaemin Lee, Marcus Mann, Friedolin Merhout, and Alexander Volfovsky. 2018. Exposure to opposing views on social media can increase political polarization. *Proceedings of the National Academy of Sciences* 115, 37 (Aug. 2018), 9216–9221. DOI: https://doi.org/10.1073/pnas.1804840115

[7] Oliver Beatson, Rachel Gibson, Marta Cantijoch Cunill, and Mark Elliot. 2021. Automation on twitter: Measuring the effectiveness of approaches to bot detection. *Social Science Computer Review* 41, 1 (Aug. 2021), 181–200. DOI: https://doi.org/10.1177/08944393211034991

[8] Mikhail Belkin and Partha Niyogi. 2003. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation* 15, 6 (Jun. 2003), 1373–1396.

[9] Vincent D. Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. 2008. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment* 2008, 10 (Oct. 2008), P10008. DOI: https://doi.org/10.1088/1742-5468/2008/10/p10008

[10] Ceren Budak, Brendan Nyhan, David M. Rothschild, Emily Thorson, and Duncan J. Watts. 2024. Misunderstanding the harms of online misinformation. *Nature* 630, 8015 (Jun. 2024), 45–53. DOI: https://doi.org/10.1038/s41586-024-07417-w

[11] Nikan Chavoshi, Hossein Hamooni, and Abdullah Mueen. 2016. DeBot: Twitter bot detection via warped correlation. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*. IEEE, 817–822. DOI: https://doi.org/10.1109/icdm.2016.0096

[12] Zi Chu, Steven Gianvecchio, Haining Wang, and Sushil Jajodia. 2010. Who is tweeting on twitter: Human, bot, or cyborg? In *Proceedings of the 26th Annual Computer Security Applications Conference (ACSAC '10)*. ACM, 21–30. DOI: https://doi.org/10.1145/1920261.1920265

[13] Rudi Cilibrasi and Paul Vitanyi. 2003. *Clustering by Compression*. arXiv:cs/0312044. Retrieved from https://arxiv.org/abs/cs/0312044

[14] Lorenzo Cima, Lorenzo Mannocci, Marco Avvenuti, Maurizio Tesconi, and Stefano Cresci. 2024. Coordinated behavior in information operations on twitter. *IEEE Access* 12 (2024), 61568–61585. DOI: https://doi.org/10.1109/access.2024.3393482

[15] Lorenzo Cima, Lorenzo Mannocci, Marco Avvenuti, Maurizio Tesconi, and Stefano Cresci. 2024. *Twitter Dataset about Information Operations in Honduras and UAE*. DOI: https://doi.org/10.5281/ZENODO.10619747

[16] M. Cinelli, M. Conti, L. Finos, F. Grisolia, P. Kralj Novak, A. Peruzzi, M. Tesconi, F. Zollo, and W. Quattrociocchi. 2019. (Mis)information operations: An integrated perspective. *Journal of Information Warfare* 18, 3 (2019), 83–98. Retrieved from https://www.jstor.org/stable/26894683

[17] Matteo Cinelli, Stefano Cresci, Walter Quattrociocchi, Maurizio Tesconi, and Paola Zola. 2022. Coordinated inauthentic behavior and information spreading on twitter. *Decision Support Systems* 160 (Sept. 2022), 113819. DOI: https://doi.org/10.1016/j.dss.2022.113819

[18] Matteo Cinelli, Gianmarco De Francisci Morales, Alessandro Galeazzi, Walter Quattrociocchi, and Michele Starnini. 2021. The echo chamber effect on social media. *Proceedings of the National Academy of Sciences* 118, 9 (Feb. 2021). DOI: https://doi.org/10.1073/pnas.2023301118

[19] Stefano Cresci. 2020. A decade of social bot detection. *Communications of the ACM* 63, 10 (Sept. 2020), 72–83. DOI: https://doi.org/10.1145/3409116

[20] Stefano Cresci, Roberto Di Pietro, Marinella Petrocchi, Angelo Spognardi, and Maurizio Tesconi. 2015. Fame for sale: Efficient detection of fake twitter followers. *Decision Support Systems* 80 (Dec. 2015), 56–71. DOI: https://doi.org/10.1016/j.dss.2015.09.003

[21] Stefano Cresci, Roberto Di Pietro, Marinella Petrocchi, Angelo Spognardi, and Maurizio Tesconi. 2016. DNA-inspired online behavioral modeling and its application to spambot detection. *IEEE Intelligent Systems* 31, 5 (Sept. 2016), 58–64. DOI: https://doi.org/10.1109/mis.2016.29

[22] Stefano Cresci, Roberto Di Pietro, Marinella Petrocchi, Angelo Spognardi, and Maurizio Tesconi. 2017. The paradigm-shift of social spambots: Evidence, theories, and tools for the arms race. In *Proceedings of the 26th International Conference on World Wide Web Companion (WWW '17 Companion)*. ACM Press, 963–972. DOI: https://doi.org/10.1145/3041021.3055135

[23] Stefano Cresci, Fabrizio Lillo, Daniele Regoli, Serena Tardelli, and Maurizio Tesconi. 2018. $FAKE: Evidence of spam and bot activity in stock microblogs on twitter. *Proceedings of the International AAAI Conference on Web and Social Media* 12, 1 (Jun. 2018). DOI: https://doi.org/10.1609/icwsm.v12i1.15073

[24] Stefano Cresci, Marinella Petrocchi, Angelo Spognardi, and Stefano Tognazzi. 2019. On the capability of evolved spambots to evade detection via genetic engineering. *Online Social Networks and Media* 9 (Jan. 2019), 1–16. DOI: https://doi.org/10.1016/j.osnem.2018.10.005

[25] Clayton Allen Davis, Onur Varol, Emilio Ferrara, Alessandro Flammini, and Filippo Menczer. 2016. BotOrNot: A system to evaluate social bots. In *Proceedings of the 25th International Conference Companion on World Wide Web (WWW '16 Companion)*. ACM Press, 273–274. DOI: https://doi.org/10.1145/2872518.2889302

[26] Bart De Clerck, Juan Carlos Fernandez Toledano, Filip Van Utterbeeck, and Luis E. C. Rocha. 2024. Detecting coordinated and bot-like behavior in twitter: The jürgen conings case. *EPJ Data Science* 13, 1 (Jun. 2024). DOI: https://doi.org/10.1140/epjds/s13688-024-00477-y

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

[28] Niccolò Di Marco, Sara Brunetti, Matteo Cinelli, and Walter Quattrociocchi. 2025. Post-hoc evaluation of nodes influence in information Cascades: The case of coordinated accounts. *ACM Transactions on the Web* 19, 2 (May 2025), 1–19. https://doi.org/10.1145/3700644

[29] Niccolò Di Marco, Matteo Cinelli, Shayan Alipour, and Walter Quattrociocchi. 2024. Users volatility on reddit and voat. *IEEE Transactions on Computational Social Systems* 11, 5 (2024), 5871–5879. DOI: https://doi.org/10.1109/TCSS.2024.3379318

[30] Niccolò Di Marco, Max Falkenberg, Alessandro Galeazzi, Maddalena Torricelli, and Maximilian Puelma Touzel. 2021. *Growing Climate Polarisation on Social Media*. DOI: https://doi.org/10.17605/OSF.IO/NU75J

[31] N. Di Marco, Edoardo Loru, Anita Bonetti, Alessandra Olga Grazia Serra, Matteo Cinelli, and Walter Quattrociocchi. 2024. Patterns of linguistic simplification on social media platforms over time. *Proceedings of the National Academy of Sciences of the United States of America* 121, 50 (Dec. 2024). DOI: https://doi.org/10.1073/pnas.2412105121

[32] Max Falkenberg, Alessandro Galeazzi, Maddalena Torricelli, Niccolò Di Marco, Francesca Larosa, Madalina Sas, Amin Mekacher, Warren Pearce, Fabiana Zollo, Walter Quattrociocchi, et al. 2022. Growing polarization around climate change on social media. *Nature Climate Change* 12, 12 (2022), 1114–1121. DOI: https://doi.org/10.1038/s41558-022-01527-x

[33] Shangbin Feng, Herun Wan, Ningnan Wang, and Minnan Luo. 2021. BotRGCN: Twitter bot detection with relational graph convolutional networks. In *Proceedings of the 2021 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM '21)*. ACM, 236–239. DOI: https://doi.org/10.1145/3487351.3488336

[34] Emilio Ferrara, Herbert Chang, Emily Chen, Goran Muric, and Jaimin Patel. 2020. Characterizing social media manipulation in the 2020 U.S. presidential election. *First Monday* (Oct. 2020). DOI: https://doi.org/10.5210/fm.v25i11.11431

[35] Emilio Ferrara, Onur Varol, Clayton Davis, Filippo Menczer, and Alessandro Flammini. 2016. The rise of social bots. *Communications of the ACM* 59, 7 (Jun. 2016), 96–104. DOI: https://doi.org/10.1145/2818717

[36] Santo Fortunato and Marc Barthélemy. 2007. Resolution limit in community detection. *Proceedings of the National Academy of Sciences* 104, 1 (Jan. 2007), 36–41. DOI: https://doi.org/10.1073/pnas.0605965104

[37] Maria Giatsoglou, Despoina Chatzakou, Neil Shah, Christos Faloutsos, and Athena Vakali. 2015. *Retweeting Activity on Twitter: Signs of Deception*. Springer International Publishing, 122–134. DOI: https://doi.org/10.1007/978-3-319-18038-0_10

[38] Fabio Giglietto, Nicola Righetti, Luca Rossi, and Giada Marino. 2020. It takes a village to manipulate the media: Coordinated link sharing behavior during 2018 and 2019 Italian elections. *Information, Communication and Society* 23, 6 (Mar. 2020), 867–891. DOI: https://doi.org/10.1080/1369118x.2020.1739732

[39] Rosario Gilmary, Akila Venkatesan, Govindasamy Vaiyapuri, and Deepikashini Balamurali. 2022. DNA-influenced automated behavior detection on twitter through relative entropy. *Scientific Reports* 12, 1 (May 2022). DOI: https://doi.org/10.1038/s41598-022-11854-w

[40] Wolfgang Glänzel and András Schubert. 1988. Characteristic scores and scales in assessing citation impact. *Journal of Information Science* 14, 2 (Apr. 1988), 123–127. DOI: https://doi.org/10.1177/016555158801400208

[41] Timothy Graham, Sam Hames, and Elizabeth Alpert. 2024. The coordination network toolkit: A framework for detecting and analysing coordinated behaviour on social media. *Journal of Computational Social Science* 7, 2 (May 2024), 1139–1160. https://doi.org/10.1007/s42001-024-00260-z

[42] Andrew M. Guess, Neil Malhotra, Jennifer Pan, Pablo Barberá, Hunt Allcott, Taylor Brown, Adriana Crespo-Tenorio, Drew Dimmery, Deen Freelon, Matthew Gentzkow, et al. 2023. How do social media feed algorithms affect attitudes and behavior in an election campaign? *Science* 381, 6656 (Jul. 2023), 398–404. DOI: https://doi.org/10.1126/science.abp9364

[43] Buyun He, Yingguang Yang, Qi Wu, Hao Liu, Renyu Yang, Hao Peng, Xiang Wang, Yong Liao, and Pengyuan Zhou. 2024. Dynamicity-aware social bot detection with dynamic graph transformers. In *Proceedings of the 33rd International Joint Conference on Artificial Intelligence (IJCAI '24)*. International Joint Conferences on Artificial Intelligence Organization, 5844–5852. DOI: https://doi.org/10.24963/ijcai.2024/646

[44] Philip N. Howard and Bence Kollanyi. 2016. Bots, #Strongerin, and #Brexit: Computational propaganda during the UK-EU referendum. arXiv.1606.06356. Retrieved from https://doi.org/https://arxiv.org/abs/1606.06356

[45] Kristina Hristakieva, Stefano Cresci, Giovanni Da San Martino, Mauro Conti, and Preslav Nakov. 2022. The spread of propaganda by coordinated communities on social media. In *14th ACM Web Science Conference 2022 (WebSci '22)*. ACM, 191–201. DOI: https://doi.org/10.1145/3501247.3531543

[46] Di Huang, Jinbao Song, and Xingyu Zhang. 2025. Semi-supervised social bot detection with relational graph attention transformers and characteristics of the social environment. *Information Fusion* 118 (Jun. 2025), 102956. DOI: https://doi.org/10.1016/j.inffus.2025.102956

[47] Di Huang, Jinbao Song, Xingyu Zhang, and Wenwen Yang. 2024. A social bot detection framework based heterogeneous graph attention and similarity graph features. In *Proceedings of the 2024 2nd International Conference on Computer, Internet of Things and Smart City (CIoTSC '24)*. ACM, 202–207. DOI: https://doi.org/10.1145/3731867.3731901

[48] Sofia Hurtado, Poushali Ray, and Radu Marculescu. 2019. Bot detection in reddit political discussion. In *Proceedings of the 4th International Workshop on Social Sensing (CPS-IoT Week '19)*. ACM, 30–35. DOI: https://doi.org/10.1145/3313294.3313386

[49] Meng Jiang, Peng Cui, Alex Beutel, Christos Faloutsos, and Shiqiang Yang. 2016. Catching synchronized behaviors in large networks: A graph mining approach. *ACM Transactions on Knowledge Discovery from Data* 10, 4 (Jun. 2016), 1–27. DOI: https://doi.org/10.1145/2746403

[50] Zhiying Jiang, Matthew Yang, Mikhail Tsirlin, Raphael Tang, Yiqin Dai, and Jimmy Lin. 2023. "Low-Resource" text classification: A parameter-free classification method with compressors. In *Findings of the Association for Computational Linguistics (ACL '23)*. Association for Computational Linguistics, 6810–6828. DOI: https://doi.org/10.18653/v1/2023.findings-acl.426

[51] Franziska B. Keller, David Schoch, Sebastian Stier, and JungHwan Yang. 2019. Political astroturfing on twitter: How to coordinate a disinformation campaign. *Political Communication* 37, 2 (Oct. 2019), 256–280. DOI: https://doi.org/10.1080/10584609.2019.1661888

[52] Tobias R. Keller and Ulrike Klinger. 2018. Social bots in election campaigns: Theoretical, empirical, and methodological implications. *Political Communication* 36, 1 (2018), 171–189. DOI: https://doi.org/10.1080/10584609.2018.1526238

[53] Ryan Kenny, Baruch Fischhoff, Alex Davis, Kathleen M. Carley, and Casey Canfield. 2022. Duped by bots: Why some are better than others at detecting fake social media personas. *Human Factors: The Journal of the Human Factors and Ergonomics Society* 66, 1 (Feb. 2022), 88–102. DOI: https://doi.org/10.1177/00187208211072642

[54] Kyumin Lee, Brian Eoff, and James Caverlee. 2021. Seven months with the devils: A long-term study of content polluters on twitter. *Proceedings of the International AAAI Conference on Web and Social Media* 5, 1 (Aug. 2021), 185–192. DOI: https://doi.org/10.1609/icwsm.v5i1.14106

[55] M. Li, X. Chen, X. Li, B. Ma, and P. M. B. Vitanyi. 2004. The similarity metric. *IEEE Transactions on Information Theory* 50, 12 (Dec. 2004), 3250–3264. DOI: https://doi.org/10.1109/tit.2004.838101

[56] Edoardo Loru, Matteo Cinelli, Maurizio Tesconi, and Walter Quattrociocchi. 2024. The influence of coordinated behavior on toxicity. *Online Social Networks and Media* 43–44 (Nov. 2024), 100289. DOI: https://doi.org/10.1016/j.osnem.2024.100289

[57] Edoardo Loru, Alessandro Galeazzi, Anita Bonetti, Emanuele Sangiorgio, Niccolò Di Marco, Matteo Cinelli, Max Falkenberg, Andrea Baronchelli, and Walter Quattrociocchi. 2025. Ideology and polarization set the agenda on social media. *Scientific Reports* 15, 1 (Oct. 2025). DOI: https://doi.org/10.1038/s41598-025-19776-z

[58] Luca Luceri, Eric Boniardi, and Emilio Ferrara. 2024. Leveraging large language models to detect influence campaigns on social media. In *Companion Proceedings of the ACM on Web Conference 2024 (WWW '24)*. ACM. DOI: https://doi.org/10.1145/3589335.3651912

[59] Luca Luceri, Valeria Pantè, Keith Burghardt, and Emilio Ferrara. 2024. Unmasking the web of deceit: Uncovering coordinated activity to expose information operations on twitter. In *Proceedings of the ACM on Web Conference 2024 (WWW '24)*. ACM, 2530–2541. DOI: https://doi.org/10.1145/3589334.3645529

[60] Yonatan Lupu, Richard Sear, Nicolas Velásquez, Rhys Leahy, Nicholas Johnson Restrepo, Beth Goldberg, and Neil F. Johnson. 2023. Offline events and online hate. *PloS One* 18, 1 (Jan. 2023), e0278511. DOI: https://doi.org/10.1371/journal.pone.0278511

[61] Michele Mazza, Stefano Cresci, Marco Avvenuti, Walter Quattrociocchi, and Maurizio Tesconi. 2019. *Italian Retweets Timeseries*. DOI: https://doi.org/10.5281/ZENODO.2653138

[62] Michele Mazza, Stefano Cresci, Marco Avvenuti, Walter Quattrociocchi, and Maurizio Tesconi. 2019. RTbust: Exploiting temporal patterns for botnet detection on twitter. In *Proceedings of the 10th ACM Conference on Web Science (WebSci '19)*. ACM. DOI: https://doi.org/10.1145/3292522.3326015

[63] Leland McInnes, John Healy, Nathaniel Saul, and Lukas Großberger. 2018. UMAP: Uniform manifold approximation and projection. *Journal of Open Source Software* 3, 29 (Sept. 2018), 861. DOI: https://doi.org/10.21105/joss.00861

[64] Amin Mekacher, Max Falkenberg, and Andrea Baronchelli. 2024. The koo dataset: An indian microblogging platform with global ambitions. *Proceedings of the International AAAI Conference on Web and Social Media* 18 (May 2024), 1991–2002. DOI: https://doi.org/10.1609/icwsm.v18i1.31442

[65] Marcelo Mendoza, Maurizio Tesconi, and Stefano Cresci. 2020. Bots in social and interaction networks: Detection and impact estimation. *ACM Transactions on Information Systems* 39, 1 (Oct. 2020), 1–32. DOI: https://doi.org/10.1145/3419369

[66] Marco Minici, Luca Luceri, Francesco Fabbri, and Emilio Ferrara. 2024. *IOHunter: Graph Foundation Model to Uncover Online Information Operations*. arXiv:2412.14663. Retrieved from https://arxiv.org/abs/2412.14663

[67] Mary Luz Mouronte-López, Javier Gómez Sánchez-Seco, and Rosa M. Benito. 2024. Patterns of human and bots behaviour on twitter conversations about sustainability. *Scientific Reports* 14, 1 (Feb. 2024). DOI: https://doi.org/10.1038/s41598-024-52471-z

[68] Seung Ho Na, Sumin Cho, and Seungwon Shin. 2023. Evolving bots: The new generation of comment bots and their underlying scam campaigns in YouTube. In *Proceedings of the 2023 ACM on Internet Measurement Conference (IMC '23)*. ACM, 297–312. DOI: https://doi.org/10.1145/3618257.3624822

[69] Lynnette Hui Xian Ng and Kathleen M. Carley. 2023. BotBuster: Multi-platform bot detection using a mixture of experts. *Proceedings of the International AAAI Conference on Web and Social Media* 17 (June. 2023), 686–697. DOI: https://doi.org/10.1609/icwsm.v17i1.22179

[70] Lynnette Hui Xian Ng and Kathleen M. Carley. 2023. A combined synchronization index for evaluating collective action social media. *Applied Network Science* 8, 1 (Jan. 2023). DOI: https://doi.org/10.1007/s41109-022-00526-3

[71] Lynnette Hui Xian Ng and Kathleen M. Carley. 2024. Assembling a multi-platform ensemble social bot detector with applications to US 2020 elections. *Social Network Analysis and Mining* 14, 1 (Feb. 2024). DOI: https://doi.org/10.1007/s13278-024-01211-2

[72] Leonardo Nizzoli, Serena Tardelli, Marco Avvenuti, Stefano Cresci, and Maurizio Tesconi. 2021. Coordinated behavior on social media in 2019 UK general election. *Proceedings of the International AAAI Conference on Web and Social Media* 15 (May 2021), 443–454. DOI: https://doi.org/10.1609/icwsm.v15i1.18074

[73] Leonardo Nizzoli, Serena Tardelli, Marco Avvenuti, Stefano Cresci, and Maurizio Tesconi. 2021. Twitter Dataset—Coordinated Behavior on Social Media in 2019 UK General Election. DOI: https://doi.org/10.5281/ZENODO.4647893

[74] Alexander C. Nwala, Alessandro Flammini, and Filippo Menczer. 2023. A language framework for modeling social media account behavior. *EPJ Data Science* 12, 1 (Aug. 2023). DOI: https://doi.org/10.1140/epjds/s13688-023-00410-9

[75] Brendan Nyhan, Jaime Settle, Emily Thorson, Magdalena Wojcieszak, Pablo Barberá, Annie Y. Chen, Hunt Allcott, Taylor Brown, Adriana Crespo-Tenorio, Drew Dimmery, et al. 2023. Like-minded sources on Facebook are prevalent but not polarizing. *Nature* 620, 7972 (Jul. 2023), 137–144. DOI: https://doi.org/10.1038/s41586-023-06297-w

[76] Diogo Pacheco, Pik-Mai Hui, Christopher Torres-Lugo, Bao Tran Truong, Alessandro Flammini, and Filippo Menczer. 2021. Uncovering coordinated networks on social media: Methods and case studies. *Proceedings of the International AAAI Conference on Web and Social Media* 15 (May 2021), 455–466. DOI: https://doi.org/10.1609/icwsm.v15i1.18075

[77] Tiago P. Peixoto. 2014. Hierarchical block structures and high-resolution model selection in large networks. *Physical Review X*. 4, 1 (Mar. 2014), 011047. 10.1103/PhysRevX.4.011047

[78] Dorian Quelle and Alexandre Bovet. 2024. *Bluesky: Network Topology, Polarisation, and Algorithmic Curation*. arXiv:2405.17571. Retrieved from https://arxiv.org/abs/2405.17571

[79] William M. Rand. 1971. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association* 66, 336 (Dec. 1971), 846. DOI: https://doi.org/10.2307/2284239

[80] Jacob Ratkiewicz, Michael Conover, Mark Meiss, Bruno Goncalves, Alessandro Flammini, and Filippo Menczer. 2021. Detecting and tracking political abuse in social media. *Proceedings of the International AAAI Conference on Web and Social Media* 5, 1 (Aug. 2021), 297–304. DOI: https://doi.org/10.1609/icwsm.v5i1.14127

[81] Adrian Rauchfleisch and Jonas Kaiser. 2020. The false positive problem of automatic bot detection in social science research. *PloS One* 15, 10 (Oct. 2020), e0241045. DOI: https://doi.org/10.1371/journal.pone.0241045

[82] Ana Lucía Schmidt, Fabiana Zollo, Michela Del Vicario, Alessandro Bessi, Antonio Scala, Guido Caldarelli, H. Eugene Stanley, and Walter Quattrociocchi. 2017. Anatomy of news consumption on facebook. *Proceedings of the National Academy of Sciences* 114, 12 (2017), 3035–3039.

[83] David Schoch, Franziska B. Keller, Sebastian Stier, and JungHwan Yang. 2022. Coordination patterns reveal online political astroturfing across the world. *Scientific Reports* 12, 1 (Mar. 2022). DOI: https://doi.org/10.1038/s41598-022-08404-9

[84] M. Ángeles Serrano, Marián Boguñá, and Alessandro Vespignani. 2009. Extracting the multiscale backbone of complex weighted networks. *Proceedings of the National Academy of Sciences* 106, 16 (Apr. 2009), 6483–6488. DOI: https://doi.org/10.1073/pnas.0808904106

[85] Chengcheng Shao, Giovanni Luca Ciampaglia, Onur Varol, Kai-Cheng Yang, Alessandro Flammini, and Filippo Menczer. 2018. The spread of low-credibility content by social bots. *Nature Communications* 9, 1 (2018). DOI: https://doi.org/10.1038/s41467-018-06930-7

[86] Kate Starbird, Ahmer Arif, and Tom Wilson. 2019. Disinformation as collaborative work: Surfacing the participatory nature of strategic information operations. *Proceedings of the ACM on Human-Computer Interaction* 3, CSCW (Nov. 2019), 1–26. DOI: https://doi.org/10.1145/3359229

[87] Massimo Stella, Emilio Ferrara, and Manlio De Domenico. 2018. Bots increase exposure to negative and inflammatory content in online social systems. *Proceedings of the National Academy of Sciences* 115, 49 (2018), 12435–12440. DOI: https://doi.org/10.1073/pnas.1803470115

[88] Stefan Stieglitz, Florian Brachten, Bjorn Ross, and Anna Jung. 2018. Do social bots dream of electric sheep? A categorisation of social media bot accounts. In *Proceedings of the 27th Australian Conference on Information Systems (ACIS '17)*.

[89] Serena Tardelli, Leonardo Nizzoli, Marco Avvenuti, Stefano Cresci, and Maurizio Tesconi. 2024. Multifaceted online coordinated behavior in the 2020 US presidential election. *EPJ Data Science* 13, 1 (Apr. 2024). DOI: https://doi.org/10.1140/epjds/s13688-024-00467-0

[90] Serena Tardelli, Leonardo Nizzoli, Maurizio Tesconi, Mauro Conti, Preslav Nakov, Giovanni Da San Martino, and Stefano Cresci. 2024. Temporal dynamics of coordinated online behavior: Stability, archetypes, and influence. *Proceedings of the National Academy of Sciences* 121, 20 (May 2024). DOI: https://doi.org/10.1073/pnas.2307038121

[91] Kurt Thomas, Chris Grier, Dawn Song, and Vern Paxson. 2011. Suspended accounts in retrospect: An analysis of twitter spam. In *Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement Conference (IMC '11)*. ACM, 243–258. DOI: https://doi.org/10.1145/2068816.2068840

[92] Joshua Uyheng, Daniele Bellutta, and Kathleen M. Carley. 2022. Bots amplify and redirect hate speech in online discourse about racism during the COVID-19 pandemic. *Social Media + Society* 8, 3 (Jul. 2022), 205630512211047. DOI: https://doi.org/10.1177/20563051221104749

[93] Carlo M. Valensise, Matteo Cinelli, and Walter Quattrociocchi. 2023. The drivers of online polarization: Fitting models to data. *Information Sciences* 642 (2023), 119152.

[94] Nguyen Xuan Vinh, Julien Epps, and James Bailey. 2010. Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. *Journal of Machine Learning Research* 11 (Dec. 2010), 2837–2854.

[95] Padinjaredath Suresh Vishnuprasad, Gianluca Nogara, Felipe Cardoso, Stefano Cresci, Silvia Giordano, and Luca Luceri. 2024. Tracking fringe and coordinated activity on twitter leading up to the US capitol attack. *Proceedings of the International AAAI Conference on Web and Social Media*, 18 (May 2024), 1557–1570. DOI: https://doi.org/10.1609/icwsm.v18i1.31409

[96] Soroush Vosoughi, Deb Roy, and Sinan Aral. 2018. The spread of true and false news online. *Science* 359, 6380 (2018), 1146–1151.

[97] Xiujuan Wang, Keke Wang, Kangmiao Chen, Zhengxiang Wang, and Kangfeng Zheng. 2024. Unsupervised twitter social bot detection using deep contrastive graph clustering. *Knowledge-Based Systems* 293 (June. 2024), 111690. DOI: https://doi.org/10.1016/j.knosys.2024.111690

[98] Derek Weber and Frank Neumann. 2021. Amplifying influence through coordinated behaviour in social networks. *Social Network Analysis and Mining* 11, 1 (Oct. 2021). DOI: https://doi.org/10.1007/s13278-021-00815-2

[99] Jen Weedon, William Nuland, and Alex Stamos. 2017. Information operations and Facebook. Retrieved from https://about.fb.com/wp-content/uploads/2017/04/facebook-and-information-operations-v1.pdf

[100] Samuel Woolley and Philip Howard. 2016. Automation, algorithms, and politics—Political communication, computational propaganda, and autonomous agents—Introduction. *International Journal of Communication* 10, 0 (2016). Retrieved from https://ijoc.org/index.php/ijoc/article/view/6298

[101] Chao Yang, Robert Harkreader, and Guofei Gu. 2013. Empirical evaluation and new design for fighting evolving twitter spammers. *IEEE Transactions on Information Forensics and Security* 8, 8 (Aug. 2013), 1280–1293. DOI: https://doi.org/10.1109/tifs.2013.2267732

[102] Robert Chandler Chao Yang and Guofei Gu Harkreader. 2011. *Die Free or Live Hard? Empirical Evaluation and New Design for Fighting Evolving Twitter Spammers*. Springer, Berlin, 318–337. DOI: https://doi.org/10.1007/978-3-642-23644-0_17

[103] Yingguang Yang, Renyu Yang, Yangyang Li, Kai Cui, Zhiqin Yang, Yue Wang, Jie Xu, and Haiyong Xie. 2023. RoSGAS: Adaptive social bot detection with reinforced self-supervised GNN architecture search. *ACM Transactions on the Web* 17, 3 (May 2023), 1–31. DOI: https://doi.org/10.1145/3572403

Appendix

A Additional Results and Baselines

Figure A1 shows the running time of our network approximation algorithm against building the complete similarity network, in the case of the toy dataset of 1,074 users presented in Section 3.4.2. Supplementing Figure 2, these results show that even by selecting a small percentage of nodes, a promising clustering agreement can be achieved along with a substantial improvement in running time. For instance, by using $\eta = 10$ ($\approx 1\%$ of the total nodes) and $\mu = 2$, a Rand Index of 0.7 can be obtained with a 10-fold improvement in running time. As discussed in Section 3.4.2, it is important to note that these values may vary depending on the dataset or the software implementation of the heuristic and the complete network-building procedure. Additionally, because constructing a complete similarity network scales quadratically with the number of nodes, there may be situations where the network becomes too large to build without access to exceptional hardware. In these instances, an efficient heuristic that can approximate the network while preserving its overall community structure becomes a valuable alternative.

In the case of the *Honduras IO* (Section 4.1.2), the *cresci-rtbust-2019* (Section 4.1.3), and the *cresci-2015* (Section 4.2.1) datasets, we perform an unsupervised classification of accounts that depends on the selection of a threshold t. In detail, we classify as "suspicious" all users with at least an edge e such that $\text{NCD}_e < t$. The classification performance across a range of values of t is displayed in Figure A2.

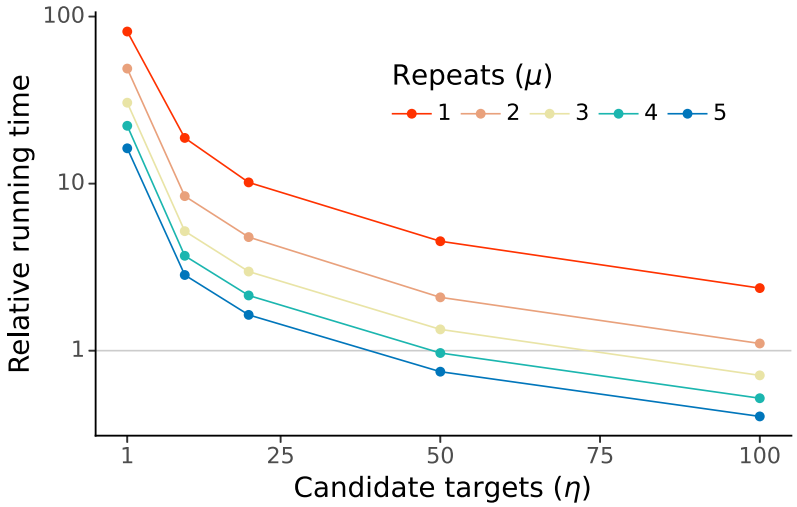

Fig. A1. *Running time of our network approximation algorithm relative to building the complete similarity network.* For each pair of η and μ, we build the approximate network of our toy dataset of 1,074 nodes and measure its running time. Then, we divide the time required to build the complete network by the obtained running time. Therefore, a value on the y-axis greater than 1 indicates that the heuristic is faster than the baseline by a factor of y, whereas a value less than 1 indicates that it is slower by a factor of $1/y$.

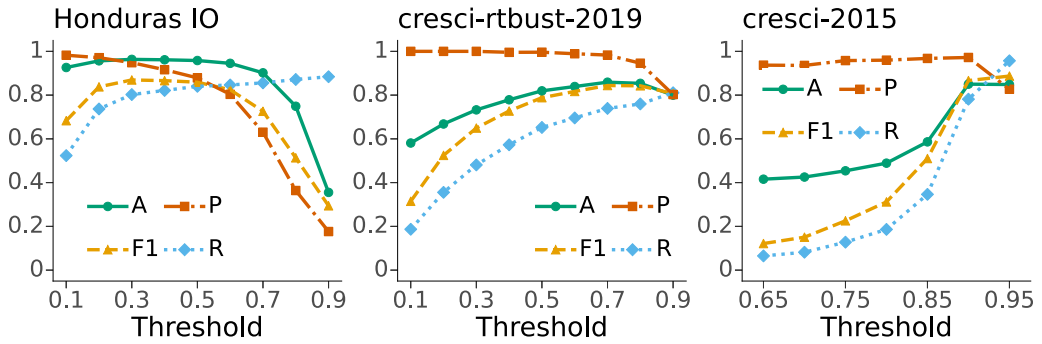

Fig. A2. *Unsupervised classification of users.* We classify as "suspicious" all users with at least one edge e such that $NCD_e < t$, evaluating Accuracy, Precision, Recall, and F1-score across a range of thresholds t.

In Figure A3, we report the clustering agreement between all clusters detected in our similarity network for the *UK 2019* dataset (see Section 4.1.1) and those originally found by the authors. The heatmap shows that each of our detected clusters has a high overlap with one of the coordinated communities of superspreaders originally detected in the dataset.

Table A1 presents a comparison of our method's performance on the three benchmark datasets against existing baselines. For each baseline, we include both the original study and the source of the reported benchmark. In case multiple F1-scores are provided for a method, only the highest is reported. Although our method does not surpass existing baselines, it achieves performance close to the state of the art while being suitable for both bot and coordination detection, unlike alternative ad-hoc approaches. Table A2 reports additional metrics from supervised classification.

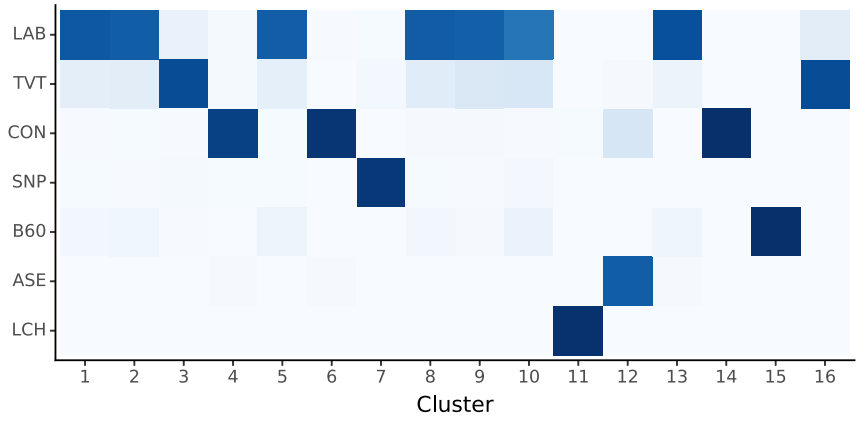

Fig. A3. *Clustering agreement with* [72] *for the UK2019 dataset.* On the x-axis, the clusters identified on our similarity network with the Louvain algorithm with resolution $\gamma = 1$, numbered according to their size (largest to smallest); on the y-axis, the clusters originally found with the Louvain algorithm by [72]. The color of a tile indicates the fraction of users in cluster x that are in cluster y, with a darker (lighter) color corresponding to a larger (smaller) fraction.

Table A1. Performance against Baselines

Dataset	Method	F1-score	Source
Honduras IO	Ours (supervised)	0.968	-
	Ours (unsupervised)	0.87	-
	Original paper [90]	0.91	[90]
cresci-rtbust-2019	Ours (supervised)	0.83	-
	Ours (unsupervised)	0.84	-
	Original paper [62]	0.8687	[62]
	BotBuster [69]	0.6720	[69]
	Botometer [25]	0.4286	[62]
cresci-2015	Ours (supervised)	0.982	-
	Ours (unsupervised)	0.89	-
	Original paper [20]	0.991	[20]
	Botometer [25]	0.669	[46]
	BotBuster [69]	0.9774	[46]
	BotRGCN [33]	0.9759	[46]
	SRGAT [46]	0.9968	[46]

For each result, we report the method's name and reference, as well as the reference for the computed F1-score.

Table A2. Performance of Our Method with Supervised Classification

Dataset	F1-score	Accuracy	Precision	Recall	AUC
Honduras IO	0.968 ± 0.019	0.9914 ± 0.0047	0.947 ± 0.031	0.993 ± 0.013	0.9863 ± 0.0095
cresci-rtbust-2019	0.836 ± 0.083	0.837 ± 0.065	0.80 ± 0.12	0.878 ± 0.056	0.902 ± 0.057
cresci-2015	0.982 ± 0.021	0.977 ± 0.029	0.985 ± 0.022	0.980 ± 0.023	0.988 ± 0.022

For each metric, we report the average and standard deviation across 10-fold cross-validation.

Received 14 January 2025; revised 21 October 2025; accepted 20 November 2025