Less is More: Semi-Supervised Causal Inference for Detecting Pathogenic Users in Social Media

Hamidreza Alvari
Arizona State University
halvari@asu.edu

Elham Shaabani
Arizona State University
eshaaban@asu.edu

Soumajyoti Sarkar
Arizona State University
ssarka18@asu.edu

Ghazaleh Beigi
Arizona State University
gbeigi@asu.edu

Paulo Shakarian
Arizona State University
shak@asu.edu

ABSTRACT

Recent years have witnessed a surge of manipulation of public opinion and political events by malicious social media actors. These users are referred to as "Pathogenic Social Media (PSM)" accounts. PSMs are key users in spreading misinformation in social media to viral proportions. These accounts can be either controlled by real users or automated bots. Identification of PSMs is thus of utmost importance for social media authorities. The burden usually falls to automatic approaches that can identify these accounts and protect social media reputation. However, lack of sufficient labeled examples for devising and training sophisticated approaches to combat these accounts is still one of the foremost challenges facing social media firms. In contrast, unlabeled data is abundant and cheap to obtain thanks to massive user-generated data. In this paper, we propose a semi-supervised causal inference PSM detection framework, SemiPsm, to compensate for the lack of labeled data. In particular, the proposed method leverages unlabeled data in the form of manifold regularization and only relies on cascade information. This is in contrast to the existing approaches that use exhaustive feature engineering (e.g., profile information, network structure, etc.). Evidence from empirical experiments on a real-world ISIS-related dataset from Twitter suggests promising results of utilizing unlabeled instances for detecting PSMs.

CCS CONCEPTS

• **Information systems → Social networks**; • **Security and privacy → Social aspects of security and privacy**.

KEYWORDS

Semi-supervised learning; Causal inference; Pathogenic users; Social media

ACM Reference Format:
Hamidreza Alvari, Elham Shaabani, Soumajyoti Sarkar, Ghazaleh Beigi, and Paulo Shakarian . 2019. Less is More: Semi-Supervised Causal Inference for Detecting Pathogenic Users in Social Media. In *Companion Proceedings of the 2019 World Wide Web Conference (WWW '19 Companion), May 13–17, 2019, San Francisco, CA, USA.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3308560.3316500

WWW '19 Companion, May 13–17, 2019, San Francisco, CA, USA

ACM ISBN 978-1-4503-6675-5/19/05.
https://doi.org/10.1145/3308560.3316500

1 INTRODUCTION

Over the past years, social media play major role in massive dissemination of misinformation online. Political events and public opinion on the Web and social networks have been allegedly manipulated by different forms of accounts including real users and automated software (a.k.a social bots or sybil accounts). Pathogenic Social Media (PSM) accounts are among those that are responsible for such a massive spread of disinformation online and swaying normal users' opinion [2, 35]. These accounts (1) are usually owned by either normal users or automated bots, (2) seek to promote or degrade certain ideas; and (3) can appear in many forms such as terrorist supporters (e.g., ISIS supporters), water armies or fake news writers. Understanding the behavior of PSMs will allow social media to take countermeasures against their propaganda at the early stage and reduce their threat to the public.

The problem of identification of PSMs has long been addressed in the past by the research community mostly in the form of bot detection. Several approaches especially supervised learning methods have been proposed in the literature and they have shown promising results [30]. However, for the most part, these approaches are often based on labeled data and exhaustive feature engineering. Examples of such feature groups include but are not limited to content, sentiment of posts, profile information and network features. These approaches are thus very expensive as they require significant amount of efforts to design features and annotate large labeled datasets. In contrast, unlabeled data is ubiquitous and cheap to collect thanks to the massive user-generated data produced on a daily basis. Thus, in this work we set out to examine if unlabeled instances can be utilized to compensate for the lack of enough labeled data.

Present Work. In this paper, semi-supervised causal inference is tailored to detect PSMs who are promoters of misinformation online. We cast the problem of identifying PSMs as an optimization problem and propose a semi-supervised causal learning framework which utilizes unlabeled examples through manifold regularization [11]. In particular, we incorporate causality-based features extracted from users' activity log (i.e., cascades of retweets) as regularization terms into the optimization problem. In this work, causal inference is leveraged in an effort to capture whether or not PSMs exert causal influences while making a message viral. Our

causality-based features are built upon *Suppes' theory of probabilistic causation* [40] whose central concept is *prima facie causes*: an event to be recognized as a cause, must occur before the effect and must lead to an increase of the likelihood of observing the effect. While there exists a prolific literature on causality and their great impact in the computer-science community (see [32] for instance), we build our foundation on *Suppes' theory* as it is computationally less complex.

Key idea and highlights. To summarize, this paper makes the following main contributions:

- We frame the problem of detecting PSM accounts as an optimization problem and present a Laplacian semi-supervised causal inference SemiPsm for solving it. The unlabeled data are utilized via manifold regularization.
- Manifold regularization used in the resultant optimization formulation is built upon causality-based features created on a notion of *Suppes' theory of probabilistic causation*.
- We conduct a suite of experiments using different supervised and semi-supervised methods. Empirical experiments on a real-world ISIS-related dataset from Twitter suggests the effectiveness of the proposed semi-supervised causal inference over the existing methods.

The remainder of the paper is organized as follows. In Section 2, we present the proposed framework. Section 3 summarizes the empirical experiments on an ISIS-related dataset from Twitter. In Section 4 we review the state-of-the-art methods. We conclude the paper in Section 5 by presenting the future work.

2 THE PROPOSED METHOD

In this section, we first provide the causal inference used to extract features out of users' activity log. Then, we detail the proposed semi-supervised causal inference, namely SemiPsm, for detecting PSM accounts.

2.1 Causal Inference

We follow the convention of [23] and assume an *action log* $\mathcal{A}$ of the form *Actions(User,Action,Time)*, which contains tuples (i, a_i, t_i) indicating that user i has performed action a_i at time t_i. For ease of exposition, we slightly abuse the notation and use the tuple (i, m, t) to indicate that user i has posted (tweeted/retweeted) message m at time t. For a given message m we define a *cascade* of actions as $\mathcal{A}_m = \{(i, m', t) \in \mathcal{A} | m' = m\}$. User i is called m-participant if there exists t_i such that $(i, m, t_i) \in \mathcal{A}$. Users who have adopted a message in the early stage of its life span are called *key users*:

Definition 1 (Key Users). *Given message m, m-participant i and cascade $\mathcal{A}_m$, we say user i is a key user iff user i precedes at least ϕ fraction of other m-participants where $\phi \in (0, 1)$. In other words, $|\mathcal{A}_m| \times \phi \leq |\{j | \exists t' : (j, m, t') \in \mathcal{A} \wedge t < t'\}|$, where $|.|$ is the cardinality of a set.*

Next, we shall define viral messages as follows.

Definition 2 (Viral Messages). *Given a threshold θ, we say a message $m \in \mathbf{M}$ is viral iff $|\mathcal{A}_m| \geq \theta$. We denote a set of all viral messages by $\mathbf{M}_{vir}$.*

The prior probability of a message going viral is $\rho = |\mathbf{M}_{vir}|/|\mathbf{M}|$. The probability of a message going viral given key user i has participated in, is computed as follows:

$$\rho_i = \frac{|\{m | m \in \mathbf{M}_{vir} \wedge i \text{ is a key user}\}|}{|\{m | m \in \mathbf{M} \wedge i \text{ is a key user}\}|} \tag{1}$$

The probability that key users i and j tweet/retweet message m chronologically and make it viral, is computed as:

$$p_{i,j} = \frac{|\{m \in \mathbf{M}_{vir} | \exists t, t' : t < t' \wedge (i, m, t), (j, m, t') \in \mathcal{A}\}|}{|\{m \in \mathbf{M} | \exists t, t' : t < t' \wedge (i, m, t), (j, m, t') \in \mathcal{A}\}|} \tag{2}$$

To examine how causal user i was in helping a message m going viral, we shall explore what will happen if we exclude user i from m. We define the probability that *only* key user j has made a message m viral, i.e. user i has not posted m or does not precede j as:

$$p_{\neg i,j} = \frac{|\{m \in \mathbf{M}_{vir} | \exists t' : (j, m, t') \in \mathcal{A} \wedge \nexists t : t < t', (i, m, t) \in \mathcal{A}\}|}{|\{m \in \mathbf{M} | \exists t' : (j, m, t') \in \mathcal{A} \wedge \nexists t : t < t', (i, m, t) \in \mathcal{A}\}|} \tag{3}$$

In this work we adopt the notion of *prima facie causes* which is at the core of Suppes' theory of probabilistic causation [40] and introduce causality metrics. According to this theory, *a certain event to be recognized as a cause, must occur before the effect and must lead to an increase of the likelihood of observing the effect.* Accordingly, prima facie causal users are defined as follows:

Definition 3 (Prima Facie Causal Users). *A user i is said to be Prima Facie causal user for cascade $\mathcal{A}_m$ iff: (1) user i is a key user of m, (2) $m \in \mathbf{M}_{vir}$, and (3) $\rho_i > \rho$.*

We use the concept of *related users* from a rule-based system [38] which was an extension to the causal inference framework in [28]. Accordingly, we call users i and j m-related if (1) they are Prima Facie causal users for m, and (2) i precedes j. We then define a set of user i's related users as $\mathbf{R}(i) = \{j | j \neq i \text{ and } i, j \text{ are } m\text{-related}\}$.

In this work, we use the time-decay causal metrics introduced in [2] which are built on Suppes' theory. The first metric used in this work is $\mathcal{E}_{K\&M}$ which is computed over a given time interval I as follows:

$$\mathcal{E}^I_{K\&M}(i) = \frac{\sum_{j \in \mathcal{R}(i)} (\mathcal{P}_{i,j} - \mathcal{P}_{\neg i,j})}{|\mathcal{R}(i)|} \tag{4}$$

where $\mathcal{R}(i)$, $\mathcal{P}_{i,j}$, and $\mathcal{P}_{\neg i,j}$ are now defined over I. This metric estimates the causality score of user i in making a message *viral*, by taking the average of $\mathcal{P}_{i,j} - \mathcal{P}_{\neg i,j}$ over $\mathbf{R}(i)$. The intuition here is that user i is more likely to be a cause of message m to become viral than user j, if $\mathcal{P}_{i,j} - \mathcal{P}_{\neg i,j} > 0$. This metric cannot spot all PSMs, hence another metric is defined, namely relative likelihood causality $\mathcal{E}_{rel}$. This metric works by assessing relative difference between $\mathcal{P}_{i,j}$, and $\mathcal{P}_{\neg i,j}$:

$$\mathcal{E}^I_{rel}(i) = \frac{\mathcal{S}(i,j)}{|\mathcal{R}(i)|} \tag{5}$$

where $\mathcal{S}(i,j)$ is defined as follows and α is infinitesimal:

$$\mathcal{S}(i,j) = \begin{cases} \frac{\mathcal{P}_{i,j}}{\mathcal{P}_{\neg i,j} + \alpha} - 1, & \mathcal{P}_{i,j} > \mathcal{P}_{\neg i,j} \\ 1 - \frac{\mathcal{P}_{\neg i,j}}{\mathcal{P}_{i,j}}, & \mathcal{P}_{i,j} \leq \mathcal{P}_{\neg i,j} \end{cases} \tag{6}$$

Two other neighborhood-based metrics were also defined in [2], first of which is computed as:

$$\mathcal{E}^I_{nb}(j) = \frac{\sum_{i \in Q(j)} \mathcal{E}^I_{K\&M}(i)}{|Q(j)|} \tag{7}$$

where $Q(j) = \{i | j \in \mathcal{R}(i)\}$ is the set of all users that user j belongs to their related users sets. Similarly, the second metric is the weighted version of the above metric and is called weighted neighborhood-based causality and is calculated as:

$$\mathcal{E}^I_{wnb}(j) = \frac{\sum_{i \in Q(j)} w_i \times \mathcal{E}^I_{K\&M}(i)}{\sum_{i \in Q(j)} w_i} \tag{8}$$

The aim of this metric is to capture different impacts that users in $Q(j)$ might have on user j.

2.2 Final set of Features

Finally, the causal metrics discussed in the previous section will be fed as features to the semi-supervised framework– this will be described in the next section. The final set of features is in the following generic form ξ^I_k where $k \in \{K\&M, rel, nb, wnb\}$ [2]:

$$\xi^I_k(i) = \frac{1}{|\mathcal{T}|} \sum_{t' \in \mathcal{T}} e^{-\sigma(t-t')} \times \mathcal{E}^{\Delta}_k(i) \tag{9}$$

Here, σ is a scaling parameter of the exponential decay function, $\mathcal{T} = \{t' | t' = t_0 + j \times \delta, j \in \mathbb{N} \wedge t' \leq t - \delta\}$ is a sequence of sliding-time windows, and δ is a small fixed amount of time, which is used as the length of each sliding-time window $\Delta = [t' - \delta, t']$.

In essence, this metric assigns different weights to different time points of a given time interval, inversely proportional to their distance from t (i.e., smaller distance is associated with higher weight). Specifically, it performs the following: it (1) breaks down the given time interval into shorter time periods using a sliding time window, (2) deploys an exponential decay function of the form $f(x) = e^{-\alpha x}$ to account for the time-decay effect, and (3) takes average of the causality values computed over each sliding time window [2].

2.3 Semi-Supervised Causal Inference

Having defined the causality-based features, we now proceed to present the proposed semi-supervised Laplacian SVM framework, SemiPsm. For the rest of the discussion, we shall assume a set of l labeled pairs $\{(x_i, y_i)\}_{i=1}^{l}$ and an unlabeled set of u instances $\{x_{l+i}\}_{i=1}^{u}$, where $x_i \in \mathbb{R}^n$ denotes the causality vector $\xi^I_k(i)$ of user i and $y_i \in \{+1, -1\}$ (PSM or not).

Recall for the standard soft-margin support vector machines, the following optimization problem is solved:

$$\min_{f_\theta \in \mathcal{H}_k} \gamma ||f_\theta||^2_k + C_l \sum_{i=1}^{l} H_1(y_i f_\theta(x_i)) \tag{10}$$

In the above equation, $f_\theta(\cdot)$ is a decision function of the form $f_\theta(\cdot) = w.\Phi(\cdot) + b$ where $\theta = (w, b)$ are the parameters of the model, and $\Phi(\cdot)$ is the feature map which is usually implemented using the kernel trick [17]. Also, the function $H_1(\cdot) = \max(0, 1 - \cdot)$ is the Hinge Loss function. The classical Representer theorem [10] suggests that solution to the optimization problem exists in a Hilbert space $\mathcal{H}_k$ and is of the form $f^*_\theta(x) = \sum_{i=1}^{l} \alpha^*_i \mathbf{K}(x, x_i)$. Here, $\mathbf{K}$ is the $l \times l$ Gram matrix over labeled samples. Equivalently, the above problem can be written as:

$$\min_{w,b,\epsilon} \frac{1}{2}||w||^2_2 + C_l \sum_{i=1}^{l} \epsilon_i \tag{11}$$

$$s.t.\ y_i(w.\Phi(x_i) + b) \geq 1 - \epsilon_i,\ i = 1, ..., l$$

$$\epsilon_i \geq 0,\ i = 1, ..., l \tag{12}$$

Next, we will use the above optimization equation as our basis to derive the formulations for our proposed semi-supervised learner.

The basic assumption behind semi-supervised learning methods is to leverage unlabeled instances in order to restructure hypotheses during the learning process [4]. Here, exogenous information extracted from causality-based features of users is exploited to make a better use of the unlabeled examples. To do so, we first introduce matrix $\mathbf{F}$ over both of the labeled and unlabeled samples with $\mathbf{F}_{ij} = ||\Phi(x_i) - \Phi(x_j)||_2$ in $||.||_2$ norm. This way, we force instances x_i and x_j in our dataset to be relatively 'close' to each other [8], i.e., having a same label, if their corresponding causal-based feature vectors are close. To account for this, a regularization term is added to the standard equation and the following optimization is solved:

$$\min_{f_\theta \in \mathcal{H}_k} \frac{1}{2} \sum_{i=1}^{l} \mathbf{F}_{ij} ||f_\theta(x_i) - f_\theta(x_j)||^2_2 = \mathbf{f}^T_\theta \mathcal{L}^T \mathbf{f}_\theta \tag{13}$$

where $\mathbf{f} = [f(x_1), ..., f(x_{l+u})]^T$ and $\mathcal{L}$ is the Laplacian matrix based on $\mathbf{F}$ given by $\mathcal{L} = \mathbf{D} - \mathbf{F}$, and $\mathbf{D}_{ii} = \sum_{j=1}^{l+u} \mathbf{F}_{ij}$. The intuition here is that causal pairs are more likely to have same labels than others.

Following the notations used in [11] and by including our regularization term, we would extend the standard equation by solving the following optimization:

$$\min_{f_\theta \in \mathcal{H}_k} \gamma ||f_\theta||^2_k + C_l \sum_{i=1}^{l} H_1(y_i f_\theta(x_i)) + C_r \mathbf{f}^T_\theta \mathcal{L} \mathbf{f}_\theta \tag{14}$$

Again, solution in $\mathcal{H}_k$ would be in the following form $f^*_\theta(x) = \sum_{i=1}^{l+u} \alpha^*_i \mathbf{K}(x, x_i)$. Here $\mathbf{K}$ is the $(l+u) \times (l+u)$ Gram matrix over all samples. The Eq. 14 could be then written as follows:

$$\min_{\alpha,b,\epsilon} \frac{1}{2} \alpha^T \mathbf{K} \alpha + C_l \sum_{i=1}^{l} \epsilon_i + \frac{C_r}{2} \alpha^T \mathbf{K} \mathcal{L} \mathbf{K} \alpha \tag{15}$$

$$s.t.\ y_i(\sum_{j=1}^{l+u} \alpha_j \mathbf{K}(x_i, x_j) + b) \geq 1 - \epsilon_i,\ i = 1, ..., l$$

$$\epsilon_i \geq 0,\ i = 1, ..., l \tag{16}$$

With introduction of the Lagrangian multipliers β and γ, we write the Lagrangian function of the above equation as follows:

$$L(\alpha, \epsilon, b, \beta, \gamma) = \frac{1}{2} \alpha^T \mathbf{K}(I + C_r \mathcal{L})\alpha + C_l \sum_{i=1}^{l} \epsilon_i \tag{17}$$

$$- \sum_{i=1}^{l} \beta_i (y_i (\sum_{j=1}^{l+u} \alpha_j \mathbf{K}(x_i, x_j) + b) - 1 + \epsilon_i) - \sum_{i=1}^{l} \gamma_i \epsilon_i$$

Obtaining the dual representation, requires taking the following steps:

$$\frac{\partial L}{\partial b} = 0 \rightarrow \sum_{i=1}^{l} \beta_i y_i = 0 \tag{18}$$

$$\frac{\partial L}{\partial \epsilon_i} = 0 \rightarrow C_l - \beta_i - \gamma_i = 0 \rightarrow 0 \leq \beta_i \leq C_l \tag{19}$$

With the above equations, we formulate the reduced Lagrangian as a function of only α and β as follows:

$$L^R(\alpha, \beta) = \frac{1}{2}\alpha^T \mathbf{K}(I + C_r\mathcal{L})\alpha - \alpha^T \mathbf{K}\mathbf{J}^T \mathbf{Y}\beta + \sum_{i=1}^{l} \beta_i \tag{20}$$

In the above equation, $\mathbf{J} = [\mathbf{I}\ \mathbf{0}]$ is a $l \times (l+u)$ matrix, $\mathbf{I}$ is the $l \times l$ identity matrix and Y is a diagonal matrix consisting of the labels of the labeled examples. We first take the derivative of L^R with respect to α and then set $\frac{\partial L^R(\alpha,\beta)}{\partial \alpha} = 0$. We have the following equation:

$$\mathbf{K}(I + C_r\mathcal{L})\alpha - \mathbf{K}\mathbf{J}^T \mathbf{Y}\beta = 0 \tag{21}$$

Accordingly, we obtain α^* by solving the following equation:

$$\alpha^* = (I + C_r\mathcal{L})^{-1}\mathbf{J}^T \mathbf{Y}\beta^* \tag{22}$$

Next, we obtain the dual problem in the form of a quadratic programming problem by substituting α back in the reduced Lagrangian function Eq. 20:

$$\beta^* = \text{argmax}_{\beta \in \mathbb{R}^l} \ -\frac{1}{2}\beta^T \mathbf{Q}\beta + \sum_{i=1}^{l} \beta_i \tag{23}$$

$$s.t. \ \sum_{i=1}^{l} \beta_i y_i = 0$$

$$0 \leq \beta_i \leq C_l \tag{24}$$

where $\beta = [\beta_1, ..., \beta_l]^T \in \mathbb{R}^l$ are the Lagrangian multipliers and $\mathbf{Q}$ is obtained as follows:

$$\mathbf{Q} = \mathbf{Y}\mathbf{J}\mathbf{K}(I + (C_r\mathcal{L})\mathbf{K})^{-1}\mathbf{J}^T\mathbf{Y} \tag{25}$$

We summarize the proposed semi-supervised framework in Algorithm 1. Our optimization problem is very similar to the standard optimization problem solved for SVMs, hence we use a standard optimizer for SVMs to solve our problem.

2.4 Computational Complexity

Here, we will explain the scalability of the algorithm in terms of big-O notation for both constituents of the proposed framework separately. For the first part of the approach, given a set of $\mathcal{A}$ cascades, and average number of $avg(\tau)$ users' actions (i.e., timestamps) in each cascade where $\tau \in \mathcal{A}$, the complexity of computing causality scores is $O(|\mathcal{A}|.(avg(\tau))^2)$ (note on average there are $(avg(\tau))^2$ pairs of users in each cascade). For the second part, i.e., learning the semi-supervised algorithm, the most time-consuming part is calculating the inverse of a dense Gram matrix which leads to $O((l+u)^3)$ complexity, where l and u are number of labeled and unlabeled instances [11].

Algorithm 1 Semi-Supervised Causal Inference for PSM detection (SemiPsm)

Input: $\{(x_i, y_i)\}_{i=1}^{l}$, $\{x_{l+i}\}_{i=1}^{u}$, $\mathcal{F}_1$, $\mathcal{F}_2$, C_l, C_r.
Output: Estimated function $f_\theta : \mathbb{R}^n \rightarrow \mathbb{R}$

1: Construct matrix **F** based on the causality-based features
2: Compute the corresponding Laplacian matrix $\mathcal{L}$.
3: Construct the Gram matrix over all examples using $\mathbf{K}_{ij} = k(x_i, x_j)$ where k is a kernel function.
4: Compute α^* and β^* using Eq. 22 and Eq. 23 and a standard QP solvers.
5: Compute function $f_\theta^*(x) = \sum_{i=1}^{l+u} \alpha_i^* \mathbf{K}(x, x_i)$

3 EXPERIMENTS

In this section we conduct experiments on a Twitter ISIS-related dataset and present results for several supervised and semi-supervised approaches. We first explain the dataset and provide some data analysis. Then, we will present the baseline methods. Finally, results and discussion are provided.

3.1 ISIS Twitter Dataset

We collect a dataset (Table 1) of 53 M ISIS related tweets/retweets in Arabic, from Feb 22, 2016 to May 27, 2016. The dataset has different fields including user ID, retweet ID, hashtags, content, posting time. The tweets were collected using 290 different hashtags such as #Terrorism and #StateOfTheIslamicCaliphate. We use a subset of this dataset which contains 35 K cascades of different sizes and durations. There are ~11 M tweets/retweets associated with the cascades. After pre-processing and removing duplicate users from cascades, cascades sizes (i.e. number of associated postings) vary between 20 to 9,571 and take from 10 seconds to 95 days to finish. The log-log distribution of cascades vs. cascade size and the cumulative distribution of duration of cascades are depicted in Figure 1.

Based on the content of tweets in our dataset, PSMs are terrorism-supporting accounts who have participated in viral cascades. We chose to use threshold $\theta = 100$ and take about 6 K viral cascades with at least 100 tweets/retweets. We demonstrate in Figure 2, the total number of users in each cascade suspended by Twitter. We note that he dataset does not have any underlying network. We only focus on the non-textual information in the form of an *action log*. We set $\phi = 0.5$ to select *key users*, i.e., we are looking for the users that participate in the cascades before the number of participants gets twice. After the data collection, we follow [41] and check through Twitter API whether users have been suspended (PSM) or are still active (normal). According to Table 1, less than 24% of the users in our dataset are PSM and the rests are normal.

3.2 Baseline Methods

We compare the proposed method SemiPsm against the following baseline methods. Note for all methods, we only report results when their best settings are used.

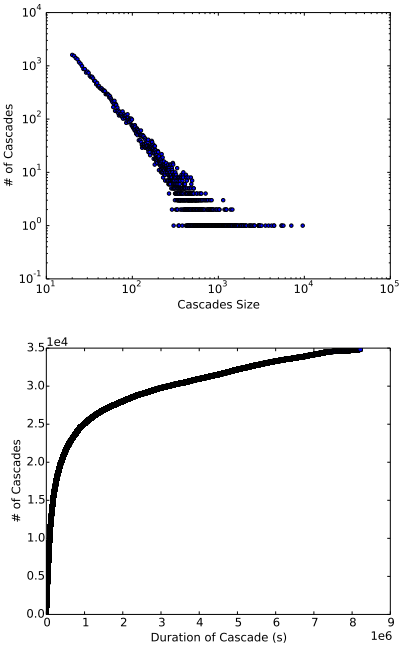

Figure 1: (Top) Log-log distribution of cascades vs. cascade size. (Bottom) Cumulative distribution of duration of cascades.

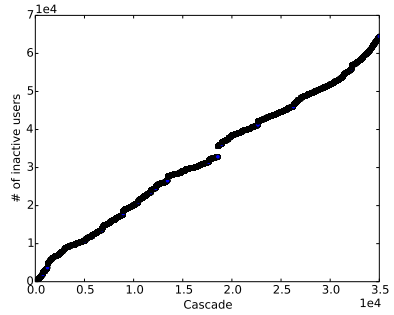

Figure 2: Total inactive users in each cascade.

Table 1: Description of the dataset.

Name	Value	
# of Cascades	35 K	
# of Viral Cascades	6,602	
# of Tweets/Retweets	10,823,168	
# of Users	PSM	Normal
	19,859	65,417

- **LabelSpreading (Rbf Kernel) [44].** This is a graph inference-based label spreading approach with radial basis function (RBF) kernel.
- **Label Spreading (Knn Kernel) [44].** Similar to the previous approach with K-nearest neighbor (KNN) kernel.
- **Lstm [30].** The word-level LSTM approach here is similar to the deep neural network models used for sequential word predictions. We adapt the neural network to a sequence classification problem where the inputs are the vector of words in each tweet and the output is the predicted label of the tweet. We first use the word2vec [31] embedding pre-trained from a set of tweets similar to the data representation in our Twitter dataset.
- **Account-Level (Rf Classifier) [30]** This approach uses the following features of the user profiles: *Statuses Count, Followers Count, Friends Count, Favorites Count, Listed Count, Default Profile, Geo Enables, Profile Uses Background Image, Verified, Protected.* We chose this method over Botometer [42] as it achieved comparable results with far less number of features ([42] uses over 1,500 features)(see also [21]). According to [30], we report the best results when Random Forest (RF) is used.
- **Tweet-Level (Rf Classifier) [30].** Similar to the previous baseline, this method uses only a handful of features extracted from tweets: *retweet count, reply count, favorite count, number of hashtags, number of URLs, number of mentions.* Likewise, we use RF as the classification algorithm.
- **SentiMetrix [39].** This approach was proposed by the top-ranked team in the DARPA Twitter Bot Challenge. We consider all features that we could extract from our dataset. Our features include tweet syntax (average number of hashtags, average number of user mentions, average number of links, average number of special characters), tweet semantics (LDA topics), and user behaviour (tweet spread, tweet frequency, tweet repeats). The proposed approach starts with a small seed set and propagates the labels. Since we have enough labeled data for the training part, we use Random Forest as the learning approach.
- **C2dc [2].** This approach uses time-decay causal community detection-based classification to detect PSM accounts [2]. For community detection, this approach uses Louvain algorithm.

3.3 Results and Discussion

All experiments were implemented in Python 2.7x and run on a machine equipped with an Intel(R) Xeon(R) CPU of 3.50 GHz with 200 GB of RAM running Linux. The proposed approach was implemented using CVXOPT[1] package. Furthermore, we split the whole dataset into 50% training and 50% test sets for all experiments. We report results in terms of F1-score in tables 2 and 3. For any approach that requires special tuning of parameters, we conducted grid search to choose the best set of parameters. Specifically, for the proposed approach, we set the penalty parameter as $C_l = 0.6$ and the regularization parameter $C_r = 0.2$, and used linear kernel. For LabelSpreading (Rbf), the default vale of $\gamma = 20$ was used and for LabelSpreading (Knn), number of neighbors was set to 5. Also, for random forest we used 200 estimators and the 'entropy' criterion was used. For computing k nearest neighbors in C2dc, we set $k = 10$.

[1] http://cvxopt.org/

Table 2: F1-score results of various methods on the labeled data. For semi-supervised learners, the size of the unlabeled data is fixed to 10% of the training set. The best performance is in bold.

Learner	F1-score
SemiPsm (Causal Features)	**0.94**
SemiPsm (Account-Level Features)	0.89
SemiPsm (Tweet-Level Features)	0.88
LabelSpreading (Knn/Causal Features)	0.89
LabelSpreading (Rbf/Causal Features)	0.88
Account-Level (Rf Classifier)	0.88
Tweet-Level (Rf Classifier)	0.82
SentiMetrix	0.54
Lstm	0.41
C2dc	0.4

Table 3: F1-score results of the semi-supervised approaches when causality-based features are used. Results are reported on different portions of the unlabeled data. The best performance is in bold.

	Percentage of Unlabeled Data				
	10%	**20%**	**30%**	**40%**	**50%**
SemiPsm	**0.94**	**0.93**	**0.91**	**0.9**	**0.88**
LabelSpreading (Knn)	0.89	0.88	0.87	0.85	0.81
LabelSpreading (Rbf)	0.88	0.86	0.85	0.82	0.80

Furthermore for LSTM, we preprocessed the individual tweets in line with the steps mentioned in [37]. Since the content of the tweets are in Arabic, we replaced special characters that were present in the text with their Arabic counterparts if they were present. We used word vectors of dimensions 100 and deployed the skip-gram technique for obtaining the word vectors where the input is the target word, while the outputs are the words surrounding the target words. To model the tweet content in a manner that uses it to predict whether an account is PSM or not, we used Long Short Term Memory (LSTM) models [25]. For the LSTM architecture, we used the first 20 words in the tokenized Arabic text of each tweet and use padding in situations where the number of tokens in a tweet are less than 20. We used 30 units in the LSTM architecture (many to one). The output of the LSTM layer was fed to a dense layer of 32 units with ReLU activations. We added dropout regularization following this layer to avoid overfitting and the output was then fed to a dense layer which outputs the category of the tweets.

We depict in Table 2 classification performance of all approaches on the labeled data. For the proposed framework SemiPsm, we examine three sets of features (1) causality-based features, (2) account-level features [30]; and (3) tweet-level features [30]. For the graph inference-based semi-supervised algorithms, i.e., LabelSpreading (Rbf) and LabelSpreading (Knn), we only report results where causality-based features are used as they achieved best performance with them. As it is observed from the table, the best results in terms of F1-score belong to SemiPsm where causality-based features are used. The runner-up is SemiPsm with account-level features and the next best approach is SemiPsm where tweet-level features are deployed. This clearly demonstrates the significance of using manifold regularization in the Laplacian semi-supervised framework over using other semi-supervised methods, LabelSpreading (Rbf) and LabelSpreading (Knn).

We further note that the supervised classifier Random Forest using both of the account-level and tweet-level features and the whole labeled dataset achieve worse or comparable results to the semi-supervised learners. The fact that obtaining several tweet and account-level features is not trivial and do not necessarily lead to the best classification performance, motivates us to use semi-supervised algorithms which use less number of labeled examples, and yet achieve competing performance. We also obtain an F1-score of 0.41 when LSTM is used– the poor performance of the this neural network model can be attributed to the raw Arabic text content. It suggests that the Arabic tokens as a representation might not be very informative about the category of accounts it has been generated from and some kind of weighting might be necessary before the LSTM module is used.

Also, Table 3 shows the classification performance of the semi-supervised approaches with causality-based features. The results are achieved using different portions of the unlabeled data, i.e., {10%, 20%, 30%, 40%, 50%} of the training set. As it is seen in the table, SemiPsm achieves the best performance on different portions of the unlabeled data compared to the other semi-supervised learners, while performances of all approaches deteriorate with increasing the percentage of the unlabeled data. Furthermore, SemiPsm still outperforms all other supervised methods as well as Lstm and C2dc when up to 50% of the data has been made unlabeled.

Observations. Overall, this paper makes the following observations:

- Among the semi-supervised learners used in this study, SemiPsm achieves the best classification performance suggesting the significance of using unlabeled instances in the form of manifold regularization. Manifold regularization is shown effective in boosting the classification performance, with three different sets of features confirming this.
- Causality-based features achieve the best performance via both Laplacian and graph inference-based semi-supervised settings. This lies at the inherent property of the causality-based features– they are designed to show whether or not user i exerts a causal influence on j. This is effective in capturing PSMs as they are key users in making a message viral.
- Compared to the supervised methods Account-Level (Rf) and Tweet-Level (Rf), semi-supervised learners achieve either comparable or best results, suggesting promising results with less number of labeled examples.
- Among the supervised methods Account-Level (Rf) and Tweet-Level (Rf), the former achieves higher F1-score indicating that account-level features are more useful in boosting the performance, although they are harder to obtain [30].
- Semi-supervised learners achieve best or comparable results with supervised learners, even with up to 50% of the data made unlabeled. This clearly shows the superiority of using unlabeled examples over labeled ones.

4 RELATED WORK

The explosive growth of the Web has raised numerous security and privacy issues. Mitigating these concerns has been studied from several aspects [1, 3, 5–7, 9, 13, 14, 18]. Our work is related to a number of research directions. Below, we will summarize some of the state-of-the-art methods in each category while highlighting their differences with our work.

Identifying PSM accounts. Compared to [34] which uses causal inference to detect PSM accounts, our work utilizes time-decay causal inference (using sliding-time window) which allows for early detection of PSM. In contrast to [2] where a causal community detection algorithm is proposed to leverage communities of PSM accounts in order to achieve higher performance, our work proposes a semi-supervised causal inference algorithm that achieves reasonable performance using less labeled data by utilizing unlabeled data.

Social Spam/Bot Detection. Recently, DARPA organized a Twitter bot challenge to detect "influence bots" [39]. Among the participants, the work of [14], used similarity to cluster accounts and uncover groups of malicious users. The work of [42] presented a supervised framework for bot detection which uses more than thousands features. In a different attempt, the work of [24] studied the problem of spam detection in Wikipedia using different spammers behavioral features. There also exist some studies in the literature that have addressed (1) differences between humans and bots [16], (2) different natures of bots [42] or (3) differences between bots and human trolls [13]. For example the work of [16] conducted a series of measurements in order to distinguish humans from bots and cyborgs, in term of tweeting behavior, content, and account properties. To do so, they used more than 40 million tweets posted by over 500 K users. Then, they performed analysis and find groups of features that are useful for classifying users into human, bots and cyborgs. They concluded that entropy and certain account properties can be very helpful in differentiating between those accounts. In a different attempt, some other studies have tried to differentiate between several natures of bots. For instance, in the work of [42], authors performed clustring analysis and revealed specific behavioral groups of accounts. Specifically, they identified different types of bots such as *spammers*, *self promoters*, and *accounts that post content from connected applications*, using manual investigation of samples extracted from clusters. Their cluster analysis emphasized that Twitter hosts a variety of users with diverse behaviors; that is in some cases the boundary between human and bot users is not sharp, i.e. some account exhibit characteristics of both.

Also, the work of [13], uses Twitter data to quantify the impact of Russian trolls and bots on amplifying polarizing and anti-vaccine tweets. They first used the Botometer API to assign bot probabilities to the users in the dataset and divided the whole dataset into 3 categories: those with scores less than 20% (very likely to be human), between 20% and 80% (e.g., cyborgs with uncertain provenance) and above 80% (high likely to be bots). Then, they posed two research questions: (1) Are bots and trolls more likely to tweet about vaccines?, and (2) Are bots and trolls more likely to tweet polarizing and anti-vaccine content? Their analysis demonstrated that Twitter bots and trolls significantly impact on online discussion about vaccination and this differs by account type. For example, Russian trolls and bots post content about vaccination at higher rates compared to an average user. Also, according to this study, troll accounts and content polluters (e.g., dissemination of malware, unsolicited commercial content, etc.) post anti-vaccine tweets 75% more than average users. In contrast, spambots which can be easily distinguished from humans, are less likely to promote anti-vaccine messages. Their closing remarks suggest strongly that distinguishing between malicious actors (bots, trolls, cyborgs, and human users) is difficult and thus anti-vaccine messages may be disseminated at higher rates by a combination of these malicious actors.

In contrast to the above works, our work does not deploy any extra information (e.g., user-related attributes or network-based features) other than users' actions (i.e., cascade with timestamps). It is also worthwhile to note that most of the existing well-known bot detection algorithms such as Botometer [19] leverage over one thousand features in order to detect high-likely bots.

Fake News Identification. A growing body of research is addressing the impact of bots in manipulating political discussion, including the 2016 U.S. presidential election [36] and the 2017 French election [20]. For example, [36] analyzes tweets following recent U.S. presidential election and found evidences that bots played key roles in spreading fake news.

Identifying Instigators. There are some work on instigator detection [22, 33] and outbreak prediction [18]. In [29], authors performed classification to detect users who adopt popular items. In [45], authors designed an approach for information source detection and in particular initiator of a cascade. Our work is focused on a set of users who *might* or *might not* be initiators. Our work is different from these works since we leverage causality analysis to detect causes of popularity of messages that go viral.

Extremism and Water Armies Detection. The work of [26] designed a behavioral model to detect extremists. Authors in [12] performed iterative vertex clustering and classification to identify Islamic Jihadists on Twitter. The works of [15, 43] also used user behavioral and domain-specific attributes to detect water armies. Our work also differs from these works as we do not use any features such as network/user attributes.

Causal Reasoning. As opposed to [27, 28, 38] which deal with preconditions as single atomic propositions, we use rules with preconditions of more than one atomic propositions.

5 CONCLUSION

We presented a semi-supervised Laplacian SVM to detect PSM users in social media who are promoters of misinformation spread. We cast the problem of identifying PSMs as an optimization problem and introduced a Laplacian semi-supervised SVM via utilizing unlabeled examples through manifold regularization. We examined different sets of features extracted from users activity log (in the form of cascades of retweets) as regularization terms: (1) causality-based features; and (2) LSTM-based features. Our causality-based features were built upon *Suppes' theory of probabilistic causation*. The LSTM-based features were extracted via LSTM which has shown promising results for different tasks in the literature.

In future, we would like to replicate the study by feeding other sets of features such as time-series features and those extracted

using LSTM to the semi-supervised framework. Also, we plan to investigate other forms of causality inferences and other regularization terms to seek if we can further improve the classification performance by distinguishing between different types of PSMs.

6 ACKNOWLEDGMENT

Some of the authors are supported through the DoS and DoD Minerva program.

REFERENCES

[1] Hamidreza Alvari, Soumajyoti Sarkar, and Paulo Shakarian. 2019. Detection of Violent Extremists in Social Media. *IEEE Conference on Data Intelligence and Security* (2019).

[2] Hamidreza Alvari, Elham Shaabani, and Paulo Shakarian. 2018. Early Identification of Pathogenic Social Media Accounts. *IEEE Intelligent and Security Informatics* (2018).

[3] Hamidreza Alvari, Paulo Shakarian, and JE Kelly Snyder. 2016. A non-parametric learning approach to identify online human trafficking. In *2016 IEEE Conference on Intelligence and Security Informatics (ISI)*. IEEE, 133–138.

[4] Hamidreza Alvari, Paulo Shakarian, and JE Kelly Snyder. 2017. Semi-supervised learning for detecting human trafficking. *Security Informatics* 6, 1 (2017), 1.

[5] Ghazaleh Beigi, Ruocheng Guo, Alexander Nou, Yanchao Zhang, and Huan Liu. 2019. Protecting User Privacy: An Approach for Untraceable Web Browsing History and Unambiguous User Profiles. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*. ACM, 213–221.

[6] Ghazaleh Beigi, Mahdi Jalili, Hamidreza Alvari, and Gita Sukthankar. 2014. Leveraging community detection for accurate trust prediction. (2014).

[7] Ghazaleh Beigi and Huan Liu. 2018. Privacy in social media: Identification, mitigation and applications. *arXiv preprint arXiv:1808.02191* (2018).

[8] Ghazaleh Beigi and Huan Liu. 2018. Similar but different: Exploiting usersâĂŹ congruity for recommendation systems. In *International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction and Behavior Representation in Modeling and Simulation*. Springer, 129–140.

[9] Ghazaleh Beigi, Kai Shu, Yanchao Zhang, and Huan Liu. 2018. Securing Social Media User Data-An Adversarial Approach. *Proceedings of the 29th on Hypertext and Social Media* (2018), 156–173.

[10] Misha Belkin, Partha Niyogi, and Vikas Sindhwani. 2005. On manifold regularization.. In *AISTATS*. Citeseer.

[11] Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani. 2006. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of machine learning research* 7, Nov (2006), 2399–2434.

[12] Matthew C Benigni, Kenneth Joseph, and Kathleen M Carley. 2017. Online extremism and the communities that sustain it: Detecting the ISIS supporting community on Twitter. *PloS one* (2017).

[13] David A Broniatowski, Amelia M Jamison, SiHua Qi, Lulwah AlKulaib, Tao Chen, Adrian Benton, Sandra C Quinn, and Mark Dredze. 2018. Weaponized health communication: Twitter bots and Russian trolls amplify the vaccine debate. *American journal of public health* 108, 10 (2018), 1378–1384.

[14] Qiang Cao, Xiaowei Yang, Jieqi Yu, and Christopher Palow. 2014. Uncovering Large Groups of Active Malicious Accounts in Online Social Networks. In *CCS*.

[15] Cheng Chen, Kui Wu, Srinivasan Venkatesh, and Xudong Zhang. 2011. Battling the Internet Water Army: Detection of Hidden Paid Posters. *CoRR* (2011).

[16] Zi Chu, Steven Gianvecchio, Haining Wang, and Sushil Jajodia. 2012. Detecting automation of twitter accounts: Are you a human, bot, or cyborg? *IEEE Transactions on Dependable and Secure Computing* 9, 6 (2012), 811–824.

[17] Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. *Machine learning* 20, 3 (1995), 273–297.

[18] Peng Cui, Shifei Jin, Linyun Yu, Fei Wang, Wenwu Zhu, and Shiqiang Yang. 2013. Cascading Outbreak Prediction in Networks: A Data-driven Approach. In *KDD*.

[19] Clayton Allen Davis, Onur Varol, Emilio Ferrara, Alessandro Flammini, and Filippo Menczer. 2016. Botornot: A system to evaluate social bots. International World Wide Web Conferences Steering Committee.

[20] Emilio Ferrara. 2017. Disinformation and social bot operations in the run up to the 2017 French presidential election. (2017).

[21] Emilio Ferrara, Onur Varol, Clayton Davis, Filippo Menczer, and Alessandro Flammini. 2016. The rise of social bots. *Commun. ACM* (2016).

[22] Huang Chung-Yuan Fu, Yu-Hsiang and Chuen-Tsai Sun. 2015. Identifying Super-Spreader Nodes in Complex Networks. *Mathematical Problems in Engineering* (2015).

[23] Amit Goyal, Francesco Bonchi, and Laks V.S. Lakshmanan. 2010. Learning Influence Probabilities in Social Networks. In *WSDM*.

[24] Thomas Green and Francesca Spezzano. 2017. Spam Users Identification in Wikipedia Via Editing Behavior. *ICWSM* (2017).

[25] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[26] Jytte Klausen, Christopher Marks, and Tauhid Zaman. 2016. Finding Online Extremists in Social Networks. *CoRR* abs/1610.06242 (2016).

[27] Samantha Kleinberg. 2011. A Logic for Causal Inference in Time Series with Discrete and Continuous Variables. In *IJCAI*.

[28] Samantha Kleinberg and Bud Mishra. 2009. The temporal logic of causal structures. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*. AUAI Press, 303–312.

[29] Takuya Konishi, Tomoharu Iwata, Kohei Hayashi, and Ken-Ichi Kawarabayashi. 2016. Identifying Key Observers to Find Popular Information in Advance. In *IJCAI*.

[30] Sneha Kudugunta and Emilio Ferrara. 2018. Deep Neural Networks for Bot Detection. *arXiv preprint arXiv:1802.04289* (2018).

[31] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. 3111–3119.

[32] Judea Pearl. 2009. *Causality: Models, Reasoning and Inference* (2nd ed.). Cambridge University Press, New York, NY, USA.

[33] Sen Pei, Lev Muchnik, JosÃľ S. Andrade Jr., Zhiming Zheng, and HernÃąn A. Makse. 2014. Searching for superspreaders of information in real-world social media. *CoRR* (2014).

[34] E. Shaabani, R. Guo, and P. Shakarian. 2018. Detecting Pathogenic Social Media Accounts without Content or Network Structure. In *IEEE Conference on Data Intelligence and Security*.

[35] Elham Shaabani, Ashkan Sadeghi-Mobarakeh, Hamidreza Alvari, and Paulo Shakarian. 2019. An End-to-End Framework to Identify Pathogenic Social Media Accounts on Twitter. *IEEE Conference on Data Intelligence and Security* (2019).

[36] Chengcheng Shao, Giovanni Luca Ciampaglia, Onur Varol, Alessandro Flammini, and Filippo Menczer. 2017. The spread of fake news by social bots. (2017).

[37] Abu Bakr Soliman, Kareem Eissa, and Samhaa R El-Beltagy. 2017. Aravec: A set of arabic word embedding models for use in arabic nlp. *Procedia Computer Science* 117 (2017), 256–265.

[38] Andrew Stanton, Amanda Thart, Ashish Jain, Priyank Vyas, Arpan Chatterjee, and Paulo Shakarian. 2015. Mining for Causal Relationships: A Data-Driven Study of the Islamic State. *CoRR* (2015).

[39] V. S. Subrahmanian, A. Azaria, S. Durst, V. Kagan, A. Galstyan, K. Lerman, L. Zhu, E. Ferrara, A. Flammini, and F. Menczer. 2016. The DARPA Twitter Bot Challenge. (2016).

[40] Patrick Suppes. 1970. A Probabilistic Theory of Causality. (1970).

[41] Kurt Thomas, Chris Grier, Dawn Song, and Vern Paxson. 2011. Suspended accounts in retrospect: an analysis of twitter spam. In *ACM SIGCOMM conference on Internet measurement conference*.

[42] Onur Varol, Emilio Ferrara, Clayton A Davis, Filippo Menczer, and Alessandro Flammini. 2017. Online human-bot interactions: Detection, estimation, and characterization. *ICWSM* (2017).

[43] Kun Wang, Yang Xiao, and Zhen Xiao. 2014. Detection of internet water army in social network.

[44] Dengyong Zhou, Olivier Bousquet, Thomas N Lal, Jason Weston, and Bernhard Schölkopf. 2004. Learning with local and global consistency. In *Advances in neural information processing systems*. 321–328.

[45] Kai Zhu and Lei Ying. 2016. Information Source Detection in the SIR Model: A Sample-path-based Approach. *IEEE/ACM Trans. Netw.* 24, 1 (2016).