Weakly Supervised Extraction of Computer Security Events from Twitter

Alan Ritter
Computer Science and Engineering
The Ohio State University*
Columbus, OH
ritter.1492@osu.edu

Evan Wright
Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA
evanw@cmu.edu

William Casey
Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA
wcasey@cert.org

Tom Mitchell
Machine Learning Department
Carnegie Mellon University
Pittsburgh, PA
tom.mitchell@cmu.edu

ABSTRACT

Twitter contains a wealth of timely information, however staying on top of breaking events requires that an information analyst constantly scan many sources, leading to information overload. For example, a user might wish to be made aware whenever an infectious disease outbreak takes place, when a new smartphone is announced or when a distributed Denial of Service (DoS) attack might affect an organization's network connectivity. There are many possible event categories an analyst may wish to track, making it impossible to anticipate all those of interest in advance. We therefore propose a weakly supervised approach, in which extractors for new categories of events are easy to define and train, by specifying a small number of seed examples. We cast seed-based event extraction as a learning problem where only positive and unlabeled data is available. Rather than assuming unlabeled instances are negative, as is common in previous work, we propose a learning objective which regularizes the label distribution towards a user-provided expectation. Our approach greatly outperforms heuristic negatives, used in most previous work, in experiments on real-world data. Significant performance gains are also demonstrated over two novel and competitive baselines: semi-supervised EM and one-class support-vector machines. We investigate three security-related events breaking on Twitter: DoS attacks, data breaches and account hijacking.

A demonstration of security events extracted by our system is available at:
`http://kb1.cse.ohio-state.edu:8123/events/hacked`

*This Work was conducted at Carnegie Mellon

WWW 2015, May 18–22, 2015, Florence, Italy.
ACM 978-1-4503-3469-3/15/05.
http://dx.doi.org/10.1145/2736277.2741083 .

Categories and Subject Descriptors

I.2.7 [**Natural Language Processing**]: Language parsing and understanding; H.2.8 [**Database Management**]: Database applications—*data mining*

General Terms

Algorithms, Experimentation

1. INTRODUCTION

Social media presents a rich and timely source of information on events taking place in the world, enabling applications such as earthquake detection [41] or identifying the location of missing persons during natural disasters [29]. Previous work on event extraction has relied on large amounts of labeled data, or taken an open-domain approach [38, 2] in which general events are extracted without a specific focus. Often an information analyst might be interested in tracking a very specific type of event, for instance Denial of Service (Denial of Service (DoS)) attacks and might not have time or expertise to build an information extraction system from scratch in response to emerging incidents. To address this challenge we introduce an approach for rapidly training automatic extractors for the raw Twitter stream. As a proof of concept, we study three specific computer-security event categories: DoS attacks, data breaches, and account hijacking.

Our approach requires an analyst simply provide 10-20 historical seed examples of the event category of interest, for example seed instances of DoS attack events might include (*Spamhaus*, 3/18/2013). A bag of tweets which mention each seed event are then gathered, for instance Spamhaus is mentioned in the following tweet written on 3/18/2013:

> "The *Spamhaus* Project is currently under a major DDoS attack"

These seed events are used as training examples to automatically detect new events from a realtime Twitter stream.

Challenge: Data Sparsity Although many tweets mention computer security events, only a tiny proportion of the overall message volume is relevant. Because only a 1% sample of the Twitter stream is available through the public

API, without focusing the data collection, virtually no security events would be found in available data. To address this challenge we simply track a user-provided keyword associated with each event, for example we tracked *ddos* for DoS attacks, *hacked* for account hijacking, and *breach* for data breach events. When tracking relevant keywords, the Twitter API allows us to retrieve roughly the same total volume of data, however a much larger proportion is relevant to the security-related events of interest. Of course not all tweets mentioning a relevant keyword will describe the events of interest, we therefore leverage the seed events previously mentioned, to train a weakly supervised extractor.

Opportunity: Redundancy in Social Media While there has been much previous work on weakly supervised extraction of static relationships between entities [6, 5, 1], there have been few efforts focused on seed-based *event* extraction. Part of the reason is the reliance of weakly supervised learning methods on redundancy - many sentences on the web are likely to mention context independent relations, such as the headquarters of a company, however most events are only mentioned in a handful of articles written around the time of the event, making learning from a few seed instances very challenging [16, 36]. In the meantime, social networking websites such as Twitter have become an important complementary source of realtime information. When important events take place, many users independently turn to microblogs to share information, resulting in a huge number of redundant messages describing each significant event, and providing an opportunity to collect large amounts of training data for weakly supervised event extraction.

We cast *seed-based event extraction* as a unique semi-supervised learning problem in which only positive and unlabeled examples are available [9, 21, 19, 10]. A new approach to learning with positive and unlabeled data is introduced, which regularizes the label distribution over unlabeled examples towards a user-provided expectation [24, 25, 11].

A huge number of security events are reported in social media every day. These include everything from average users complaining about their email accounts being hijacked to massive data breaches involving international corporations. Many of these events are not sufficiently significant to be reported in the news, though they may still be of interest to a computer security analyst. The volume of security-related messages written on social media is simply too large to constantly monitor, leading to a situation of information overload.

Of course misinformation and rumors are prevalent in social media [35], we therefore preserve provenance for the extracted events, making it easy to display information sources to an analyst who can investigate further and make judgments about reliability.

This paper makes the following contributions:

- We demonstrate that social media is a valuable resource for information on security related events.
- We present a novel approach to extracting focused events from Twitter which requires only minimal supervision for each new event category.
- We cast seed-based weakly supervised information extraction as a learning problem with positive and unlabeled examples [9, 21] and propose a new approach which regularizes the label distribution over unlabeled examples towards a user-specified expectation of the label distribution for the keyword.

Given:

- An event type E of interest (e.g., Distributed denial of service attacks)
- A keyword or keyphrase, K, associated with this event type (e.g. "DDoS")
- A named entity recognizer which can identity entities mentioned in a tweet [37, 22, 20]
- A set of positive seed examples $e_1, e_2, \ldots, e_n$ of historical instances of E, where each seed example is represented by an entity involved in the event, plus the date of the event, for example: ($AT\&T$, 2012/08/15)
- Access to the Twitter search interface to gather tweets mentioning the seed examples

Output:

- An extractor that can be applied to the Twitter stream filtered by keywords K to identify new instances of event type E as they are mentioned.

Figure 1: Summary of the weakly supervised event extraction problem.

2. WEAK SUPERVISION FOR EVENT EXTRACTION FROM TWITTER

Using a traditional annotate-train-test approach to information extraction [12, 17, 40] is problematic for our scenario, because significant time and effort would be required to develop extractors for each new event category of interest. In traditional IE, a large corpus of individual event triggers and their arguments must first be annotated in context. These annotations can then be used to train supervised machine learning models to extract new event instances. This approach has been quite successful, but requires a substantial annotation effort for each new type of event to be recognized.

A wide variety of different event types might interest an analyst, making it difficult or impossible to anticipate those which could be important and build extractors for them a-priori. This motivated us to investigate a *weakly supervised* approach to event extraction from Twitter in which new event categories are quickly defined by providing a small set of seed instances and relevant keywords.

The seed examples are used to train a classifier which is applied to a *candidate event*, to determine whether it describes a new instance of event type E. Candidate events are collected automatically by gathering tweets that match keywords of interest K, were written on the target date and mention a specific entity.

We now describe in detail the inputs and outputs of our algorithm. A summary is presented in Figure 1. We show how weakly supervised event extraction leads to a learning problem in which only positive and unlabeled examples

Figure 2: Example of gathering mentions of seed events. By querying for the event-type keyword, *hacked* and entity *Palin* in tweets on the event date ("sept. 8, 2008") we are able to obtain many high-quality mentions for the event.

are available [9, 21]. We propose a new approach to learning with positive and unlabeled data based on expectation regularization [24] in Section 3 which is experimentally compared against a variety of baselines in Section 6.

2.1 Event Representation

We define an event e as an (ENTITY, DATE) tuple, and a *mention* of e, m_e, to be any tweet which contains a reference to the ENTITY and is written on the specified DATE. Features for the event, are extracted from its mentions: $x_e = f(\{m_{e'}|e' = e\})$, which can be used to estimate the probability that the event belongs to category E,

$$p_{\theta^E}(y_e = 1|x_e) = \frac{1}{1 + e^{-\theta_E \cdot x_e}}$$

according to some parameters for the category, θ_E.

2.2 Seed Instances

To define a new event category, an information analyst provides a set of 10-20 seed events, represented as (ENTITY, DATE) tuples. Our seed lists for DoS attacks, account hijacking and data breach events are presented in tables 1, 2 and 3.

To gather mentions of a seed event (from which features can be extracted) we select tweets which reference the entity, and were written on the event's date. This can be accomplished by querying the Twitter search API[1] with the entity and keyword in addition to the specified date (see Figure 2).

Victim	Date	# Mentions
spamhaus	2013/03/18	26
soca	2011/06/20	89
etrade	2012/01/05	76
interpol	2012/02/29	45
ustream	2012/05/09	911
virgin media	2012/05/08	15
pirate bay	2012/05/16	2,265
demonoid	2012/07/27	182
att	2012/08/15	2,743
sweden	2012/09/03	28
godaddy	2012/09/10	849
github	2013/07/29	102
reddit	2013/04/19	2,042
cia	2011/06/15	36
paypal	2010/12/10	57

Table 1: Seed instances for DDoS attacks

Victim	Date	# Mentions
associated press	2013/04/23	3,846
reuters	2012/08/05	607
us marines	2013/09/03	97
sarah palin	2008/09/18	5,460
mitt romney	2012/06/05	886
cnn	2014/01/23	597
justin bieber	2012/03/27	348
mutunga	2013/09/27	19
yes scotland	2013/08/20	212
zuckerberg	2013/08/18	229

Table 2: Seed instances for account hijacking attacks

2.3 Extracting Candidate Events

New candidate events are extracted from tweets gathered using the Twitter Streaming API[2]. As mentioned in Section 1 we track keywords associated with each event to avoid sparsity. In our experiments, we track the keywords: *hacked* for account hijacking, *ddos* for denial of service attacks, and *breach* for data breaches.

As tweets are gathered, we extract named entities using a Twitter-tuned NLP pipeline [37]. These extracted entities are then combined with the date on which the tweet was written to form new candidate event instances. Not every candidate will correspond to an event of interest. For instance, many tweets which mention the keyword *ddos*, are promoting products to mitigate Disributed Denial of Service (DDoS) attacks, or talk in general about the category of the event without mentioning a specific instance, e.g.:

> "NTP Amplification **DDoS** Attacks increased over the last few months."

or similarly for the *breach* keyword:

[1] `https://twitter.com/search-home`
[2] `https://dev.twitter.com/docs/streaming-apis`

Feature Category	Sample Feature	Event Category
keyword-context-left	`security` **breach**	Data Breach
victim-context-right	**X** `admits`	Data Breach
victim-context-right	**X** `data breach affecting`	Data Breach
keyword-context-both	`DT` **ddos** `attack`	DDoS
keyword-context-left	`NNP NNP hit IN` **ddos**	DDoS
victim-context-right	**X** `getting ddos'd`	DDoS
victim-context-both	`PRP hacked` **X** `POS account`	Account Hijacking
keyword-context-left	`POS email was` **hacked**	Account Hijacking
victim-context-right	**X** `POS NNP account`	Account Hijacking

Table 4: Examples of high-weight features. Context words other than nouns and verbs are replaced with their part-of-speech tags for better generalization.

Victim	Date	# Mentions
citi	2011/06/09	62
sony	2011/04/23	11
adobe	2013/10/03	1,192
evernote	2013/03/02	3,175
facebook	2013/06/21	1,026
steam	2011/11/07	49
zappos	2012/01/16	2,617
heartland	2009/01/20	28
utah	2013/03/29	12
rsa	2011/03/17	13
nyseg	2012/01/23	18

Table 3: Seed instances for account data breach

"**Breach** of ceasefire in central Syria delays aid convoy"

We therefore need to leverage the provided seed instances in order to determine which candidate event extractions fit the category and filter out distractors.

2.4 Classifying Unlabeled Events

The seed-based approach to event extraction described so far provides a natural interface for an information analyst to define new events, however it leads to a somewhat atypical learning problem. Instead of being provided a representative sample of positive and negative examples of the category of interest, we only have a small set of seed instances, in addition to a large sample of unlabeled events. Rather than heuristically assuming unlabeled examples are negative, we propose a learning objective which regularizes the label distribution towards a user-provided expectation in Section 3. In Section 6, we demonstrate that our approach outperforms a number of competitive baselines including semi-supervised EM and one-class support vector machines.

2.5 Feature Design

We define two sets of binary features as a basis for identifying security-related events. The first consists of a window of contextual words and parts of speech surrounding the entity which represents the candidate victim of the attack. The second feature set is composed of contextual features surrounding the tracked keyword. Context windows of 1-4 words to the left and right of the target are used. All words other than common nouns and verbs are represented by their part of speech tags for better generalization. Part of speech tagging is performed using a tagger which is adapted to Twitter [37]. Representative examples of high-weight features from our data are presented in Table 4.

3. LEARNING TO EXTRACT EVENTS WITH POSITIVE AND UNLABELED DATA

In Section 2 the task of weakly supervised event extraction was cast as a semi-supervised learning problem in which only positive and unlabeled data are available [9]. Motivated by this view on the problem we discuss a number of relevant approaches which are empirically evaluated on our task of security-related event extraction in Section 6.

3.1 Baseline 1: Heuristically Labeled Negative Examples

Perhaps the simplest approach is to use a *non-traditional* classifier to distinguish positive from unlabeled examples. This approach was theoretically analyzed by Elkan and Noto [9]. They noted a classifier trained on positive and unlabeled examples is proportional in expectation to a traditional classifier trained on both positive and negative examples, under the assumption that the observed positives are randomly selected. The approach of heuristically assuming unlabeled examples are negative has been used in previous work on weakly supervised relation extraction [26].

As we demonstrate in Section 6, the assumption that unlabeled examples are negative is too strong for our task of event extraction for two reasons: Firstly, only a small sample of positive seeds are available which are not sampled from the underlying distribution; secondly, the unlabeled examples contain a relatively large proportion of positives, leading to a challenging learning problem.

3.2 Expectation Regularization

Rather than heuristically assuming all unlabeled events are negative, and maximizing their likelihood, instead we propose asking an expert to provide an estimate of the proportion of genuine security events within the unlabeled data. Inspired by recent work on semi-supervised learning, [24] we augment the likelihood term over the positive-only seed instances with a term that encourages the expectation over model predictions on unlabeled data, $\hat{p}_\theta$, to match the user-provided target expectation, $\tilde{p}$.

The objective function combines conditional log likelihood of the (positive-only) labeled data, with an expectation regularization term and L^2 regularization, and is defined as follows:

$$O(\theta) = \underbrace{\sum_i^N \log p_\theta(y_i|x_i)}_{\text{Log Likelihood}} - \underbrace{\lambda^U D(\tilde{p}||\hat{p}_\theta^{\text{unlabeled}})}_{\text{Label regularization}} - \underbrace{\lambda^{L^2} \sum_j w_j^2}_{L^2 \text{ regularization}} \quad (1)$$

Where the parametric form of $p_\theta(y|x)$ was presented in Section 2.1.

The expectation regularization term is defined as the KL divergence between the empirical expectation of the model's posterior predictions on unlabeled data, $\hat{p}_\theta$, and the user-provided target expectation, $\tilde{p}$:

$$D(\tilde{p}||\hat{p}_\theta) = \tilde{p} \log \frac{\tilde{p}}{\hat{p}_\theta} + (1-\tilde{p}) \log \frac{1-\tilde{p}}{1-\hat{p}_\theta}$$

The gradient of the objective function is simply the sum of the gradient for logistic regression with L^2 regularization, combined with the gradient for the expectation regularization term. The derivative of the KL divergence is as follows:

$$\frac{\partial}{\partial \theta_k} D(\tilde{p}||\hat{p}_\theta) = \frac{\partial}{\partial \theta_k} \tilde{p} \log \frac{\tilde{p}}{\hat{p}_\theta} + (1-\tilde{p}) \log \frac{1-\tilde{p}}{1-\hat{p}_\theta}$$

Starting with the first term:

$$\begin{aligned}
\frac{\partial}{\partial \theta_k} \tilde{p} \log \frac{\tilde{p}}{\hat{p}_\theta} &= \frac{\partial}{\partial \theta_k} \underbrace{\tilde{p} \log \tilde{p}}_{\text{constant with respect to } \theta} - \tilde{p} \log \hat{p}_\theta \\
&= -\frac{\partial}{\partial \theta_k} \tilde{p} \log \hat{p}_\theta \\
&= -\frac{\tilde{p}}{\hat{p}_\theta} \frac{\partial}{\partial \theta_k} \hat{p}_\theta \\
&= -\frac{\tilde{p}}{\hat{p}_\theta} \frac{1}{N} \sum_{i=1}^N \frac{\partial}{\partial \theta_k} p_\theta(y_i = 1|x_i) \\
&= -\frac{\tilde{p}}{\hat{p}_\theta} \frac{1}{N} \sum_{i=1}^N \underbrace{p_\theta(y_i = 1|x_i)\,(1 - p_\theta(y_i = 1|x_i))\, x_{i,k}}_{\text{sigmoid derivative}}
\end{aligned}$$

The derivation of the gradient for the second term is similar; putting these together we get:

$$\frac{\partial}{\partial \theta_k} D(\tilde{p}||\hat{p}_\theta) =$$

$$\frac{1}{N} \left(\frac{1-\tilde{p}}{1-\hat{p}_\theta} - \frac{\tilde{p}}{\hat{p}_\theta} \right) \sum_{i=1}^N p_\theta(y_i = 1|x_i)(1 - p_\theta(y_i = 1|x_i)) x_{i,k}$$

This gradient makes sense intuitively: If the empirical label distribution matches the user-provided target expectation, then $\hat{p}_\theta = \tilde{p}$ and the gradient is 0. If $\hat{p}_\theta > \tilde{p}$ or $\hat{p}_\theta < \tilde{p}$, then the second factor is either positive or negative accordingly to push the parameters θ up or down. Also note that the terms for each unlabeled example $p_\theta(y_i = 1|x_i)(1 - p_\theta(y_i = 1|x_i)) x_{i,k}$ will give more weight to uncertain cases, since the function $f(x) = x(1-x)$ has it's maximum at 0.5.

3.3 Baseline 2: Semi-Supervised EM with Constrained Class Priors

As a baseline we implemented a generative model based on Naïve Bayes which explicitly models unlabeled examples using latent variables [30]. We use EM to maximize likelihood, alternating between estimating the model's parameters, $\theta_k = p(x_k = 1|y)$, in the M-step, and updating the posterior distribution over unlabeled examples in the E-step, by applying Bayes' rule:

$$p_\theta(y_i|x_i) = \frac{p(x_i|y_i = 1)p(y_i = 1)}{p(x_i)}$$

Because only positive examples are observed, we constrain the prior distribution over positives to the same user-specified value as was used for label regularization, rather than re-estimating it from data in the M-step:

$$p(y_i = 1) = \tilde{p}$$

3.4 Baseline 3: One-Class SVMs

As an additional (novel) baseline for weakly supervised information extraction, we experimented with one-class SVMs [42], which are only trained on positive examples and ignore the unlabeled data. We used the one-class SVM implementation from LibSVM.[3] A linear kernel was used which is appropriate for our case due to the large number of features, where overfitting is a concern. We experimented with other kernel functions but found a linear kernel to have the best performance.

4. SECURITY RELATED EVENTS

Section 2 presented a generic framework for extracting targeted events from social media streams and section 3 presented a series of approaches for addressing the weakly supervised learning problem which it presented. Next we briefly define and describe the information security events we have investigated as a case study.

4.1 Denial of Service Attacks

The DoS is designed to deny the liveness properties (e.g. uptime) of a web service to other users. These attacks are most often accomplished by an agent who amplifies requests for a network service with no other intention but to saturate the service beyond some capacity of the resource(s) behind that device (e.g. bandwidth, processing or memory).

An Example: In March of 2013 an ongoing conflict between *Spamhaus* a large European spam blocking service and *Cyberbunker* a European internet service provider erupted into a massive and sustained DDoS attack. Beginning on March 18th of 2013 a distributed attack, that leveraged the Domain Name System (DNS) protocol to amplify the volume of the attack, was initialized from a set of hosts which requested DNS packets with a spoofed return addresses belonging to *Spamhaus*. During the attack (March 18th to 27th of 2013) significant side effects to internet users and services were experienced in Europe.

The use of social media to detect liveness properties of web services has been studied in [27], however we aim to take a step further to consider a specific cause of outage (i.e. DoS attack).

4.2 Data Breach

Data breach is an attack which implements sophisticated techniques to pilfer a collection of personal or digital credentials. The effects of a data breach if unmitigated may

[3] `http://www.csie.ntu.edu.tw/~cjlin/libsvm/`

result in fraudulent use of personal information; however, early detection may alert affected users to monitor for fraud and initialize preventative measures such as updating credentials. Data breach attacks my elicit signals from social media such as early discovery and warnings generated by affected users who discover fraud, further this may be useful to other users whose stolen personal information (e.g. credit card) has not yet been exploited.

An Example: On December 19, 2013, Target Brands Inc. announced that up to 40 million credit/debit cards of customers from Target stores during the time period November 27th 2013 to December 15th of 2013 may be affected by a databreach on their point of sale system. While stolen information may be used for fraudulent charges, the timely notification for affected customers allows for mitigation options.

4.3 Account Hijacking

Account hijacking may involve an intruder guessing, cracking, or using default passwords to gain unauthorized access to a user accounts or system privileges. Account hijacking usually focuses on the problem of determining an unknown user password by using techniques including brute force attacks, dictionary attacks (using frequently used passwords), Rainbow Tables [28] when hash values are available, or profiling an individual to boost other techniques.

An Example: On April 23, 2013, the Associated Press (AP) twitter account was hijacked and the intruders used the AP twitter account to falsely announce that the white house had been attacked and president Obama injured. The Dow Jones industrial average dropped immediately 143 points from 14,697 but quickly recovered. Phishing attacks (social engineering) on AP were alleged to have preceded the hijacked accounts. The Syrian Electronic Army (SEA) claimed responsibility for the intrusion. In addition to AP the hijackings of several high profile Twitter accounts have been reported [4] including: Financial Times (claimed by SEA), The Onion, The Guardian (claimed by SEA), North Korea (claimed by Anonymous), Burger King, Jeep, and Agence France-Presse (AFP).

5. DATA

5.1 Seed Events

To gather historical seed events the authors used targeted search queries, and several events they had in memory. The list of seeds is displayed in Tables 1, 2 and 3.

5.2 Unlabeled Events

Unlabeled event candidates were gathered by tracking keywords, and extracting named entity mentions as described in Section 2 using a set of NLP tools tuned to work on noisy Twitter data.[4] We gathered data over roughly a 1 month period from January 17 2014 until February 20. This resulted in roughly 14,610,000 raw tweets. Non-English tweets were filtered using `langid.py` [23], and named entities were extracted [37]. Events which are mentioned 3 or more times in the data were considered, that is: named entities mentioned in 3 or more tweets on the same day. This resulted in 4,014 extracted candidate account hijacking events, 570 candidate DDOS attacks and 1,738 data breaches.

[4] `https://github.com/aritter/twitter_nlp`

	Hijacking	DDOS	Data Breach
Logistic Regression	0.419	0.252	0.444
Expectation Reg.	**0.716**	**0.459**	**0.676**
Semi-Supervised EM	0.453	0.419	0.442
One-Class SVM	0.681	0.436	0.641

Table 5: Area under precision recall curve comparing various methods for learning with positive and unlabeled data. Our expectation-regularization based approach significantly outperforms all baselines in each event category according to a paired t-test with p value less than 0.05.

5.3 Features

We extract binary features from the bag of tweets associated with each event as described in Section 2.5. A sample of high-weight features inferred using our approach are listed in Table 4. We removed any features which did not appear in association with at least two distinct entities in the data. In total there are 52,995 features for account hijacking, 3,790 features for DoS attacks and 11,271 for data breach.

6. EVALUATION

To evaluate the ability of our models to extract new events, we manually annotated a random sample of candidate events for each category. We then compared the model's predictions against human judgments in order to measure precision and recall.

For each category one of the authors manually annotated 200 randomly sampled instances. In cases where the judgment was not immediately clear, the annotator followed any links in the tweets and used Google searches to determine whether the event did indeed correspond to an instance of the category.

We set the target expectation in each experiment to a value slightly larger than 0.5, and the L^2 regularization to 100. Following Mann et. al. [24], λ^U is set to 10 times the number of labeled examples.

6.1 Results

Precision and recall curves for account hijacking, DoS attacks and data breach are presented in Figures 3, 4, and 5. Several observations can be made: First the baseline of heuristically assuming unlabeled examples are negative results in generally poor performance. This is due to the unique nature of our problem: our seeds are not randomly sampled from the underlying distribution of positive examples, and the unlabeled examples contain a relatively high proportion of positives. While the one-class SVM presents a very strong baseline for this problem, our approach based on expectation regularization achieves significantly higher area under the precision-recall curve according to a paired t-test. Area under the curve for each method and event category is presented in Table 5. Examples of high-confidence extractions in each category are presented in Table 6.

6.2 Independent Validation with Computer Network Measurements

We consider validity by independently measuring the computer network traffic associated with one anecdotal confirmation of a discovered security event by our classification

Victim	Date	Category	Sample Tweet
namecheap	Feb-20-2014	DDoS	My site was down due to a DDoS attack on NameCheap's DNS server. Those are lost page hits man...
bitcoin	Feb-12-2014	DDoS	Bitcoin value dramatically drops as massive #DDOS attack is waged on #Bitcoin http://t.co/YdoygOGmhv
europe	Feb-20-2014	DDoS	Record-breaking DDoS attack in Europe hits 400Gbps.
barcelona	Feb-18-2014	Account Hijacking	Lmao, the official Barcelona account has been hacked.
adam	Feb-16-2014	Account Hijacking	@adamlambert You've been hacked Adam! Argh!
dubai	Feb-09-2014	Account Hijacking	Dubai police twitter account just got hacked!
maryland	Feb-20-2014	Data Breach	SSNs Compromised in University of Maryland Data Breach: https://t.co/j69VeJC4dw
kickstarter	Feb-15-2014	Data Breach	I suspect my card was compromised because of the Kickstarter breach. It's a card I don't use often but have used for things like that.
tesco	Feb-14-2014	Data Breach	@directhex @Tesco thanks to the data breach yesterday it's clear no-one in Tesco does their sysadmin housekeeping!

Table 6: Example high-confidence events extracted using our system.

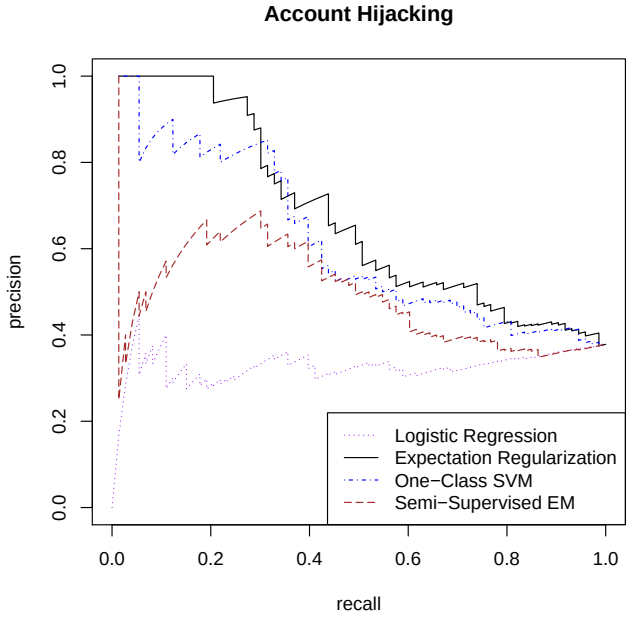

Figure 3: Precision and recall comparing various approaches to extracting events using only positive seeds and unlabeled data.

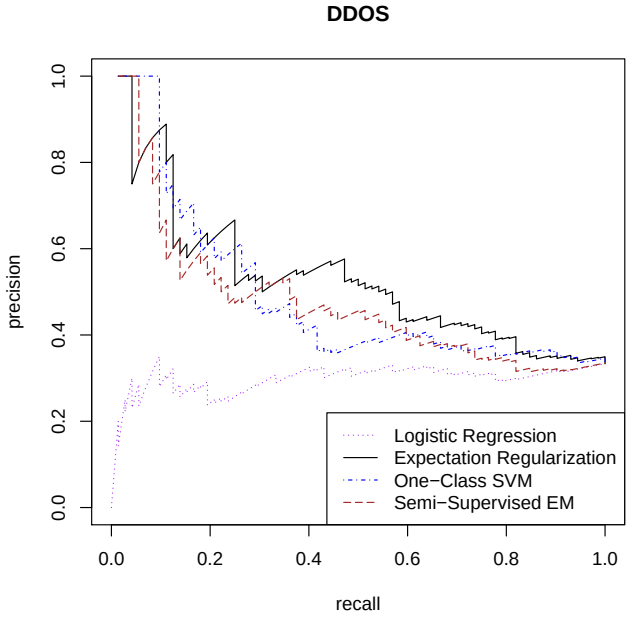

Figure 4: Precision and recall comparing various approaches to extracting events using only positive seeds and unlabeled data.

system. Our system identified *weasyl.com* in a DDoS attack on January 18. For example, without prior knowledge of *weasyl.com* or the attack date the classifier indicates the following messages as a DDoS event:

> "Over the last 24 hours, Weasyl has been subject to a few DDoS attacks on the site."

A popular data source to summarize large quantities of network data and network properties is network flow data, or *netflow* [8]. This data allows us to monitor network traffic and retrospectively identify events that occurred. Netflow data is gathered on routers or special-purposed devices, to account network activity. Netflow is valuable to understand traffic volume and actors as well as retrospectively accomplish network forensics.

Using netflow we measure the traffic between a large corporate network and *weasyl.com* during a nine day time period. Figure 6 shows the number of unique devices sending RST packets per hour to *weasyl.com*. The RST packet is the device's response to a malformed communication channel, including the case when the sender was spoofed. The spoofed communication channels are generated by the attacker and is a well-known technique in a DoS attack.

The traffic baseline is consistently under ten devices per hour, for the days before and after the DDoS event detected by our classifier. Yet, during the DDoS event we observe the number of devices with traffic communicating to *weasyl.com*

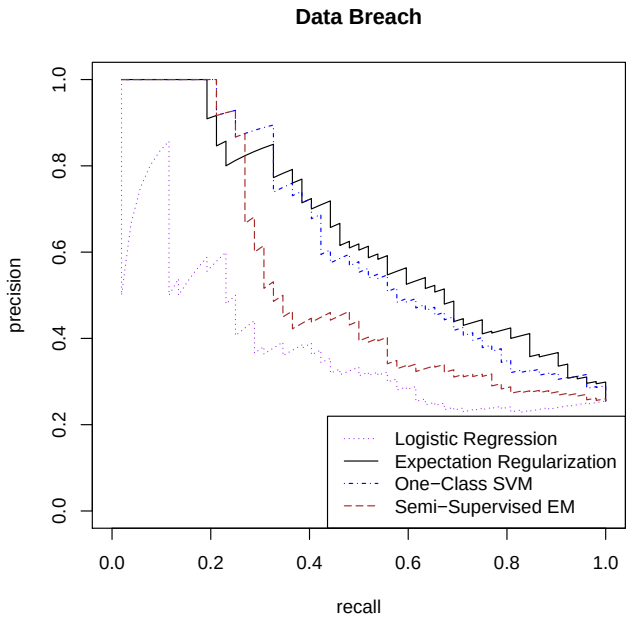

Figure 5: Precision and recall comparing various approaches to extracting events using only positive seeds and unlabeled data.

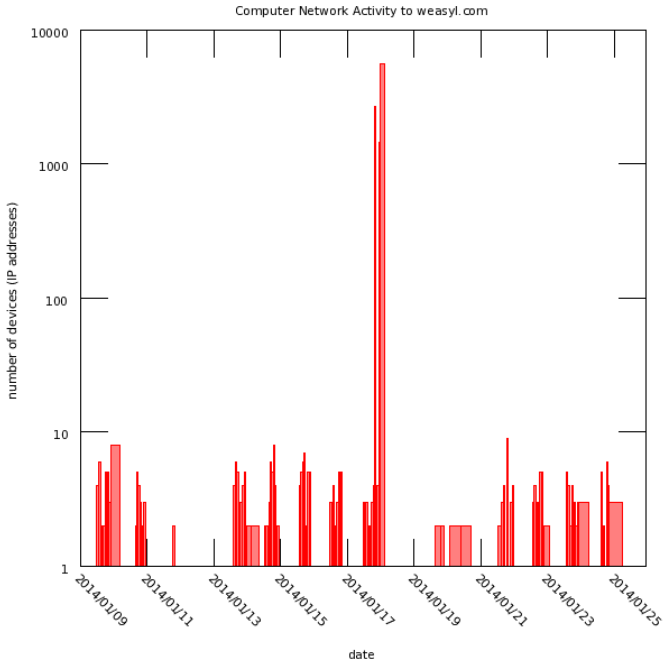

Figure 6: Network measurements (on log scale) for *weasyl.com* showing activity peak with DDoS detected by our classifier.

increasing by over three orders of magnitude. Further investigation of the network traffic led to identifying two distinct events approximately four hours apart. Both events were web-based SYN floods [33] where each of the source devices were randomly spoofed. Consequently, our detection of the event measured the backscatter [33] of responses from the webserver mistakenly responding to devices in our monitored network.

7. RELATED WORK

7.1 Weakly Supervised Relation Extraction

There has been substantial amount of work on weakly supervised relation extraction. Most previous work has focused on extracting static relations which remain relatively constant over time, for instance book authors [5, 1], class/instance pairs, named entities [31] or *hypernyms* [14, 32, 43]. The NELL system in particular [6] has pushed the boundaries of which relations are possible to extract.

Previous work on relation extraction has also addressed the challenge of false-negatives in weakly supervised learning [44, 39]. We believe these approaches are not directly applicable for our scenario, however, as the multiple instance learning assumption they use is inappropriate for our event data. Typically all tweets mentioning an entity-date tuple are instances of the category, or all are false positives.

There has been much less work in contrast on weakly supervised *event extraction*, however, as newswire coverage of a specific event is generally too sparse to support seed-based learning. There are hundreds of thousands of sentences on the web which mention William Shakespeare as the author of Comedy of Errors, or that Nirvana plays grunge music, however there are typically only one or a handful of news articles that report on a specific event, such as a DoS attack.

Social media, however greatly lowers the barrier to publishing making it easy for anyone to comment on events as they take place. This leads to substantial redundancy of information, as many users will typically comment on events of interest. This redundancy on Twitter makes seed-based weakly supervised event extraction a feasible task because it is easy to find a large number of event mentions for an event category given a few seed instances.

7.2 Event Extraction

There has been growing interest in information extraction and event identification in Social Media [13]. Benson et al. [3] use distant supervision to train a relation extractor which identifies artists and venues mentioned within tweets of users who list their location as New York City. Sakaki et al. [41] train a classifier to recognize tweets reporting earthquakes in Japan; they demonstrate their system is capable of recognizing almost all earthquakes reported by the Japan Meteorological Agency. Ritter et. al. [38] demonstrate how to extract an open-domain calendar of events referenced in the future. Becker et. al. [2] link tweets to structured data sources describing events. Petrović et. al. present a streaming approach to identifying Tweets which are the first to report breaking news stories [34]. Chierichetti et. al [7] showed it is possible to detect high-volume events such as goals during the world cup using only non-textual evidence.

In contrast to previous work on event identification in social media, we take a weakly supervised approach to extracting focused event categories where only a few seed instances are provided. We also are the first to demonstrate the preva-

lence of security-related events reported on Twitter, and investigate how to automatically detect them.

A small amount of recent work has explored extracting security related information from text, such as understanding software vulnerability ontologies [18]. There has also been work demonstrating that private information can be inadvertently revealed on social media and studying how to prevent this [15]. In contrast we demonstrate Twitter is a valuable resource for information on security events, explore the challenges and opportunities presented by security event extraction in Twitter and focus on the goal of timely extraction of events as soon as they are reported. As far as we are aware this is the first work to explore extracting security-related events from social media.

8. CONCLUSIONS

Motivated by the wide variety of event categories which might be of interest to track, we proposed a weakly supervised seed-based approach to event extraction from Twitter. We showed how this leads to an unusual learning problem where only a small number of positive seeds and a sample of unlabeled candidate events are available. A number of approaches were investigated to address this challenge, including label regularization, constrained semi-supervised EM and one-class SVMs. We applied this approach to detect several security-related events including DoS attacks and account hijacking incidents, and demonstrated that a large number of security-related events are mentioned on Twitter.

9. ACKNOWLEDGMENTS

This research has been supported in part by DARPA (under contract number FA8750-13-2-0005) and the Department of Defense under Contract No. FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

10. REFERENCES

[1] E. Agichtein and L. Gravano. Snowball: Extracting relations from large plain-text collections. In *Proceedings of the fifth ACM conference on Digital libraries*, pages 85–94. ACM, 2000.

[2] H. Becker, D. Iter, M. Naaman, and L. Gravano. Identifying content for planned events across social media sites. In *Proceedings of the fifth ACM international conference on Web search and data mining*, pages 533–542. ACM, 2012.

[3] E. Benson, A. Haghighi, and R. Barzilay. Event discovery in social media feeds. In *ACL*, 2011.

[4] B. Bishop. High-profile twitter account hijackings leave questions about security. web, May 2013.

[5] S. Brin. Extracting patterns and relations from the world wide web. In *The World Wide Web and Databases*. 1999.

[6] A. Carlson, J. Betteridge, B. Kisiel, B. Settles, E. R. Hruschka Jr, and T. M. Mitchell. Toward an architecture for never-ending language learning. In *AAAI*, 2010.

[7] F. Chierichetti, J. Kleinberg, R. Kumar, M. Mahdian, and S. Pandey. Event detection via communication pattern analysis. 2014.

[8] B. Claise. RFC 3954 - Cisco Systems NetFlow Services Export Version 9, Oct. 2004.

[9] C. Elkan and K. Noto. Learning classifiers from only positive and unlabeled data. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2008.

[10] H. Fei, Y. Kim, S. Sahu, M. Naphade, S. K. Mamidipalli, and J. Hutchinson. Heat pump detection from coarse grained smart meter data with positive and unlabeled learning. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '13, pages 1330–1338, New York, NY, USA, 2013. ACM.

[11] K. Ganchev, J. Graça, J. Gillenwater, and B. Taskar. Posterior regularization for structured latent variable models. *The Journal of Machine Learning Research*, 11:2001–2049, 2010.

[12] R. Grishman and B. Sundheim. Message understanding conference-6: A brief history. In *COLING*, 1996.

[13] W. Guo, H. Li, H. Ji, and M. T. Diab. Linking tweets to news: A framework to enrich short text data in social media. In *ACL (1)*, pages 239–249. Citeseer, 2013.

[14] M. A. Hearst. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th conference on Computational linguistics-Volume 2*, pages 539–545. Association for Computational Linguistics, 1992.

[15] R. Heatherly, M. Kantarcioglu, and B. Thuraisingham. Preventing private information inference attacks on social networks. *Knowledge and Data Engineering, IEEE Transactions on*, 25(8):1849–1862, 2013.

[16] R. Huang and E. Riloff. Bootstrapped training of event extraction classifiers. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, 2012.

[17] H. Ji and R. Grishman. Refining event extraction through cross-document inference. In *Proceedings of ACL-08: HLT*, Columbus, Ohio, 2008. Association for Computational Linguistics.

[18] A. Joshi, R. Lal, T. Finin, and A. Joshi. Extracting cybersecurity related linked data from text. In *Semantic Computing (ICSC), 2013 IEEE Seventh International Conference on*, pages 252–259. IEEE, 2013.

[19] W. S. Lee and B. Liu. Learning with positive and unlabeled examples using weighted logistic regression. In *ICML*, volume 3, pages 448–455, 2003.

[20] C. Li, J. Weng, Q. He, Y. Yao, A. Datta, A. Sun, and B.-S. Lee. Twiner: named entity recognition in targeted twitter stream. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, pages 721–730. ACM, 2012.

[21] X. Li and B. Liu. Learning to classify texts using positive and unlabeled data. In *IJCAI*, pages 587–592, 2003.

[22] X. Liu, S. Zhang, F. Wei, and M. Zhou. Recognizing named entities in tweets. In *Proceedings of the 49th Annual Meeting of the Association for Computational*

Linguistics: Human Language Technologies-Volume 1, pages 359–367. Association for Computational Linguistics, 2011.

[23] M. Lui and T. Baldwin. langid. py: An off-the-shelf language identification tool. In *Proceedings of the ACL 2012 System Demonstrations*, pages 25–30. Association for Computational Linguistics, 2012.

[24] G. S. Mann and A. McCallum. Simple, robust, scalable semi-supervised learning via expectation regularization. In *Proceedings of the 24th international conference on Machine learning*, pages 593–600. ACM, 2007.

[25] G. S. Mann and A. McCallum. Generalized expectation criteria for semi-supervised learning with weakly labeled data. *The Journal of Machine Learning Research*, pages 955–984, 2010.

[26] M. Mintz, S. Bills, R. Snow, and D. Jurafsky. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*. Association for Computational Linguistics, 2009.

[27] M. Motoyama, B. Meeder, K. Levchenko, G. M. Voelker, and S. Savage. Measuring online service availability using twitter. *WOSNâĂŹ10*, pages 13–13, 2010.

[28] A. Narayanan. Fast dictionary attacks on passwords using time-space tradeoff. In *In ACM Conference on Computer and Communications Security*, pages 364–372. ACM Press, 2005.

[29] G. Neubig, Y. Matsubayashi, M. Hagiwara, and K. Murakami. Safety information mining-what can nlp do in a disaster-. In *IJCNLP*, pages 965–973, 2011.

[30] K. Nigam, A. McCallum, and T. Mitchell. Semi-supervised text classification using em. *Semi-Supervised Learning*, 2006.

[31] M. Paşca. Weakly-supervised discovery of named entities using web search queries. In *Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management*, CIKM '07, pages 683–690, New York, NY, USA, 2007. ACM.

[32] P. Pantel and M. Pennacchiotti. Espresso: Leveraging generic patterns for automatically harvesting semantic relations. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 113–120. Association for Computational Linguistics, 2006.

[33] T. Peng, C. Leckie, and K. Ramamohanarao. Survey of network-based defense mechanisms countering the dos and ddos problems. *ACM Comput. Surv.*, 39(1), April 2007.

[34] S. Petrović, M. Osborne, and V. Lavrenko. Streaming first story detection with application to twitter. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 181–189. Association for Computational Linguistics, 2010.

[35] V. Qazvinian, E. Rosengren, D. R. Radev, and Q. Mei. Rumor has it: Identifying misinformation in microblogs. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1589–1599. Association for Computational Linguistics, 2011.

[36] K. Reschke, M. Jankowiak, M. Surdeanu, C. D. Manning, and D. Jurafsky. Event extraction using distant supervision. 2014.

[37] A. Ritter, S. Clark, O. Etzioni, et al. Named entity recognition in tweets: an experimental study. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2011.

[38] A. Ritter, O. Etzioni, S. Clark, et al. Open domain event extraction from twitter. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1104–1112. ACM, 2012.

[39] A. Ritter, L. Zettlemoyer, Mausam, and O. Etzioni. Modeling missing data in distant supervision for information extraction. *TACL*, 2013.

[40] K. Roberts, T. Goodwin, and S. M. Harabagiu. Annotating spatial containment relations between events. In *LREC*, 2012.

[41] T. Sakaki, M. Okazaki, and Y. Matsuo. Earthquake shakes twitter users: real-time event detection by social sensors. In *WWW*, 2010.

[42] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson. Estimating the support of a high-dimensional distribution. *Neural computation*, 2001.

[43] B. Wellner. Weakly supervised learning methods for improving the quality of gene name normalization data. In *Proceedings of the ACL-ISMB Workshop on Linking Biological Literature, Ontologies and Databases: Mining Biological Semantics*, ISMB '05, pages 1–8, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics.

[44] W. Xu, R. Hoffmann, L. Zhao, and R. Grishman. Filling knowledge base gaps for distant supervision of relation extraction. In *ACL (2)*, 2013.