On the Potential for Mining Unstructured Public Data to Aid Network Intelligence

Robert Beverly
Naval Postgraduate School
rbeverly@nps.edu

Lance Alt
Naval Postgraduate School
laalt@nps.edu

1. PROBLEM

Within both the network security and measurement fields is a need to obtain representative data on the current state of the network. Today, distributed network sensors collect such data in order to help detect, characterize, and mitigate emerging abusive or malicious network-borne attacks. These sensors may monitor traffic passively, or actively attract traffic, for instance by acting as a honeypot. Assimilating traffic and features collected from many sensors can help reveal e.g. new exploits, rapidly spreading malware, and denial-of-service attacks.

While such sensors are an invaluable part of modern network security, their utility is limited by the size and distribution of deployment. A passive network sensor will only observe traffic from connections that traverse the link it is monitoring. Similarly, the results of active network measurements are highly dependent on the location of the sensor in the network – consider, for example, a BGP hijack event that pollutes only a portion of the Internet, rendering it detectable at the data-plane only by monitors within affected autonomous systems.

We posit that *unstructured public data* can be data mined to aid network intelligence. Such non-traditional data sources include any available real-time data feeds: electronic mailing lists, RSS, twitter, or other social media. In this nascent work we limit our analysis to the potential mine data from an electronic mailing list, as well as making several simplifying assumptions.

While significant prior work has examined data mining social media, the focus has traditionally been on understanding the social network, shard topics of interest, recommendations, etc [5]. Other work seeks to understand opinions and sentiment of social media messages [3]. Our work utilizes similar techniques, but seeks to automatically fuse critical information within messages with the network entities they discuss. Closely related to our present effort is the work of Trestian et al. [6] who obtain host profiles by using a search engine to find all web-archived references to the host. Using a set of rules, end hosts can be classified a one of several different classes. In contrast, we use data-mining techniques on real-time unstructured data feeds to understand the context in which a particular host is being discussed relative to a particular moment in time.

Our goal, as yet unrealized, is to determine whether the per-host inferences we make represent real network events of interest and whether such inferences are predictive. In this abstract, we describe our approach, initial results, and plans for future work.

2. APPROACH

Given one or more incoming, real-time data feeds containing unstructured data, e.g. mailing list, RSS, twitter, etc, we seek to efficiently automate two tasks: i) finding any referenced network entities, e.g. IP addresses, prefixes, or autonomous systems; and ii) understanding the context in which the entities are being referenced. We have simplified the first problem by restricting our attention to IP addresses using a regular expression. However, to understand the context of a message containing an IP address reference, we employ Natural Language Processing (NLP) techniques [2]. Figure 1 shows our high-level system overview.

As a proof-of-concept, we examine 497 email messages from the North American Network Operators Group (NANOG) mailing list [1] archived from June to September, 2014. NANOG is an operationally focused list serving as a point of coordination among large and small providers alike. For example, a network administrator might post a message inquiring whether anyone else is experiencing problems reaching a particular network, or seek more information on an ongoing attack (for instance, whether it is affecting other networks). Thus, the NANOG mailing list contains a diverse set of messages ranging from outages to equipment reviews to security events to pure socializing.

We treat understanding the meaning of each mailing list message as a supervised learning problem and manually label 497 messages selected at random into three groups: security, outage, and junk (Table 1).

Messages in the "security" group contain any information relating to network security such as: security updates and patches, active attacks, and vulnerability

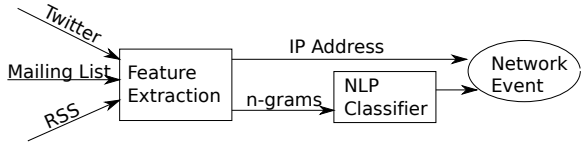

Figure 1: System overview: Mining unstructured data for network events.

disclosures. Messages in the "outage" group contain any information relating to networks/sites that are unreachable, have high latency, or are experiencing routing problems. All other messages are grouped as "junk."

Our preliminary approach is to use the Naïve Bayes machine learning algorithm in NLTK [2] to classify messages. Naïve Bayes requires a set of feature to describe each message. Our features include:

- ***Structure:*** Message features evaluate a message in its entirety. We evaluated the message length in words, whether it contained an IP address or URL, and the number of unique words.
- ***Language:*** The natural language features include word or phrase tokens found in each message by separating on whitespace. The entire data set was parsed finding all unique n-grams, where n represents the number of consecutive tokens. All n-grams with $1 \leq n \leq 3$ were evaluated using the *most_informative_features* function provided by the NLTK Bayes classifier. The top 100 unique features were selected for testing.

Prior to extracting features, each message is pre-processed by first lower-casing each character and removing punctuation. For the purposes of NLP classification, we are interested in the presence of IP addresses and URLs, not their specific values. We therefore replace these with special tokens. For example, each instance of an IP address (as parsed by a regular expression) is replaced with the token *@IPADDR@*. The rationale behind this approach is to encode the n-gram context surrounding an IP address rather than the IP address itself; this is important for e.g. traceroutes in messages.

Next, the message is tokenized into words and stopwords are removed. Finally, the remaining words are stemmed using the Porter [4] stemmer found in NLTK. Stemming reduces each word to its root form allowing comparison between words with the same root.

3. PRELIMINARY RESULTS

Using the 497 manually classified messages and the feature set described above, we evaluated the performance of the classifier using 10-fold cross-validation of the data set. Overall we obtained an accuracy of 81.7%. Table 1 shows the precision (fraction of items identified in a group that were truly in the class), recall (fraction of all items in a group that were identified as being in the class), and F-Score from the cross-validation testing. F-Score is computed as the harmonic mean of the precision and recall which provides a measure of the individual classification accuracy.

Group	Count	Precision	Recall	F-Score
Security	54	0.792	0.352	0.487
Outage	121	0.924	0.603	0.73
Junk	322	0.797	0.975	0.877

Table 1: Precision, recall, and F-Score observed from cross-validation of the data set using the Naïve Bayes model.

We note that the F-Score is proportional to the number of labeled examples in each class. Both the Outage and Junk groups are classified fairly accurately while the Security group suffered. Further analysis revealed that the majority of the Security messages were categorized as Junk. We hypothesize this is due to the small number of Security messages and large number of Junk messages in the data set. Providing more examples of Security messages to the classifier should increase accuracy and F-Score.

4. FUTURE WORK

Our preliminary results are encouraging. We plan two research thrusts going forward. First, we must improve our classification accuracy. Clearly, the unbalanced data set complexion influences our results. We intend to expand the data set by adding additional messages from the NANOG mailing list along with exploring other sources of unstructured data. Additionally, we will experiment with other machine learning algorithms that have shown promise in NLP, such as decision trees and support vector machines.

Second, once we obtain sufficient classification accuracy, we will use extracted IP addresses from Security and Outage messages to better understand whether our system can provide useful network intelligence. By correlating our inferred network event (an IP address and category) with external observables, such as reputation databases, or global routing tables, or other network sensors, we hope to build a better picture of the true network state.

5. REFERENCES

[1] North American Network Operators Group, 2014. `http://nanog.org/list`.
[2] S. Bird. NLTK: the natural language toolkit. In *Proceedings of the COLING/ACL*, pages 69–72. Association for Computational Linguistics, 2006.
[3] A. Pak and P. Paroubek. Twitter as a corpus for sentiment analysis and opinion mining. In *LREC*, 2010.
[4] M. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.
[5] M. A. Russell. *Mining the Social Web: Data Mining Facebook, Twitter, LinkedIn, Google+, GitHub, and More*. O'Reilly Media, 2013.
[6] I. Trestian, S. Ranjan, A. Kuzmanovi, and A. Nucci. Unconstrained endpoint profiling (googling the internet). In *Proceedings of ACM SIGCOMM*, pages 279–290, 2008.