View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by The Australian National University

# Towards a Feature Rich Model for Predicting Spam Emails containing Malicious Attachments and URLs

**Khoi-Nguyen Tran*, Mamoun Alazab#, Roderic Broadhurst#**
*Research School of Computer Science, #Regulatory Institutions Network
The Australian National University, ACT 0200, Australia
{khoi-nguyen.tran, mamoun.alazab, roderic.broadhurst}@.anu.edu.au

## Abstract

Malicious content in spam emails is increasing in the forms of attachments and URLs. Malicious attachments and URLs both attempt to deliver software that compromises the security of a computer. Malicious attachments try to disguise their content, but many email services offer virus scanners. Malicious URLs add another layer of disguise, where the email content tries to coerce the recipient to click a URL linking to a malicious Web site or download a malicious attachment. In this paper, we present our initial work in predicting spam emails containing these highly dangerous spam emails from two real world data sets. We propose a rich set of novel features for the content of emails to capture regularities in emails containing malicious content. We show these features can predict malicious attachments with an area under the precious recall curve (AUC-PR) up to 95.24%, and up to 68.09% for URLs. Our work reduces the need for virus scanners and URL blacklists, which often do not update as quickly as malicious content appears and requires many different resources to identify malicious content.

*Keywords*: Email, Spam, Malicious, Attachment, URL, Machine Learning.

## 1 Introduction

Email spam, unsolicited bulk email (Blanzieri & Bryl, 2008), accounts for an average of 66.5% of all emails sent in the first quarter of 2013, where 3.3% of all emails contained malicious attachments[1]. Estimates show that approximately 183 billion emails (6 billion emails with malicious attachments) are sent every day in the first quarter of 2013[2]. Malicious attachments and URLs (Universal Resource Locators – also known as Web links) are attempts to infect the computer of a recipient with malware (malicious software) such as viruses, trojans, and keyloggers. Malicious attachments in emails are a direct delivery method for malware, whereas malicious URLs are indirect. These spam emails with malicious content (attachments or URLs) try to coerce the recipient into opening attachments or click on URLs. These spam emails have subject and content text that entices or alarms the recipient to act on the malicious content.

To find this type of dangerous spam emails, scanning the attachments of emails and URLs with virus scanners or against blacklists often reveals their malicious content. However, scanning emails require external resources that are often computationally expensive and difficult to maintain (Ma, Saul, Savage, & Voelker, Identifying Suspicious URLs: An Application of Large-Scale Online Learning, 2009). This method of identifying spam and other spam filtering methods often aim to be more reactive to changes in spamming techniques than spammers, and are not robust to handle variations in spam emails (Blanzieri & Bryl, 2008).

The task of identifying malicious content (attachments or URLs) in spam emails is not well studied, as far as we are aware. Our specific definition of malicious to include only malware differentiates from research in classifying phishing emails by analysing URLs in their content. This task is important as it identifies one of the most harmful types of spam emails for recipients.

In this initial work, we propose novel features for predicting malicious attachments and URLs in spam emails. We hypothesise that spam emails with malicious attachments or URLs can be predicted only from the text content in the email subject and body. Our work differs from related work as it is self-contained (do not require external resources) and do not add risks of exposure to malicious content by analysing or scanning attachments, or by following URLs. We use two real world data sets obtained from two different sources. The first data set is from the Habul plugin for the Thunderbird mail client, and the second data set, Botnet, is collected from honeypots around the world to study the characteristics of email spam botnets.

We extract many features from metadata and text content of these real world spam emails. These proposed features are: self-contained (no need to scan emails using external resources such as virus scanners and blacklists); robust (high adaptability to changes in spamming techniques); and time efficient (process many emails per second). We apply a Random Forest classifier on these features to show their effectiveness in distinguishing spam emails with and without malicious attachments. However, our features are insufficient to classify spam emails with and without malicious URLs. We discuss

Copyright © 2013, Australian Computer Society, Inc. This paper appeared at Eleventh Australasian Data Mining Conference (AusDM 2013), Canberra, 13-15 November 2013. Conferences in Research and Practice in Information Technology, Vol. 146. Peter Christen, Paul Kennedy, Lin Liu, Kok-Leong Ong, Andrew Stranieri and Yanchang Zhao, Eds. Reproduction for academic, not-for profit purposes permitted provided this text is included.

[1] Kaspersky Lab Securelist article: "Spam in Q1 2013." (8 May 2013) http://www.securelist.com/en/analysis/204792291/Spam_in_Q1_2013

[2] Radicati Group Reports – Executive Summary: "Email Statistics Report, 2013-2017." (22 April 2013) http://www.radicati.com/wp/wp-content/uploads/2013/04/Email-Statistics-Report-2013-2017-Executive-Summary.pdf

reasons for success and failure of our features and potential research directions from this initial work.

Our contributions in this initial work are (1) developing novel features that do not require external resources for the task of classifying malicious spam emails, (2) evaluating these features on two real-world data sets, and (3) demonstrating malicious attachments can be predicted from only the email itself with high classification scores. Our work reduces the need to scan emails for malicious content, saving time and resources.

The rest of this paper is organised as follows. Section 2 summarises related work. Section 3 explains the structure and content of malicious spam emails, and Section 4 details our real world data sets. Section 5 presents our proposed features to capture malicious intent in these emails. Section 6 details our evaluation methodology and Section 7 summarises our results. We discuss our results in Section 8 and conclude our findings in this initial work in Section 9.

## 2 Related Work

We summarise related work in four aspects of our work, highlighting text and machine learning based approaches. We look at spam filtering and related work specifically on classifying malicious attachments and URLs. From a related field of Wikipedia vandalism detection, we borrow some features and adapt them to our problem.

### 2.1 Email Spam Filtering

Spam filtering is a well-developed field with many techniques for many types of spam (Blanzieri & Bryl, 2008). A survey of machine learning based approaches to spam filtering by Blanzieri & Bryl (2008) covers the ambigious definitions of spam, summarises a variety of spam detection methods and their applicability to different parts of an email, and summarises the various data sets used in research. The survey shows a variety of machine learning approaches that rely on features extracted from the email header, body, and the whole email message.

In summary, email spam filtering is a mature research field with many filtering techniques such as rule based, information retrieval based, machine learning based, graph based, and hybrid techniques. However, identifying emails with malicious content is a problem within this research area that has not been well investigated.

### 2.2 Classification of Malicious Attachments

Emails containing malicious attachments are one the most dangerous types of emails as its malware has the potential to do significant damage to computers and to spread rapidly. The email usage behaviour can change depending on the malware's intent on spreading infection. By engineering features that capture behavioural properties of email usage and emails, the outgoing email behaviour of users can predict when malware has compromised a computer (Martin, Nelson, Sewani, Chen, & Joseph, 2005). Applying feature reduction techniques can further improve classification accuracy of malware propagating in outgoing mail (Masud, Khan, & Thuraisingham, 2007). These approaches aim to identify new malware by behaviour after infection.

For preventative solutions without needing to scan attachments, analysing properties of the software executables can reveal malicious intent (Wang, Yu, Champion, Fu, & Xuan, 2007). Our work also aims to be preventative, but without adding the risk of infection by analysing software executables.

### 2.3 Classification of Malicious URLs

Research on classifying URLs for malicious intent extend beyond spam emails, because of the common nature of URLs in many Web documents and communications. Blacklisting is a highly efficient method of preventing access to malicious URLs, but it relies on knowing those URLs are malicious beforehand (Ma, Saul, Savage, & Voelker, Learning to Detect Malicious URLs, 2011). Furthermore, blacklisting services cannot keep up with spamming bots operating at various URLs and IP addresses (Ramachandran, Dagon, & Feamster, 2006).

To be effective and adaptive to new malicious URLs, engineering URL features based on text and hosting properties for classifiers has shown to be successful (Ma, Saul, Savage, & Voelker, Beyond Blacklists: Learning to Detect Malicious Web Sites from Suspicious URLs, 2009) (Le, Markopoulou, & Faloutsos, 2011). However, these features require many external resources such as IP blacklists, domain registration details, DNS records, and geographical location of IP addresses. Although they can be applied in real-time classification of URLs, there are trade-offs in accuracy and processing quantity (Ma, Saul, Savage, & Voelker, Identifying Suspicious URLs: An Application of Large-Scale Online Learning, 2009).

Other methods of detecting malicious URLs require accessing the Web pages of URLs and performing further analysis. Parts of Web pages can be obfuscated to hide malicious intent, such as malicious Javascript code (Likarish, Jung, & Jo, 2009). However, developing many feature sets over the structure and content of provides a comprehensive analysis of the malicious nature of Web pages (Canali, Cova, Vigna, & Kruegel, 2011).

### 2.4 Wikipedia Vandalism Detection

In this initial work, we borrow some text features from a related field of vandalism detection on Wikipedia. The problem of vandalism (a malicious edit) detection and detecting emails with malicious content have similar characteristics. In both cases, the text within a Wikipedia article and text in an email may contain content that distinguishes it from a normal article or normal (spam) email, respectively. For example, abnormal use of vulgar words or excessive uppercase words may hint at malicious intent. Our initial work provides a pathway to share classification models between these two research areas to address the problem of insufficient training samples for classification models.

The PAN Workshops in 2010 and 2011 held competitions for vandalism detection in Wikipedia, where they released a data set containing manually classified cases of vandalism. In Section 7, we describe our selected text features from the winners of the competitions in 2010 (Velasco, 2010) and 2011 (West & Lee, 2011). These text features aim to show text regularities within spam emails.

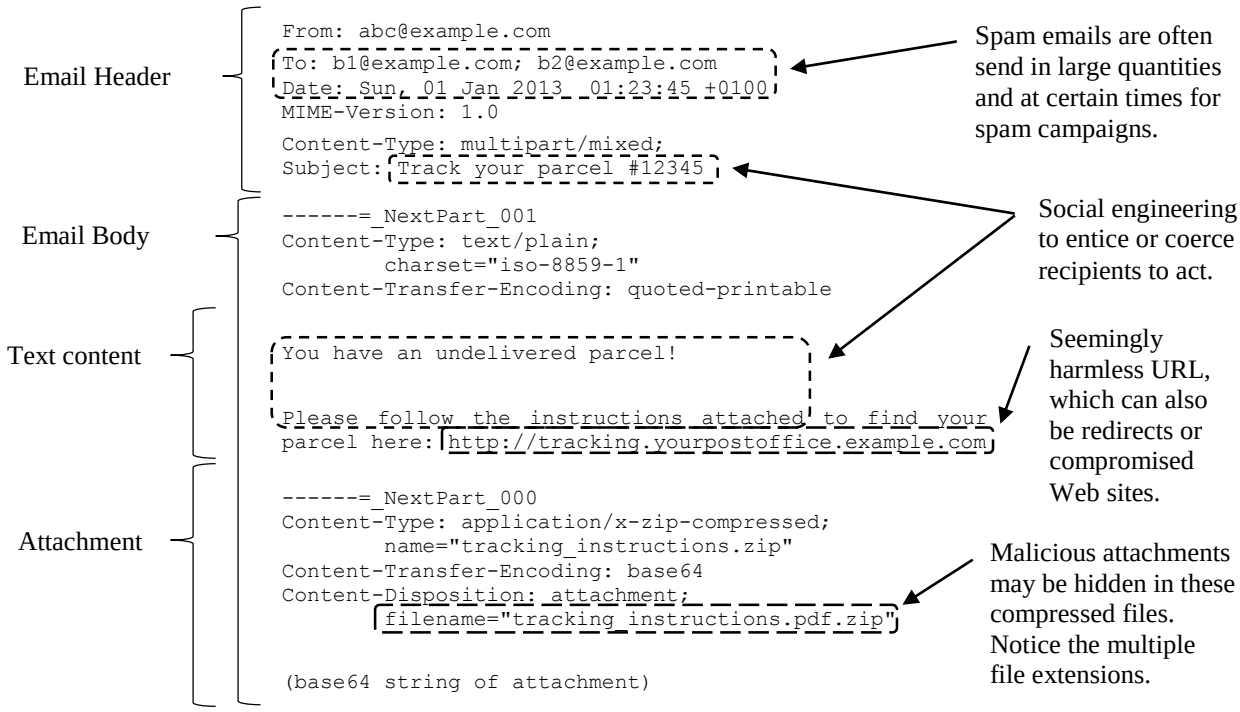



**Figure 1: An example (fake) spam email with a potential malicious attachment and URL.**

## 3 Malicious Spam Emails

Spam emails vary from annoying, but harmless, advertising to dangerous scams, fraudulent activity, and enabling cybercrime. Spam emails with malware or URLs directing to malware are cybercriminals looking to find new victims. For example, spammers may be wanting to expand their botnets or cybercriminals may be looking to propagate their computer viruses to harvest passwords, credit cards, bank accounts, and other sensitive personal information. Our work aims to be a preventative method to stop the propagation of malware using spam emails. Before presenting our results, we briefly describe our raw data of malicious spam emails and how cybercriminals send spam emails.

Emails are well-known to many people, but not the raw email data where we construct our features. We present an example of a (fake) spam email with potential malicious content in Figure 1, stripped of irrelevant metadata. The figure shows an email in raw text format with annotations showing important parts of the email for feature construction. We have the email header that contains delivery instructions for mail servers, and the email body that can have many sections for text, attachments, and other types of attachable data. Emails are identified as spam in two ways: a human determines an email is spam, and emails collected from known spamming networks. Both scenarios of determining spam are captured in our two real world data sets.

Our example in Figure 1 shows a typical structure of a malicious spam email. The subject or text content of malicious spam emails often contains social engineering methods to manipulate recipients into reading or acting on the email. In this case, we have the premise of a fake undelivered parcel requiring the recipient to download a compressed file (purposefully misleading with multiple file extensions). This compressed file serves the purpose of hiding malware executables, and hiding its malware from virus scanners operating at mail servers. The URL in this example acts as a secondary method of delivering malicious content. Similar to attachments, malicious URLs can disguise its true malicious Web site (e.g. example.com) by adding subdomains representing a known Web site (e.g. tracking.yourpostoffice). Our example also shows a possible spam template, where attachments or URLs may have different names, but the same malicious intent.

Spam templates are often used in spam campaigns, where many emails are sent in a short period of time with lexical variations to their content (Stone-Gross, Holz, Stringhini, & Vigna, 2011). In our example in Figure 1, variations can occur in the tracking number, attachment name, and URL. These variations are attempts to prevent basic spam detection methods by mail servers. Other obfuscation methods include manipulation of email headers to include legitimate email addresses to avoid spam filtering and allowing more spam emails to be sent.

The emergence and proliferation of botnets have allowed large quantities of spam emails to be sent in a coordinated way, and amplify cybercrime activities (Broadhurst, et al., 2013). Botnets are networks of compromised computers controlled by a person, named as the botmaster. Botnets are the backbone of spam delivery, where estimates suggest approximately 85% of the world's spam email are sent by botnets each day (John, Moshchuk, Gribble, & Krishnamurthy, 2009). The use of botnets show how spammers understand and manipulate the networks of compromised computers and servers around the world to ensure high volumes of spam are delivered to many people.

Overall, the use of spam emails to propagate malware is an important problem as the social engineering in spam emails provides a direct infection method to recipients.

## 4 Email Spam Data Sets

We use two real world data sets from two different spam collection sources. The first comes from the Habul Plugin for Thunderbird (an offline mail client) that uses an adaptive filter to learn from a user's labelling of spam and normal email. Table 3 summarizes the statistics of the Habul data set, which are compiled monthly. The second data set is compiled from a global system of spam traps designed to monitor information about spam and other malicious activities. We name the second data set as Botnet. Table 2 summarizes the statistics of Botnet data set, which are also compiled monthly. We receive both data sets in anonymised form, so no identifiable email addresses or IPs are available for analysis.

For each email, we extract attachments and URLs and upload to VirusTotal[3], a free online virus checker that offers support for academic researchers, to scan for viruses and suspicious content. VirusTotal uses over 40 different virus scanners, where we consider an attachment or URL to be malicious if at least one scanner shows a positive result. For this initial study, we only focus on emails with attachments or URLs to predict emails with malicious content.

The Habul data set is relatively smaller than the Botnet data set, but has the advantage of emails being manually labelled as spam. This means spam in the Habul data set has reached its recipient and has been viewed. The Botnet data set contains spam that circulates the world, but without certainty that the emails have reached their intended targets.

Both data sets show some similarities, such as nearly half of spam emails contain at least one URL, but only a low percentage are malicious. In contrast, many more emails with attachments are malicious. For each data set, there are peaks of spam with and without malicious content, which suggests different types of spam campaigns. These campaigns usually have shared similarities in the content of their emails, which may indicate malicious content without needing to scan.

| Habul | | with Attachments | | with URLs | |
|---|---|---|---|---|---|
| Month | Emails | Total | Mal. | Total | Mal. |
| Jan | 67 | 7 | 3 | 25 | 3 |
| Feb | 104 | 10 | 2 | 33 | 6 |
| Mar | 75 | 5 | 0 | 28 | 4 |
| Apr | 65 | 4 | 2 | 26 | 2 |
| May | 83 | 4 | 0 | 38 | 5 |
| Jun | 94 | 1 | 0 | 41 | 5 |
| Jul | 72 | 2 | 1 | 26 | 11 |
| Aug | 85 | 0 | 0 | 46 | 10 |
| Sep | 363 | 11 | 7 | 140 | 4 |
| Oct | 73 | 1 | 1 | 11 | 3 |
| Nov | 193 | 4 | 0 | 89 | 13 |
| Dec | 95 | 6 | 3 | 31 | 12 |
| Total | 1,369 | 55 | 19 | 534 | 78 |

**Table 1: Habul Data Set Statistics**

| Botnet | | with Attachments | | with URLs | |
|---|---|---|---|---|---|
| Month | Emails | Total | Mal. | Total | Mal. |
| Jan | 31,991 | 139 | 27 | 12,480 | 4 |
| Feb | 49,085 | 528 | 66 | 14,748 | 4 |
| Mar | 45,413 | 540 | 52 | 19,895 | 23 |
| Apr | 33,311 | 328 | 175 | 12,339 | 0 |
| May | 28,415 | 753 | 592 | 13,645 | 3 |
| Jun | 11,587 | 102 | 56 | 8,052 | 80 |
| Jul | 16,251 | 425 | 196 | 5,615 | 92 |
| Aug | 21,970 | 291 | 113 | 16,970 | 707 |
| Sep | 27,819 | 282 | 12 | 17,924 | 442 |
| Oct | 13,426 | 899 | 524 | 4,949 | 2 |
| Nov | 17,145 | 1,107 | 882 | 7,877 | 49 |
| Dec | 20,696 | 621 | 313 | 7,992 | 241 |
| Total | 317,109 | 6,015 | 3,008 | 142,486 | 1,647 |

**Table 2: Botnet Data Set Statistics**

## 5 Feature Engineering

In this initial work, we explore a comprehensive set of features for email content. We borrow some of these features from a related field of vandalism detection on Wikipedia, where the aim is to identify malicious modifications to articles. In particular, we borrow some text features from the winners of vandalism competitions held at the PAN Workshops in 2010 and 2011 (Velasco, 2010) (West & Lee, 2011). As far as we are aware, none of the features described below have been used to predict malicious content in emails. We describe their novelty in the context of their applications in other areas of research.

### 5.1 Feature Description

Table 3 shows our features and a summary description. Features with prefix H are email header features; prefix S are subject features; prefix P are payload features (or content of email); prefix A are features of attachments; and prefix U are features of URLs. We describe these features in detail below in these groups of relatedness.

#### 5.1.1 Header Features

Features **H01** to **H04** are simple time features that captures when emails were sent. The times of emails have been normalised to Greenwich Median Time (GMT) to account for emails being sent from different servers around the world. Emails from spam campaign are often sent at the same time in mass quantities.

Features H05 and H06 are counts of the email addresses of the sender and intended recipients. Since these features have been anonymised, we only count the number of addresses. We intend on expanding analysis on these anonymised email addresses in future work for features such as targeted spam campaigns.

#### 5.1.2 Text Features

These features are applied to the subject (prefix S) and payload (prefix P) of emails. Although we apply calculate these features identically on different data, they have some differences in meaning for subject and payload data. For text in the subject and payload, we extract a list of words and the count of appearance of each word.

Feature S01 (P01) is a simple count of the number of characters in the text of the subject or payload.

Features S02 to S04 (P02 tp P04) are a count of special words in emails. We obtain lists of these words

[3] https://www.virustotal.com/en/

| Feature | Description |
|---|---|
| **H01-DAY** | Day of week when email was sent. |
| **H02-HOUR** | Hour of day when email was sent. |
| **H03-MIN** | Minute of hour when email was sent. |
| **H04-SEC** | Second of minute when email was sent. |
| **H05-FROM** | Number of "from" email addresses, known as email senders. |
| **H06-TO** | Number of "to" email addresses, known as email recipients. |
| S01-LEN | Number of characters. |
| S02-PW | Number of pronoun words. |
| S03-VW | Number of vulgar words. |
| S04-SW | Number of slang words. |
| S05-CW | Number of capitalised words. |
| S06-UW | Number of words in all uppercase. |
| S07-DW | Number of words that are digits. |
| S08-LW | Number of words containing only letters. |
| S09-LNW | Number of words containing letters and numbers. |
| S10-SL | Number of words that are single letters. |
| S11-SD | Number of words that are single digits. |
| S12-SC | Number of words that are single characters. |
| **S13-UL** | Max ratio of uppercase letters to lowercase letters of each word. |
| **S14-UA** | Max of ratio of uppercase letters to all characters of each word. |
| **S15-DA** | Max of ratio of digit characters to all characters of each word. |
| **S16-NAA** | Max of ratio of non-alphanumeric characters to all characters of each word. |
| **S17-CD** | Min of character diversity of each word. |
| **S18-LRC** | Max of the longest repeating character. |
| **S19-LZW** | Min of the compression ratio for the lzw compressor. |
| **S20-ZLIB** | Min of the compression ratio for the zlib compressor. |
| **S21-BZ2** | Min of the compression ratio for the bz2 compressor. |
| **S22-CL** | Max of the character lengths of words. |
| **S23-SCL** | Sum of all the character lengths of words. |
| P01 to P12, **P13** to **P23** | Same as features S01 to S23, but for the email payload (content). |
| **A01-UFILES** | Number of unique attachment files in an email. |
| **A02-NFILES** | Number of all attachment files in an email. |
| **A03-UCONT** | Number of unique content types of attachment files in an email. |
| **A04-NCONT** | Number of all content types of attachment files in an email. |
| **U01-UURLS** | The number of unique URLs in an email. |
| **U02-NURLS** | The number of all URLs in an email. |

**Table 3: Email Features. Features in bold text are novel features not seen in other research areas.**

from Wiktionary[4] for English. This gives 27 unique pronoun words, 1064 unique vulgar words, and 5,980 unique slang words. These features are strong indicators of spam emails and possibly malicious content as the email payload attempts to persuade users to download files or follow URLs. These features are borrowed from the PAN Workshops (Velasco, 2010) (West & Lee, 2011), but using different sources for these words.

Features S05 to S12 (P05 to P12) are also borrowed from the PAN Workshops (Velasco, 2010) (West & Lee, 2011). These features are self descriptive and look for patterns in the words used in the subject and payload of emails. We expect these features to distinguish emails from spam campaigns as these campaigns often use email templates (Kreibich, et al., 2009).

Features **S13** to **S23** (**P13** to **P23**) are our set of novel proposed features. These features look closer at the distribution of character types in the form of ratios. We select out the maximum and minimum of each features applied to each word to highlight unique oddities in the words used in the email subject and payload. We give definitions of some less self-descriptive features:

- Character diversity is a concept borrowed from Velasco (2010). We interperet it here as a measure of different characters in a word compared to the word length: $length^{\frac{1}{unique\ characters}}$

- Compression ratio is defined as: $\frac{uncompressed\ size}{compressed\ size}$

In the subject of spam emails, these emphasise unique words much stronger than features S02 to S12, because of the relatively shorter length of text to the payload.

Features **S18** to **S21** are variants of the same concept of identifying words with repeating characters. We use these features to account for simple misspellings of words by repeating characters. These are the most computationally intensive features, with feature **S19** taking on average 4ms per email, and features **S18**, **S20**, and **S21** taking on average less than 1ms. All other features take on average between 0.0050ms and 0.0100ms per email. Note that these are timings to generate a single feature and does not include parallelisation and batch pre-processing of required data.

### 5.1.3 Attachment Features

These features (prefix A) are specific to spam emails with attachments. We do not use URL features with these attachment features. Our initial investigation only looks at simple, but novel, features of how attachments appear in emails. In particular, we count the number of files and the declared content types (such as image or zip files). For spam emails with attachments, malicious attachments may appear as the only attachment in emails, or attempt to hide in many different types of attachments. In future work, we look to generate more features from filenames or other attributes of attachments to avoid needing to scan for malicious content.

### 5.1.4 URL Features

These features (prefix U) are specific to spam emails with attachments. We do not use these features with the attachment features. These few features are novel in this classification task. In future work, we will look to apply more complex text analysis specifically for URLs to extract features that may distinguish URLs that direct to websites with and without malicious content. For example, the number of URLs with common domain names or common access pages.

## 5.2 Feature Ranking

With many varieties of features, we find features important to our classification task and compare across different data sets. The Random Forest classifier produces a ranking of these features based on their entropy scores (Pedregosa, et al., 2011). Please see Section 7. for a description of our classifier and classification results.

[4] http://wiktionary.org

| Type | Attachments | | | | URLs | | | |
|---|---|---|---|---|---|---|---|---|
| Data Set | Habul | | Botnet | | Habul | | Botnet | |
| Month | Feature | Score | Feature | Score | Feature | Score | Feature | Score |
| Nov | S05-CW | 0.1115 | **S21-BZ2** | 0.1066 | **U02-NURLS** | 0.0875 | **H01-DAY** | 0.0628 |
| | **S23-SCL** | 0.0812 | **S20-ZLIB** | 0.0860 | **U01-UURLS** | 0.0719 | P01-LEN | 0.0562 |
| | S09-LNW | 0.0741 | **S17-CD** | 0.0722 | P09-LNW | 0.0530 | **P23-SCL** | 0.0536 |
| | **S15-DA** | 0.0665 | **S19-LZW** | 0.0581 | **P21-BZ2** | 0.0508 | **H03-MIN** | 0.0531 |
| | **H02-HOUR** | 0.0628 | **S22-CL** | 0.0451 | P08-LW | 0.0406 | **H02-HOUR** | 0.0476 |

**Table 4: Top 5 features determined by Random Forest classifier. Scores are the information entropy of features.**

The entropy scores measure the information gained when splitting a decision tree (in the forest) on that feature. The aim is to have the most homogenous decision branches after a split, which improves classification results. For example, for emails with attachments in the Botnet data set, we gain more than twice as much information by splitting on feature S21 (0.1066) than on feature S22 (0.0451). To account for randomness in the Random Forest classifier, we present the average scores of 10 training iterations in Table 4 for the data split of November (details in Section 7). We bold features that are our novel contributions.

From Table 4, we see the majority of the top features are our proposed features for this classification task. In particular, for the larger Botnet data set with many email samples, we find our features perform consistently well. The variety of features show no feature dominates as a top 5 performer across data sets, and attachments and URLs. This further emphasise the need for a feature rich model to capture variations in different types of spam emails containing malicious content.

For the Habul data set, predicting malicious attachments and URLs from email content shows different important features. For attachments, we find features S05, S23, S09, and S15, all suggesting emails with capitalised words containing letters and digits in the subject line. This formality in the subject line attempts to gain the trust of recipients to open the email and download the attachments. The presence of feature H02 suggests these malicious spam email may be originating from spam campaigns. For URLs, we find URL and payload features. Features U02 and U01 appearing together suggests a few unique URLs. This suggests malicious spam emails contain few URLs with content to persuade recipients to click on those URLs.

For the Botnet data set, we find the subject of the email to be the strongest predictor of malicious attachments, whereas when the email was sent to be a good predictor of malicious URLs. For attachments, we have the email subject having low compressibility of words for all three compression algorithms (S21, S20, and S19), many different characters (S17), and long words (S22). This suggests subject lines with seemingly random characters, which may trigger curiosity from the recipient to download the malicious attachments within the email. For URLs, the time features are highly predictive along with the length of the content of the email. This suggests spam campaigns with email templates of with strange subject text to induce the curiosity of recipients to download attachments.

For the two different data sets, we find similarities in the features that are predictive for predicting malicious attachments and URLs. Emails with attachments indicate their malicious intent mainly in their subject line. For emails with URLs, the malicious intent is seen in the number of URLs, the text, and when the emails were sent.

## 6 Evaluation Methodology

As our data sets are already partitioned into months, we combine the data sets by months, learn on the earlier months and test our classifier on the later months. Figure 2 illustrates our data splitting process into training and testing data sets for months Jul and Nov. For example, for the month of Jul, we train on all spam emails with malicious content from Jan to Jul, and test on spam emails with attachments or URLs from Aug to Dec. This shows the effects of different training sample size on classification quality, and adaptability of classifiers.

We combine the feature sets differently for classification of attachments and URLs. For attachments, we choose features with the prefixes of H, S, P, and A. For URLs, we choose with prefixes of H, S, P, and U.

We use three classifiers to evaluate our features: Naïve Bayes (NB), Random Forest (RF), and Support Vector Machine (SVM); and evaluation metrics from the Scikit-learn toolkit (Pedregosa, et al., 2011). The NB and SVM classifiers are commonly used in spam classification, whereas the RF classifier is not commonly used (Blanzieri & Bryl, 2008). We perform a standard grid search with 10-fold cross validation to determine the best parameters for each classifier.

We measure the performance of the classifier using the average precision score, also known as the area under the precision-recall curve (AUC-PR), and the accuracy (ACC). The AUC-PR scores give a probability that a randomly selected email with malicious content is correctly labelled by our classifier. The ACC scores give the percentage of spam emails that are correctly classified as containing malicious content or not. These measures are defined from four different scenarios from spam

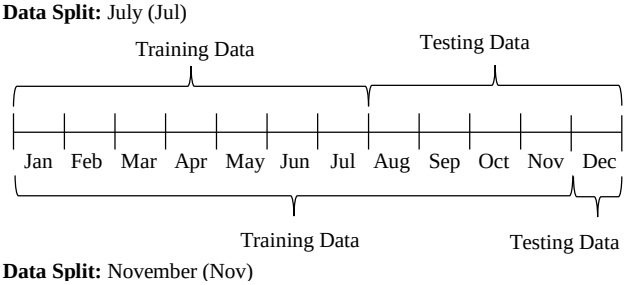
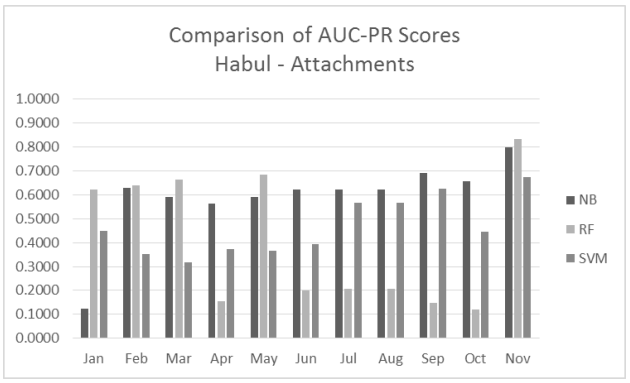
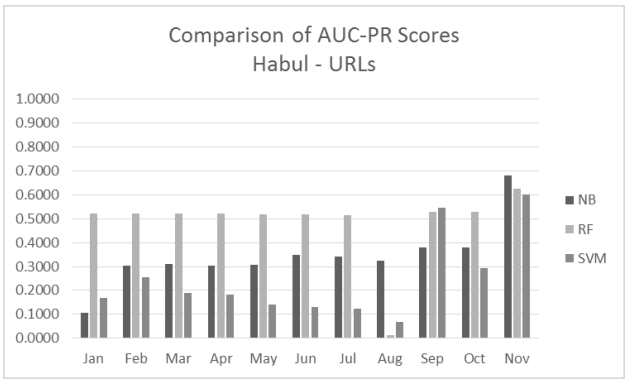


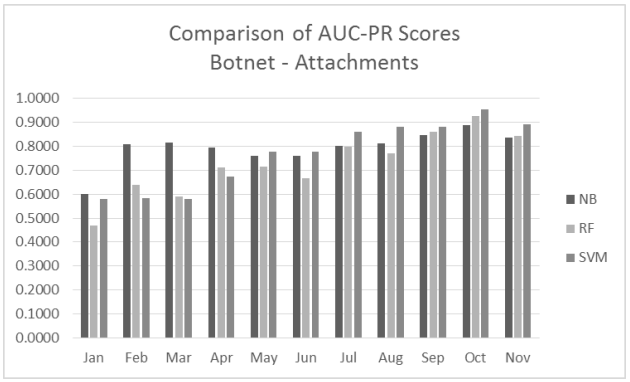

**Figure 2: Illustration of splitting data into training and testing sets.**



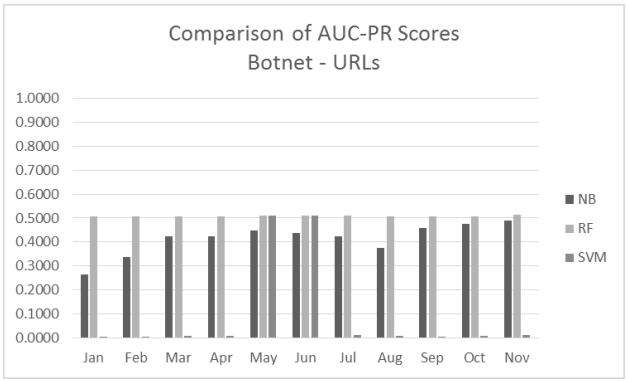

**(a)**




**(b)**

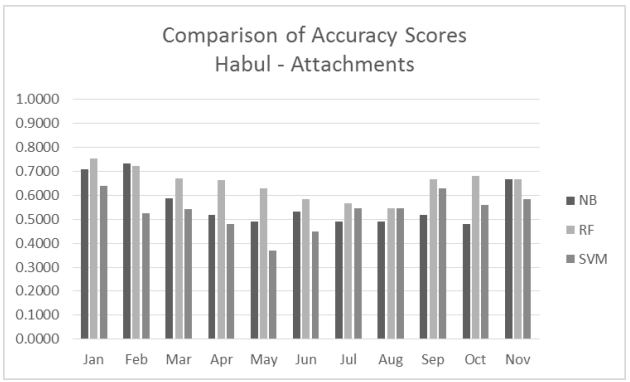



**(c)**

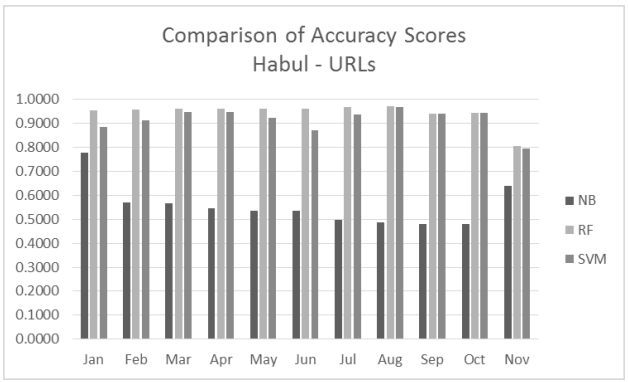



**(d)**

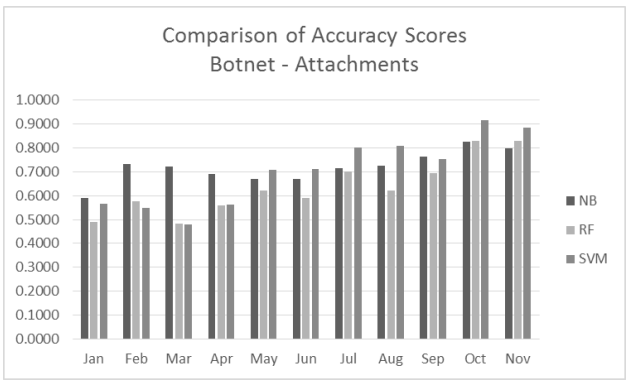
**Figure 4: Comparison of AUC-PR scores for three classifiers Naïve Bayes (NB), Random Forest (RF) and Support Vector Machine (SVM)**



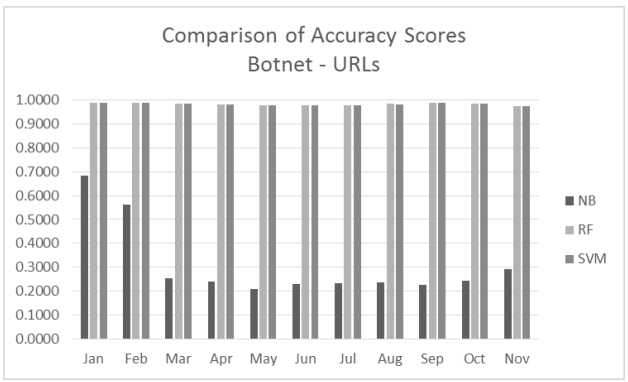

**(a)**




**(b)**

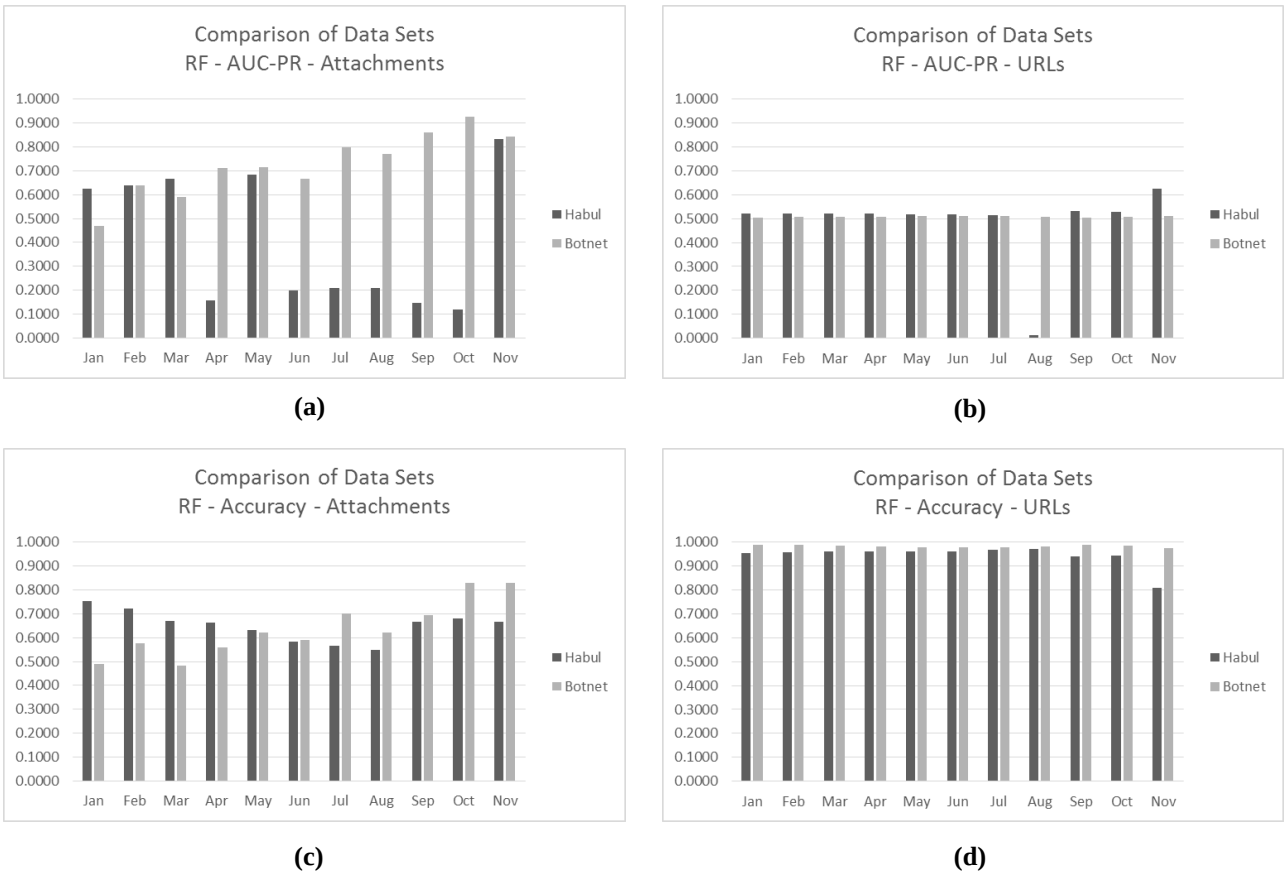



**(c)**




**(d)**

**Figure 3: Comparison of Accuracy (ACC) scores for three classifiers Naïve Bayes (NB), Random Forest (RF) and Support Vector Machine (SVM)**




**(a)** **(b)**

**(c)** **(d)**

**Figure 5: Comparison of RF classification scores across different data sets.**

emails with attachments or URLs: true positive (TP), emails correctly classified as containing malicious attachments or URLs; true negative (TN), emails correctly classified as non-malicious; false positive (FP), emails incorrectly classified as malicious; and false negative (FN), emails incorrectly classified as non-malicious. From these definitions, we have the positive precision value (precision) as $PPV = \frac{TP}{TP+FP}$, and the true positive rate (recall) as $TPR = \frac{TP}{TP+FN}$. By plotting $PPV$ against $TPR$ with instances of positive and negative values, we obtain a precision-recall (PR) curve, and calculate its area. We calculate the accuracy as: $ACC = \frac{TP+TN}{TP+FN+FP+TN}$.

As we are the first (as far as we are aware) to predict malicious content in emails. Thus, there are no comparable baseline measures available. In future work, we plan to expand our set of URL features and compare to related work of predicting phishing URLs in emails. For now, we present our classification results and discuss our findings in this initial work.

## 7 Classification Results

We compare the classification results for the three classifiers in Figure 4 for AUC-PR scores and in Figure 3 for ACC scores. In Figure 5, we compare our classification results for the SVM classifier. We compare the data splits in each figure for two different data sets and three different classifiers. Our figures also show the effect of accumulating spam data each month for predicting malicious emails in the subsequent months.

For emails with attachments, predicting whether attachments are malicious is highly successful on the Botnet data set, reaching a peak AUC-PR score of 0.9261 (Figure 4 (a) and (c)). The low AUC-PR score for training set split in Jan is expected as we have insufficient data to learn whether attachments are malicious in the subsequent months (Feb to Dec). The classifier shows very poor performance on the Habul data set for many data splits (Figure 4 (a)). The reason is clear from Table 1, where we see very few emails with attachments for the classifier to learn from. In some months corresponding with the data splits (e.g. Aug), we do not have any or few emails with malicious attachments. The low AUC-PR (Figure 4 (a) and (c)) and high ACC scores (Figure 3 (a) and (c)) suggests many false negatives as emails with malicious content are not classified correctly. However, for the data split of Nov, where we have the more training data compared to the testing data, the three classifiers perform well with AUC-PR scores for both data sets above 0.8 (Figure 4 (c)). The classifier performs well for the Botnet data set for attachments as we have many training samples for each month as seen in Table 2.

For emails with URLs, all three classifiers show poor performance with AUC-PR scores (Figure 4 (b) and (d)) around or below 0.5. This means for an email with malicious URLs, the classifiers NB and SVM will label them correctly less than 50% of the time, worse than a random guess. However, we have very high accuracy scores for the classifiers RF and SVM in both data sets (Figure 3 (b) and (d)) for most data splits. The low AUC-PR scores and high ACC scores show the classifiers

cannot distinguish emails with malicious URLs from emails with no malicious URLs. The reason for the poor performance of the classifier is the overwhelming number of emails with no malicious URLs. Our proposed features are insufficient to distinguish malicious URLs as they are underrepresented in the data set, as seen in Tables 1 and 2. This means we cannot determine malicious URLs only from the text of emails with URLs.

In Figure 5, we compare the classification results between our two data sets for the most robust classifier: Random Forest (RF). As discussed above, the comparatively numerous training samples in the Botnet data set allow high classification performance for both AUC-PR and ACC scores. The data split of Nov with the most training samples show high classification scores, especially in the Habul data set, where there fewer data samples. Figure 5 shows the Botnet data set is generally better for predicting malicious content in emails.

Overall, our initial work shows the viability of predicting whether attachments and URLs in emails are malicious. Our proposed feature-rich model shows our hypothesis is true for malicious attachments as those emails can be predicted from the email subject and payload with high AUC-PR and ACC scores. For URLs, the subject and payload of emails do not indicate malicious URLs. In future work, we look to add more features for URLs, focusing on the lexical content (as in related work) to avoid requiring external resources, such as blacklists. Our initial success with predicting malicious attachments reduces the need to scan attachments for malicious content. When the data set is numerous, we can reduce the need to scan over 95% of emails with attachments (from AUC-PR scores) by analysing the text in emails with attachments.

## 8 Discussion

Our initial results are encouraging as they suggest we may be able to correctly identify over 95% of the 6 billion emails with malicious attachments sent everyday (see Section 1) by analysing only the email subject and text content. While our success is not as high with identifying malicious URLs, our results show a manually labelled data set of spam emails with malicious URLs (Habul) can outperform (see Figure 4 (b) and (d)) an automated collection of spam emails with malicious URLs (Botnet). Our results reduce the need to scan large quantities of emails for malicious content

The main advantage of our approach is the self-contained sets of features extracted from only the email itself, without needing external resources such as virus scanners or blacklists. This means our machine learning algorithms can quickly adapt to changes in spam emails and later verify its results when scanners and blacklists have been updated.

A limitation of our approach is the descriptiveness of our proposed sets of features. Our results show that the features are more suitable for predicting malicious attachments than malicious URLs. This suggests emails with malicious URLs do not have sufficient commonalities in the subject or text content to suggest the malicious intent of its URLs. Some exploit kits such as the Blackhole Exploit Kit simply inserts malicious URLs into emails without changing their content (Oliver, et al., 2012). Thus, non-malicious spam emails can become malicious without any changes to their original spam content. To resolve this limitation, in future work we intend to add lexical features from related work (see Section 2.3) and propose our own for URLs, and compare their classification performance.

Another limitation is the possibility of a few spam campaigns being overrepresented in our data sets. We have not performed a detailed spam campaign analysis as it is another research area beyond the scope of this paper. Reviewing statistics from Tables 1 and 2, for the Habul data set, we find 13 unique malicious attachments (in 19 emails with malicious attachments), and 70 unique malicious URLs (in 78 emails with malicious URLs); and for the Botnet data set, we find 847 unique malicious attachments (in 3,008 emails with malicious attachments), and 889 unique malicious URLs (in 1,647 emails with malicious URLs). If each unique attachment or URL represented one spam campaign (thus having similar features in campaign emails), then the diversity of these spam campaigns are high, which strengthens our results as the classifiers can recognise a wide variety of spam campaigns with high AUC-PR and ACC scores for malicious attachments. In future work, we look to address this issue more closely by performing spam campaign analysis to see their influence on classification results.

Overall, we confirm a part of our hypothesis that emails with malicious attachments can be predicted from only the email text features. Our evaluation on two real-world data sets composing of only spam emails, show the effects of data set size, cumulative learning of spam emails over a year, and importance of features for classification. As this work in identifying one of the most dangerous type of spam email continues, we aim to prevent one avenue of cybercrime to expand by limiting exposure of malware to potential victims.

## 9 Conclusion

We presented rich descriptive sets of text features for the task of identifying emails with malicious attachments and URLs. We use two real-world data sets of spam emails, sourced from a manually labelled corpus (Habul) and automated collection from spamtraps (Botnet). Our initial results show that emails with malicious attachments can be predicted using text features extracted only from emails, without requiring external resources. However, this is not the case with emails with malicious URLs as their text features do not differ much from emails with URLs. We compare classification performance for three classifiers: Naïve Bayes, Random Forest, and Support Vector Machine. We compare the performance of features across our two data sets with the generally best performing Random Forest classifier. We discuss the effects of differences in data set sizes, potential overrepresentation of spam campaign emails, and advantages and limitations of our approach. Our initial success suggests we may be able to correctly identify over 95% of emails with malicious attachments without needing to scan the attachments. This is a huge saving in resources and prevention of cybercrime, as estimates show approximately 6 billion emails with malicious attachments are sent every day.

In future work, we look to add features to improve the classification of emails with malicious URLs. We intend to extract more features from the header of emails, such as graph relationships of common (anonymised) email addresses. One important issue for our work is the effects of spam campaigns on classification results, which has not been addressed in related work. We plan a comprehensive analysis methodology with feature combinations, and balancing for data set sizes and spam campaigns. We plan to extend our work to prominent email data sets, such as the Enron email data set.

## 10 Acknowledgements

The research is funded by an ARC Discovery Grant on the Evolution of Cybercrime (DP 1096833), the Australian Institute of Criminology (Grant CRG 13/12-13), and the support of the ANU Research School of Asia and the Pacific. We also thank the Australian Communications and Media Authority (ACMA) and the Computer Emergency Response Team (CERT) Australia for their assistance in the provision of data and support.

## 11 References

Blanzieri, E., & Bryl, A. (2008). A survey of learning-based techniques of email spam filtering. *Artificial Intelligence Review, 29*, 63-92.

Broadhurst, R., Grabosky, P., Alazab, M., Bouhours, B., Chon, S., & Da, C. (2013). Crime in Cyberspace: Offenders and the Role of Organized Crime Groups. *Social Science Research Network (SSRN).*

Canali, D., Cova, M., Vigna, G., & Kruegel, C. (2011). Prophiler: a fast filter for the large-scale detection of malicious web pages. *Proceedings of the 20th international conference on World wide web*.

John, J. P., Moshchuk, A., Gribble, S. D., & Krishnamurthy, A. (2009). Studying Spamming Botnets Using Botlab. *NSDI.*

Kreibich, C., Kanich, C., Levchenko, K., Enright, B., Voelker, G. M., Paxson, V., & Savage, S. (2009). Spamcraft: An inside look at spam campaign orchestration. *Proc. of 2nd USENIX LEET.*

Le, A., Markopoulou, A., & Faloutsos, M. (2011). PhishDef: URL Names Say It All. *Submitted to IEEE INFOCOM.*

Likarish, P., Jung, E., & Jo, I. (2009). Obfuscated malicious javascript detection using classification techniques. *Malicious and Unwanted Software (MALWARE), 2009 4th International Conference on.*

Ma, J., Saul, L. K., Savage, S., & Voelker, G. M. (2009). Beyond Blacklists: Learning to Detect Malicious Web Sites from Suspicious URLs. *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining.*

Ma, J., Saul, L. K., Savage, S., & Voelker, G. M. (2009). Identifying Suspicious URLs: An Application of Large-Scale Online Learning. *Proceedings of the 26th Annual International Conference on Machine Learning.*

Ma, J., Saul, L. K., Savage, S., & Voelker, G. M. (2011). Learning to Detect Malicious URLs. *ACM Trans. Intell. Syst. Technol.*

Martin, S., Nelson, B., Sewani, A., Chen, K., & Joseph, A. D. (2005). Analyzing Behavioral Features for Email Classification. *Second Conference on Email and Anti-Spam (CEAS).*

Masud, M. M., Khan, L., & Thuraisingham, B. (2007). Feature based techniques for auto-detection of novel email worms. *Advances in Knowledge Discovery and Data Mining.*

Oliver, J., Cheng, S., Manly, L., Zhu, J., Paz, R. D., Sioting, S., & Leopando, J. (2012). Blackhole Exploit Kit: A Spam Campaign, Not a Series of Individual Spam Runs. *Trend Micro Incorporated.*

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., . . . Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research, 12*, 2825-2830.

Ramachandran, A., Dagon, D., & Feamster, N. (2006). Can DNSBased Blacklists Keep Up with Bots? *In Collaboration, Electronic messaging, Anti-Abuse and Spam Conference.*

Stone-Gross, B., Holz, T., Stringhini, G., & Vigna, G. (2011). The underground economy of spam: A Botmasters perspective of coordinating large-scale spam campaigns. *In USENIX Workshop on Large-Scale Exploits and Emergent Threats.*

Velasco, S. (2010). Wikipedia vandalism detection through machine learning: Feature review and new proposals. *Lab Report for PAN-CLEF*.

Wang, X., Yu, W., Champion, A., Fu, X., & Xuan, D. (2007). Detecting worms via mining dynamic program execution. *Security and Privacy in Communications Networks and the Workshops, 2007. SecureComm 2007. Third International Conference on.*

West, A. G., & Lee, I. (2011). Multilingual Vandalism Detection using Language-Independent {\&} Ex Post Facto Evidence - Notebook for PAN at CLEF 2011. *CLEF (Notebook Papers/Labs/Workshop).*