Bull. Nov. Comp. Center, Comp. Science, 31 (2010), 29–43

Models and algorithms for detection of spam and senders of spam

Tatiana Batura, Feodor Murzin, Alexey Proskuryakov,
Jennifer Trelewicz

Abstract. In the paper, we give an overview of several approaches that we use to analyze "spam" (undesired bulk advertising) and the credentials of senders of spam, for the purpose of automatic detection. We use some of these approaches for discrimination between stolen account credentials and "spam-bots" (accounts opened purely for the purpose of distributing spam), since the methods for handling spam senders differs by type of account. Other approaches described in this paper we use to automatically detect classes of spam messages, assisting spam analysts in their work to find the messages, so that the messages may be automatically deleted.

1. Introduction

Our work concerns the detection of undesired bulk advertising by Internet ("spam" in the sequel, as in the vernacular), including by email and social networks. As noted by Kaspersky Laboratory [1], spam detected in email traffic exceeded 80% in August 2010. This situation puts a strain not only on the users who must sort through the undesired content in search of legitimate emails, but also on the infrastructure that passes and hosts the email. In effect, one must plan approximately five times more infrastructure for every legitimate message to be handled. Furthermore, much of the spam-advertising is of an illegal nature, for example, promoting pornographic sites or money scams. For this reason, sites may also have legal interest in finding and removing spam from their traffic.

Social networks pose a challenging addition to the question of spam detection. On an email system, in order for one million users to see a given announcement, one million or more copies must be sent. On the other hand, a spam message in a popular chat group on a popular social network may reach the eyes of hundreds of thousands of users with one copy. When spam message detection is done by frequency, the complication posed by social networks becomes clear.

In this paper, we give an overview of several approaches that we use to detect spam and senders of spam. We use some of these approaches for the detection of spam-bots among user accounts, so that we can close the accounts of spam-bots and block the stolen accounts. Other approaches we use to automatically detect classes of spam messages, assisting spam analysts in their work to find the messages, so that the messages may be automatically deleted.

2. A brief overview of spam

Detecting the 80% of spam messages among the billions of messages that cross the Internet each day is not a trivial task. Many anti-spam systems, such as Google Gmail and Kaspersky Internet Security, rely to a large extent on pattern matching. Users register complaints with the hosting company about spam messages, which sends the messages to an automatic processor or to a human analytic. Other messages may be routed to the system or analytic by their frequency; i.e., when the number of copies of the messages sent in a given period of time crosses some threshold. A group of spam messages, based on a URL, pattern of text, or other content, may be identified by a "signature". All of the senders of this signature ("spammers") may then be grouped together. These are very active areas of research, for example as shown in [2-5].

A new anti-spam technique including a mechanism of active feedback from users and Maximum entropy-based spam filtering approach are presented in [2]. Also, there are presented a prototype based on the methods evaluating the technique by using a well-known mail corpus as well as a real dataset collected from an existing mail server.

The maximum entropy approach described in the work exhibits certain advantages. First, it provides for the spam filtering system a multi-dimensional view of the mail traffic by classifying packets according to sets of their features. Second, it detects spam that causes abrupt changes in the incoming mail traffic. A large deviation from the baseline distribution can only be caused by packets that make up an unusual portion of the traffic.

At present spam is often framed by messages that are irrelevant or have little to do with the theme of spam, which makes thematic analysis by current email filtering methods becoming quite unreliable because of a large number of irrelevant words. This impacts the effectiveness of spam filtering directly and increases the difficulty of spam filtering.

Paper [3] describes using an ontology for analysis of semantic elements and bodies in email texts and proposes a method of constructing semantic bodies and calculating ways of similarity between semantic bodies based on sentence similarity.

The approach described in the paper is realized as an on-line commonsense knowledge based on unveiling inter-conceptual relations and inter-attribute relations of concepts as connoting in lexicons of Chinese and their English equivalents.

Concerning the requirement of email filtering to improve the efficiency and accuracy in email mining, topic detection and many other specific applications, learnt from traditional spam filtering methods, an approach based on feature analysis and text classification is proposed in [4]. Utilizing some structural features which are very likely to identify an irrelevant email, such as group sending, embedded pictures, and so on, feature analysis filtering compensates the disadvantage of spending too much in text classification.

An idea of identifying the category of a group-sent mail by the presence of personal names is proposed and the method of email filtering based on URL blacklist is improved. Considering the different contribution of subject and body text to the category, the algorithm of Naive Bayesian email classification is improved.

An attempt to categorize the prevalent popular techniques for classifying email as spam or legitimate is made in [5]. This analysis led to the conclusion that context-based email filtering has the biggest potential in improving quality by learning various contexts, such as n-gram phrases, linguistic constructs or users' profile based context, to tailor their filtering scheme. Statistical approaches such as Naive Bayes classifier, Decision tree, Support vector machine, Fuzzy logic, etc. are examined in the paper. Another examined approach is Context based text classification that takes into account how a word $w1$influences the occurrence of another word $w2$in the document. Thus, the presence or absence of $w1$affects a classification based on $w2$.

This paper presents a quantitative as well as qualitative comparative evaluation of existing text classification methods with focus on email filtering. Additionally, the accuracy results of different text classifiers on different data sets for spam filtering are listed in the paper. The authors also assessed the strengths and weaknesses of each technique when considering its application to email filtering.

Unfortunately, an exhaustive survey on this topic is difficult to make because many sources are unavailable.

Our professional experience shows us that there are several general categories of spam:

1. That distributed by "spam-bots"; i.e., by accounts not associated with real users, but created and used solely for the purpose of distributing advertising;
2. That distributed by stolen accounts; i.e., by accounts associated with real users, whose credentials were somehow learned by a third party, who controls the account to distribute advertising unknown to the account owner;
3. That distributed deliberately by live users, who attempt to build their own business at the expense of the site on which they are spamming.

In our experience, type #3 comprises a minority of all spam. Types #1 and #2 comprise the bulk of spam, where a given spam signature may be distributed only by spam-bots, only by stolen accounts, or by a combination of both.

It is important to be able to distinguish between types #1 and #2, since our actions on the sender accounts vary. We wish to close the accounts of spam-bots, since these are not associated with real users. On the other hand, we block stolen accounts, until such time as the account owner can contact support services to regain control of the credentials. Closing the stolen accounts is not desirable, since it is simply bad business – real users support the business of email and social networking sites. For this reason, a major direction of our work is on distinguishing spam-bots from stolen accounts.

Having the spammers grouped by signature helps detection a great deal. However, some signatures are difficult to establish. Especially in social networking sites, where a small number of copies of a message may reach millions of people, a spammer may slightly alter the content of the message, so that each copy is unique, complicating detection by frequency. As such, we need a method for deriving a more complex content signature, rather than simple text matching. For this reason, we also work in text semantics to extract more complex signatures.

3. Statistical analysis of usernames and logins

3.1. Method based on bigram frequency

In this study, we consider usernames and logins (i.e., email addresses) of users on a popular Russian Internet site, for the purpose of distinguishing spam-bots from stolen user accounts, that are used for spam. Among usernames and logins are recognizable words, such as "Natasha", as well as those chosen by random keyboard entry. Either of these may be potential spam-bots. We do not consider words shorter than a given length; e.g., a 3-letter word may be the first letter of each of family, given, and middle names. For purposes of analysis, we strip from the email address all symbols after the "@", since the domain is chosen by the site and not by the user. We also discard sequences of numbers from the email address, since we do not know the meaning of the number (e.g., random number, birthday, or car registration number), and as such cannot analyze the randomness of the sequence.

The question that we address in the study is to what extent recognizable words may be distinguished from random, or chaotic, sequences. We adapt approaches from synergetics, where normally analysis is performed over long text sequences, rather than the short sequences of our data. An additional specificity of our data set include the need for high-performance algorithms, since we wish to analyze the texts in real time on the working system, where addresses need to be analyzed in quantities of tens of thousands at a time. This condition restricts our ability to perform exhaustive search in dictionaries. Furthermore, since email addresses are written in Latin characters, transliterated from Cyrillic, utilizing one of a myriad of popular methods for transliteration, and since names may be shortened into nick-names in many different ways, the dictionary approach becomes prohibitive.

We propose the following algorithm that we demonstrate with a concrete example on the name "sofia". First we restore the name to Russian, where it is "софия". We treat the subject of transliteration in detail in the sequel. We then take all bigrams of characters from the name, which include со, оф, фи, ия.[1]

In scientific literature [6], there is much data on the frequency of characters in bigrams from the Russian language. To us, the frequencies of the bigrams themselves are of interest. Using information on character frequency from the literature [7], we built a table in the following manner. Given a text, we count the number of different bigrams and add to the table. The first and second characters of the bigram become the row and column of the table, respectively. As a result, we receive a table of the number of occurrences of the characters in bigrams. Frequency is obtained by dividing by the total number of bigrams in the table.

We may use the pseudo-frequency data from the table in our example of "sofia". We receive со – 27, оф – 2, фи – 2, ия – 17, giving a sum of 48. If there were a potentially erroneous sequence, such as "sfaio", obtained by transposition of characters, in Russian we would receive "сфаио", with pseudo-frequencies сф – 0, фа – 2, яи – 3, ио – 8 and sum 13. As such, it appears that a threshold could be used to distinguish recognizable words from chaotic sequences. We note that we do not analyze the frequency of the bigrams or letters in a given word, but instead focus on the degree to which the bigram is "typical" for the text in the given language (here, Russian). Concretely, we choose the pseudo-frequency from the table of typical text.

Returning to the topic of transliteration, it is unavoidable to consider multiple variants. For example, the name "Таня" may be transliterated to "Tania", "Tanya", "Tanja", etc. In Latin-alphabet literature, one encounters the first variant rarely, the second frequently in English transcription, the third in German transcription. Combinations of vowels and consonants give another level of complication. For example, the family name "Кузнецова" may correspond to "Kouznetsova" in French, "Kuznetsova" or "Kuznecova" in English, etc., with the obvious correspondence of Latin and Russian characters. It may also be noted, that the calculated pseudo-frequencies are

[1] Alternatively, so that the fist and last letters of the word are considered equally with the others, it is possible to consider a "quasiperiodic" form of the word formed by periodic repetition, so that the bigrams would include со, оф, фи, ия, яс. For n-grams, when n is relatively large, such approach is particularly useful. The trade-off is in the additional entropy introduced by forming n-grams that do not occur in the language within a given word.

specific to the language, but still give “reasonable” correspondence for other Slavic languages. For example, in Ukranian, “Володимир” is commonly used instead of “Владимир”, “Олександр” instead of “Александр”. As such, it is possible with some degree of accuracy to use the pseudo-frequency data for Ukrainian, Belorussian, Polish, Czech, Slovak, Serbo-Croat, Bulgarian, and Macedonian. In short, the Russian pseudo-frequencies may provide a starting point for work with other Slavic languages. However, the finer points of this extension are beyond the scope of this study.

In the above-described method, it is possible to make separate analyses, based on the frequency of bigrams of the Russian and English languages, since frequently users use English words in their logins. However, one should exercise care when making such extensions, since it is also possible to find German and French words in the logins as well, with corresponding different bigram frequencies.

3.2. Entropy-based criteria

Let w be a word in the alphabet $X = \{x_1, \dots, x_k\}$. Set $H(w) = -\sum\limits_i p_i \log p_i$ the entropy of w, where p_i is the frequency of the character x_i in the word w. Correspondingly, we may calculate $H(W)$, where $W$is a set of words. In this situation, the frequency p_i corresponds to the entire set W.

It is well known that the entropy of a word or set of words increases with the chaos of the words [8], which may be used for the analysis of usernames and addresses. Specifically, such analysis should distinguish between “recognizable” names and chaotic text sequences. For example, we may set a threshold based on $H(w) \geqslant \lambda$, where λ is the threshold of “recognizability”. As such, we raise the question of the choice of the parameter λ.

We consider the following method for choosing λ. We take a subset $W_0 \subseteq W$ of usernames, that we consider to be recognizable, including given names, family names, and their combinations; concrete objects and concepts; characters from films and other artistic works; commonly-used terms in usernames, etc. Then we take $\lambda = H(W_0) + \varepsilon$, where ε is an experimentally-determined additional input parameter, corresponding approximately to the “distance” between order and chaos.

We further note, that it is possible to allow different alphabets for the words. A “natural” choice of alphabet would include letters, digits, and punctuation characters as allowed by the computer registration system, such as “.”, “-“, “ _”. As a result, in the set W_0 we include usernames, in combination with number strings, the potential significance of which are beyond the scope of our analysis, as they may be dates of birth, phone numbers, various special combinations of figures of type 1 ... 1, etc.

Since usernames may be short, a pure entropic approach may need to be adjusted, since it is our experience that normally entropic criteria work best with long sequences.

We may “improve” the statistics by considering the frequency of bigrams or trigrams of characters. As a result, we receive three types of entropy, based on n in our n-grams: $H_1(w) = H(w),\ H_2(w),\ H_3(w)$.

In order to correctly determine $H_i(W),\ i = 2, 3$, we consider $Pairs(W) = \cup\{Pairs(w) : w \in W\}$ and $Triples(W) = \cup\{Triples(w) : w \in W\}$, where $Pairs(W)$ is the union over $w \in W$ of the sets $Pairs(w)$ of bigrams from the set of words W. *Triples*(W) is determined analogously. The number of occurrences of concrete bigrams in W we determine as the sum of the number of occurrences over all $w \in W$. In other words, we count occurrences of bigrams and trigrams separately. We cannot take concatenated words, since on these boundaries, we will encounter new bigrams and trigrams, that are artificially and arbitrarily created by any arbitrary ordering of the set during calculation. Taking all of this into consideration, we may calculate pseudo-frequency and three thresholds $\lambda_i = H_i(W_0) + \varepsilon_i$.

We may also take different criteria for determining if a word is considered to be chaotic:

1) $\exists i\ (H_i(w) \geqslant \lambda_i + \varepsilon_i)$; i.e., one of the three entropies is sufficiently larger than the others;

2) $\forall i\ (H_i(w) \geqslant \lambda_i + \varepsilon_i)$; i.e., all three of the entropies are large;

3) $\sum\limits_i \alpha_i H_i(w) \geqslant F(\lambda_1, \lambda_2, \lambda_3)$, (where α_i are the weight coefficients and F is a given function); i.e., a weighted average of entropies, where the weight coefficients may be chosen heuristically, and the most common F functions include max, min, and mean.

Exploration of the most appropriate of these approaches for our research is a part of our current study.

3.3. Criterion based on an estimation of entropy

Let $w = x_{i_1} \ldots x_{i_n}$ be a word in the alphabet $X = \{x_1, \ldots, x_k\}$. Take $\bar{H}(w) = -\sum\limits_j p(x_{i_j}) \log p(x_{i_j})$, where $p(x_{i_j})$ is the frequency of the character x_{i_j} in the set of words W_0. That is, in the current case we calculate entropy for a given word, on the basis of analysis of the set of words W_0, and from this calculate the value $\bar{H}(w)$. It is possible to say that the value is in the form of entropy. This method may be applied to bigrams and trigrams of characters.

4. Structural and content-based analysis of logins

Content-based analysis of either usernames or login/email may be used for the purposes of separating spam-bots from stolen accounts. At this point in the study, we focus on login, since logins on the system are unique, while usernames may be used an unlimited number of times; i.e., a group of spam-bots may simply copy the usernames of existing users for the purposes of looking "legitimate". There is indeed information to be gleaned from analyzing the username and login together, but that information is out of scope of this paper and reserved for our future study.

4.1. Patterns of names

Logins may be considered from the point of view of their structure. It is possible to exhaustively divide the strings into the following simple classes, which we have found to suit the needs of analysis as detailed in the sequel:

Number – a word made entirely of digits;

Char – a word made entirely of letters;

Char+ – a word made of words, followed by a sequence of characters that do not include letters;

Char(/Char+)Number – Based on our observations, we decided not to consider this as a separate class, since numbers appended to strings tend to be either random sequences (in the cases of many spam-bots) or numeric information of importance to the user, used for the purpose of distinguishing the login from others with a similar string. For example, sasha1984 may append the birth year of 1984 to distinguish the login from that of another Sasha, who registered first. As a result, we do not analyze these suffix numbers, and the resulting logins fall into one of the Char or Char+ classes.

NumberChar(/Char+) – a word consisting of a sequence of digits, followed by a sequence of characters or characters with symbols.

Char(/Char+)NumberChar(/Char+) – a word made of a concatenation of Char or Char+, a Number, and then Char or Char+.

NumberChar(/Char+)Number – a word, analogously made from a Char or Char+ sequence concatenated between two Number sequences.

Other classes, formed from more complex concatenations, we have found to occur with low frequency as "understandable" logins, as compared to random sequences. For this reason, we do not analyze such sequences in the sequel.

First of all, we answer the question as to the types of words encountered, on the basis of these classes.

4.2. Content-based analysis

In the prequel, we discussed statistical methods for separating logins by their degree of chaos. In this section, we discuss the classification of "recognizable" logins and usernames, for the purpose of making finer groupings. At the current time, there is little published research on the connection between usernames or logins and the purpose or tone of the spam sent from them. Given the influence on spam "brand" and other marketing factors provided by the sending account, our thesis is that connection can be made between message content and groups of “recognizable” logins. The result could assist us in locating the harder-to-identify groups of spam-bots.

It is well known, that many users put various pseudonyms in place of their family and given names in either or both of the username and in the email. Frequently the pseudonym relates to profession, personality, concepts, characters from artistic works and computer games, etc. We attempt to classify these strings with the help of an appropriate dictionary.

Classification of usernames differs from that of email addresses. Often usernames as nick names include punctuation, special symbols, mixes of alphabets and upper and lower case, which are not permitted in logins and email addresses. For example, we could consider *John Smith, FantomAS, Bal∼БЕС∼ka.*

If we consider only the lettered portion of the username that is longer than 3 symbols, we raise the following questions:

1. In several situations it is difficult to determine the class to which a given name belongs. For example, English *beauty* could be an abstract quality, as well as a concrete measure, or even the name of a person or character, such as “Black Beauty” from the book. As well, English *glass* can be a category of object (material), as well as a concrete object (drinking glass). As such, a number of words will belong to more than one class.

2. A full name may be shortened or combined in different ways, sometimes including the middle name or a shortened form. This last group is one of the most popular subsets of nick names, both in usernames and in email addresses. For example, *vstasov* is one shortened form of Виталий Стасов, *galseliv* of Галина Селиванова, *niko* of Николай, *seliboba* of Селиванов Борис.

3. There are situations, as discussed in the prequel, where different variants of transliteration are used, as well as intentional or unintentional spelling errors; e.g. *nezabu**t**ka* instead of *nezabu**d**ka, maroz* instead of *moroz.* Although this group is not very large, it is significant in Internet slang, and as such in email and usernames.

4. The existence of frequently-encountered character sequences, on one hand, can signify that the sequence is an existing word (or slang) in a real language. On the other hand, there are a limited number of programs, used by organizations that create spam-bots, that generate spam-bots from a relatively small number of senseless but “real-sounding” strings, such as *lesly, netsy, asiko.*

5. Another method of chaotic word generation is to enter a word in one alphabet, while the computer keyboard is in a different language mode. For example, using English keyboard layout for Russian words gives *hfleuf* instead of *радуга.* These situations are less common in usernames and emails, and are relatively simple to handle. Programs such as Key Switcher automatically detect and translate such strings into the appropriate alphabet.

Table 1. Classifications [9], [10] of logins and parts of usernames containing nouns

Common Nouns							
Concrete/Individual					**Collective**	**Material**	**Abstract**
Objects	**Creatures, Organisms**			**Time**			
	Humans, Status, Behaviour, Profession	**Birds, Insects, Animals**	**Plants**				
robot	nachalnik	wolf	tree	april	humanity	steel	freedom
pencil	mechtatel	zhuk	osoka	sunday	narod	moloko	uzhas
	killer	scorpion	rose	morning	mebel	kirpichi	time
	balamut	rjbka	rediska		herd	amber	revolution
						snow	sunset

Proper Nouns						
Personal				**Geographic, Administrative**	**Titles**	**Time**
Names, Last names	**Names of animals**	**Characters of literature, films, games**	**Nationalities**			
gena	tuzik	dedmoroz	russian	niagara	sony	april
maxim	burenka	cheburashka	american	kremlin	beatles	sunday
natali	bobik	shrek		colosseum	moby	morning
ivanov		babayaga			google	
newton		strashila				

We have noted in our data, that often adjectives appear in usernames and email, both by themselves and together with nouns. These combinations, as well as noun-noun combinations and others, may be written as one word, or split. For example, we may find strings such as *redfox*, *goodfriend*, *pretty*, *deleted.*

Usually for purposes of classification, the meaning of adjectives must be considered in their context [11, 12]. However, in our classification of usernames and emails, there is often no context. Thus, we will refer (associate) an adjective immediately with several groups. Specifically, many adjectives

Table 2. Classifications [11], [12] of user names and logins containing adjectives

Qualitative Adjectives							
Color	**Feelings**	**Condition**	**Quantity**	**Time, Age**	**Shape, Size, Weight**	**Appearance, Qualities**	**Touch, Taste, Smell, Sound**
yellow	happy	unknown	empty	new	round	puzzled	hot
red	nice	dead	heavy	long	petite	magnificent	noisy
black	kind	rich		rapid	tiny	beautiful	delicious
		real		young	big		

Relative Adjectives							
Material	**Place**	**Time**	**Person**	**Concept**	**Purpose**	**Quantity**	**Action**
iron	local	daily	English	scientific	sports	double	training
milk	Moscow	wintry	Russian	artistic	school	binary	excited
icy	river	present					

may express a quality or relation, depending on the context. For example, “сегодняшний день” (today) is relational, while “сегодняшний хлеб” (daily bread) is a quality; (качеств.); “соломенная кукла” (straw doll) is relational, while “соломенные волосы” (straw-coloured hair) is a quality; “обледенелая дорожка” (icy road) is relational, while “обледенелая душа” (icy soul) is a quality.

Many of the quality adjectives have various degrees of comparison: positional (neutral), comparative, final. Comparative and final degrees have different forms of expression; e.g. in English, the best, cleverest, bigger. But exact determination of the degree, based on a series of the adjectives, is very difficult. A series of quality adjectives may indicate a combined quality, that is not expressed by the different levels, including form (round, square), total (eternal, perfect) or partial degree (palish, reddish), incomparable qualities (dead, blind). Obviously, grammatical divisions by degree of comparison and semantics also have their place in the series.

Sometimes, adjectives that extend the noun, such as “goose”, as in “goose egg”, appear in the main series. By comparison with other forms, extension adjectives are encountered not frequently. Moreover, the email addresses and usernames of these forms occur rarely.

5. Analysis of spam text

5.1. Markov chains

Our main goal in this study is to determine if the message under consideration is spam. There are a number of standard methods for such determination, based, for example, on the frequency of the message itself in the system. We focus on one domain (particular type) of spam messages, which presents interesting challenges.

In this study, we make our determination on the basis of pairs of words in the messages; i.e., on a Markov chain [13,14]. Our training data consists of a large collection of sample spam messages of one particular type, concerning a particular telephone scam. We preprocess the messages to remove interjections, greetings/pretexts, formatting, and other "non-content" in the message. Afterwards, we calculate the frequency of word pairs in the series of words in the messages.

Let a message be given from our domain that we will analyze to decide whether it is spam. After the preprocessing, we analyze all pairs of words in the message, calculating their probabilities (on the basis of spam training data), and take the product of these probabilities. If the product exceeds a given threshold, we consider the message to be spam.

5.2. Logical approach

In this section, we address analysis of a message text. Questions of interest include determining the approximate theme of a message; e.g., the message may concern sale of cellular telephones. Alternatively, we may identify the classes of statements in the text, including greeting, apology, recommendation, an appeal to any action, imperative, etc. Such tasks may be formulated with simple predicates.

In the current method, we use predicates in the form $P(w_1, \dots, w_n, t)$, where $w_1, \dots, w_n$ is a sequence of words, and t is the whole text.

The simplest predicates that we use are of the following type:

1. The words $w_1, \dots, w_n$ occur in the text t;
2. Alternate morphological forms of the word w occur in the text t;
3. In the text t occurs a word in which w is a "subword".

The ideal case corresponds to predicate 1 and $n = 1$; i.e., the current word is present in the text in its nominal form. Verification of predicate 2 requires generation of all morphological forms of the word[2], and performing a disjunc-

[2] Alternatively, a morphological analyzer may be used in the preprocessor, so that words are replaced with their normalized forms. However, given the usual slow calculation speed of such processors, because of their complexity and context dependence, we reserve the use of such analyzer to the small lexicon of words that are of direct interest to our work.

tion of the corresponding predicates.

Predicate 3 allows verification of the occurrence of “root words” in compound words, or verbs to which prefixes have been added, which occur frequently in these types of messages.

Consider a tuple of predicates $\langle P_1(\bar{w}_1, t), \ldots, P_1(\bar{w}_k, t)\rangle$, where $\bar{w}_i$ is a collection of words determined *a priori*, and t is the text. We calculate using the true values, obtaining the tuple of zeros and ones $\langle i_1, \ldots, i_k\rangle$, which may be called a vector of flags from the text. The resulting tuples may be clustered by various metrics, for example, by Hamming distance. The choice of most appropriate distance function is a question for our future search, and Hamming distance is chosen here as a simple variant.

Questions of interest include determining the relative volume of various classes of text in a large collection of spam messages; e.g., the percentage of spam messages concerning the sale of counterfeit cellular telephones.

5.3. Comparison with data from the catalog

This method is frequently used with library systems for creating thematic catalogues [15]. In the general case, the catalogue may be structured hierarchically. On the lowest level of the catalogue are the thematic divisions, such as image processing, bioinformatics, etc. for a journal of topics in information technology. Every division is characterized by a collection of keywords.

Assume that we have some text (e.g., an article, preprint, or book) and must classify it into one of the divisions. Normally, only a portion of the text is analyzed, such as chapter titles, annotations, abstract, table of contents. Less frequently the introduction and conclusion may also be analyzed. The text belongs to the division with which it shares the most keywords, where more complex decisions are made, when n >1 divisions provide the best match, or when no suitable divisions are found.

There are various modifications of the current method, such as where keywords have weights and/or formulae of relevancy to the thematic division. The current method may be combined in a straightforward manner with the logical approach in the prequel. Theoretically, the vectors of flags to some extent may play the role of keywords, and the message belongs to the thematic division in which there is minimum distance from the message vector to the keyword vector for the division, with consideration that the result depends strongly on the set of key words and on the chosen threshold. This question is part of our future research.

6. Additional source of information

When the senders of spam are registered on the receiving site (as is often the case with social networks, for example), we are able to use the site’s user databases to obtain information about the users. For our work, we utilize the login of the sender, the IP address of the computer from which the account was registered, the name of the user (given, middle, family), etc.

The Internet provides various resources for obtaining additional data about a user and his data. For example, *WhoIs* in Linux provides detailed information about the owner of an IP address and his geography. This can be useful in identifying a group of spammers, or the stealing of an account (if the geographical use pattern changes suddenly). On the other hand, sophisticated spammers often utilize networks of stolen Internet routers ("bot-nets") for distributing data from geographically-distributed locations at the same time. Bot-nets reduce the amount of information that is readily available for targeting the source of an attack.

Other sources of data on the Internet allow us to analyze given and family names, names of characters, etc. We use some of these sources in building dictionaries for our work. The sources include

- Wikipedia;
- The number of search results given by a popular search engine, such as Google, Yandex, Bing, Yahoo! when we give our term as the search query;
- Suggestions from popular search engines given in response to our query ("Did you mean…"), which can signify misspellings or alternate spellings of common names or terms.

Data that we obtain about logins and user names can be used to assist with classification of these fields and grouping by signature. For example, a group of spam-bots all based on women's given names may be grouped with the assistance of our dictionary.

7. Conclusion

In this paper, we have given an overview of categories of data analysis algorithms that we are applying and adapting in the detection and identification of spam and spam-bots. As shown, the data provide two primary directions for attack, including analysis of usernames and analysis of the sent text. When taken together, the results of analysis on the basis of entropy, correlation, and semantic structure provide strong assistance to human analysts.

The conducted study allowed us to obtain a large amount of data and perform a lot of different calculations. Presenting the results of application of the suggested methods was not among the goals of this paper. In our future publications, we will show the results of applying these algorithms to real data, including hypothesis-based discrimination between spam and not-spam, between legitimate users and spam-bots.

References

[1] "Spam Report: August 2010", SecureList // http://www.securelist.com/en/analysis/204792138/Spam_Report_August_2010

[2] Zhong Shaohong, Huang Huajun, Pan Lili An effective spam filtering technique based on active feedback and Maximum entropy // 7th Internat. Conf. Fuzzy Systems and Knowledge Discovery (FSKD). – 2010. – Vol. 5. – P. 2437–2440.

[3] Zhang Qiu-yu, Yang Hui-juan, Yuan Zhan-ting, Sun Jing-tao Studies on the Semantic Body-Based Spam Filtering // Internat. Conf. Information Science and Management Engineering (ISME). – 2010. – Vol. 1. – P. 233–236.

[4] Xiao Li, Junyong Luo, Meijuan Yin E-Mail Filtering Based on Analysis of Structural Features and Text Classification // 2nd Internat. Workshop Intelligent Systems and Applications (ISA). – 2010. – P. 1–4.

[5] Upasana, Chakravarty S. A Survey on Text Classification Techniques for E-mail Filtering // 2nd Internat. Conf. Machine Learning and Computing (ICMLC). – 2010. – P. 32-36.

[6] Application of Frequency Characteristics of Texts . – http://www.lg–web.chat.ru/texts.html (In Russian)

[7] The Analysis of Texts // http://www.statsoft.ru/home/portal/exchange/textanalysis.htm (In Russian)

[8] Yaglom A.M., Yaglom I.M. Probability and Information. – M.: Nauka, 1973. – 512 p. (in Russian)

[9] Nikonova M.N. Modern Russian: the Manual. – Omsk: Omsk State Tech. Univ., 2008. – 164 p. (In Russian)

[10] The national collection of Russian. – http://www.ruscorpora.ru/corpora-sem.html (in Russian)

[11] Megrabova E.G. About categories of adjectives in English language. – Gorkii, 1976. (In Russian)

[12] English Dictionary. Adjectives by Category . – http://vocabwilleasy.info/word-must-know/adjectives/adjectives-by-category/

[13] MacDonald I.L., Zucchini W. Hidden Markov and Other Models for Discrete-Valued Time Series. – CRC Press, 1997. – 236 p.

[14] Rong Li., Jia-heng Zheng, Chun-qin Pei Text Information Extraction based on Genetic Algorithm and Hidden Markov Model // First Internat. Workshop on Education Technology and Computer Science, Xinzhou Teachers College, China. – 2009 – P. 334–338.

[15] Salton G. Automatic Information Organization and Retrieval. – McGraw-Hill edition, 1968. – 514 p.