A FUZZY SIMILARITY APPROACH FOR AUTOMATED SPAM FILTERING AND NAÏVE BAYES CLASSIFIER

First A. CH. N. V. V. Sivakumar, M.Tech student in Computer Science and Engineering
Dr.MGR Educational and Research Institute Chennai, Tamilnadu, INDIA
Sivakumar.eluru@gmail.com

Abstract

E-mail communication is indispensable nowadays, but the e-mail spam problem continues growing drastically. In recent years, the notion of collaborative spam detection system with a novel e-mail abstraction scheme with near-duplicate matching scheme has been widely discussed. The primary idea of the similarity matching for spam detection is to maintain a known spam database, On purpose of achieving efficient similarity matching and reducing storage utilization, prior works mainly represent each e-mail by a succinct abstraction derived from e-mail content text. However, these abstractions of e-mails cannot fully catch the evolving nature of spams, and are thus not effective enough in near-duplicate detection. In this paper, we propose a novel e-mail abstraction scheme, which considers e-mail layout structure to represent e-mails. We present a procedure to generate the e-mail abstraction using HTML content in e-mail,imap,pop3 and this newly devised abstraction can more effectively capture the A Fuzzy Similarity Aproch For Automated Spam Filtering And Naïve Bayes Classifier is a near-duplicate phenomenon of spams.
Index Terms—Spam detection, e-mail abstraction, near-duplicate matching.

Introduction

E-MAIL communication is prevalent and indispensable nowadays. However, the threat of unsolicited junk e-mails, also known as spams, becomes more and more serious. According to a survey by the website TopTenREVIEWS, 40 percent of e-mails were considered as spams in 2006. The statistics collected by MessageLabs1 show that recently the spam rate is over 70 percent and persistently remains high. The primary challenge of spam detection problem lies in the fact that spammers will always find new ways to attack spam filters owing to the economic benefits of sending spams. Note that existing filters generally perform well when dealing with clumsy spams, which have duplicate content with suspicious keywords or are sent from an identical notorious server. Therefore, the next stage of spam detection research should focus on coping with cunning spams which evolve naturally and continuously.

Although the techniques used by spammers vary constantly, there is still one enduring feature: spams with identical or similar content are sent in large quantities and successively. Since only a small amount of e-mail users will order products or visit websites advertised in spams, spammers have no choice but to send a great quantity of spams to make profits. It means that even with developing and employing unexpected new tricks, spammers still have to send out large quantities of identical or similar spams simultaneously and in succession. This specific feature of spams can be designated as the near-duplicate phenomenon, which is a significant key in the spam detection problem.

In view of above facts, the notion of collaborative spamfiltering with near-duplicate similarity matching scheme has recently received much attention. The primary idea of the near-duplicate matching scheme for spam detection is to maintain a known spam database, formed by user feedback, to block subsequent spams with similar content. Collabora- tive filtering indicates that user knowledge of what spam may subsequently appear is collected to detect following spams. Overall, there are three key points of this type of spam detection approach we have to be concerned about. First, an effective representation of e-mail is essential. Since a large set of reported spams has to be stored in the known spam database, the storage size of e-mail abstraction should be small. Moreover, the e- mail abstraction should capture the near-duplicate phe- nomenon of spams, and should avoid accidental deletion of non-spam e-mails (also known as hams). Second, every incoming e-mail has to be matched with the large database, meaning that the near-duplicate matching process should be substantially efficient. Finally, the latest spams have to be included instantly and successively into the database so as to effectively block subsequent near-duplicate spam.

Although previous researchers have developed various-methods on near-a duplicate spam detection these works

are still subject to some drawbacks. To achieve the objectives of small storage size and efficient matching, prior works mainly represent each e-mail by a succinct abstraction derived from e-mail content text. Moreover, hash-based text representation is applied extensively. One major problem of these abstractions is that they may be too brief and thus may not be robust enough to withstand intentional attacks. A common attack to this type of representation is to insert a random normal paragraph without any suspicious key- words into unobvious position of an e-mail. In such a context, if the whole e-mail content is utilized for hash- based representation, the near-duplicate part of spams cannot be captured. In addition, the false positive rate (i.e., the rate of classifying hams as spams) may increase because the random part of e-mail content is also involved in e-mail abstraction. On the other hand, hash-based text representation also suffers from the problem of not being suitable for all languages. Finally, images and hyperlinks are important clues to spam detection, but both of them are unable to be included in hash-based text representation.

In this paper, we explore to devise a more sophisticated e- mail abstraction, which can more effectively capture the near- duplicate phenomenon of spams. Motivated by the fact that e- mail users are capable of easily recognizing similar spams by observing the layouts of e-mails, we attempt to represent each e-mail based on the e-mail layout structure. Fortunately, almost all e-mails nowadays are in Multipurpose Internet Mail Extensions format with the text/html content- type. That is, HTML content is available in an e-mail and provides sufficient information about e-mail layout structure. In view of this observation, we propose the specific procedure Structure Abstraction Generation, which generates an HTML tag sequence to represent and euseing with splay tree of e-mail.Then SAG focuses on the e-mail layout structure instead of detailed content text. In this regard, each paragraph of text without any HTML tag embedded will be transformed to a newly defined tag<mytext=> and Naïve Bayes Classifing.

Definition1.(<mytext=>) this is newlydefined tag that represents a paragraph of text without any HTML tag embedded.Since we ignore the semantics of the text, the proposed abstraction scheme is inherently applicable to e-mails in all languages. This significant feature is superior to most existing methods. Once e-mails are represented by our newly devised e-mail abstractions, two e-mails are viewed as near-duplicate if their HTML tag sequences are exactly identical to each other. Note that even when spammers insert random tags into e-mails, the proposed e-mail abstraction scheme will still retain efficacy since arbitrary tag insertion is prone to syntax errors or tag mismatching, meaning that the appearance of the e-mail content will be greatly altered. Moreover, the proposed procedure SAG also adopts some heuristics to better guarantee the robustness of our approach.

While a more sophisticated e-mail abstraction is introduced, one challenging issue arises: how to efficiently match each incoming e-mail with an existing huge spam database. To resolve this issue, we devise an innovative tree structure, SpTrees, to store large amounts of the e-mail abstractions of reported spams, and SpTrees contribute to substantially promoting the efficiency of matching. In the design of the near-duplicate matching scheme based on SpTrees, My aim is reducing the number of spams and tags integrating above techniques, in this paper, we designa complete spam detection system A Fuzzy Similarity Aproach For Automatedd Spam Filtering And By Navie Bayes possesses an efficient near-duplicate matching scheme and a progressive update scheme. The progressive update scheme not only adds in new reported spams, but also removes obsolete ones in the database. With A Fuzzy Similarity Approch For Automated spam Filtering and Naiye Bayes for a Classifing a rating spam and up-to-date spam database, the detection result of each incoming e-mail can be determined by the near-duplicate similarity matching process.

To the best of our knowledge, there is no prior research in considering e-mail layout structure to represent e-mails in the field of near-duplicate spam detection. In summary, the contributions of this paper are as follows:

1. We propose the specific procedure SAG to generate the e-mail abstraction using HTML content in e-mail, and this newly devised abstraction can more effectively capture the near-duplicate phenomenon of spams.
2. We devise an innovative tree structure, SpTrees, to store large amounts of the e-mail abstractions of reported spams. SpTrees contribute to the accomplishment of the efficient near-duplicate matching with a more sophisticated e-mail abstraction.
3. We design a complete spam detection system AFSAFASF and NB with an efficient near-duplicate matching scheme and a progressive update scheme. The progressive update scheme enables system AFSAFASF and NB to keep the most up-to-date information for near- duplicate detection.The rest of this paper is outlined as follows: In Section 2, preliminaries including the definition of near-duplicate and the related works are given. In Section 3, we introduce the novel e-mail abstraction scheme.

Preliminaries

In this definition of near-duplicate,this paper, is presented in 3definitions. We then review the related works on spam detection in Section.

Definition1. Near-Duplicate

The central idea of near-duplicate spam detection is to exploit reported known spams to block subsequent ones which have similar content. For different forms of e-mail representation, the definitions of similarity between two e-mails are diverse. Unlike most prior works representing e-mails based mainly on content text, we investigate representing each e-mail using an HTML tag sequence, which depicts the layout structure of e-mail, and look forward to more effectively capturing the near-duplicate phenomenon of spams. Initi- ally, the definition of <anchor> tag is given as follows.

Definition2. (<anchor>). This tag<anchor> is one type of newly defined tag that records the domain name or the e-mail address in an anchor tag.

For example the anchor tag<a href="http://arbor.ee. ntu.edu.tw/index.htm">is transformed to <arbor.ee.ntu. edu.tw>.Theanchortag<ahref="mailto:cytseng@arbor.ee.ntu .edu.tw"is transformed to<cytseng@arbor.ee. ntu.edu.tw>. The purpose of creating the <anchor> tag is to minimize the false positive rate when the number of tags in an e-mail abstraction is short. The less the number of tags in an e-mail abstraction, the more possible that a ham may be matched with known spams and be misclassified as a spam. Therefore, when the number of tags in an e-mail abstraction is smaller than a predefined threshold, for each anchor tag <a>, we specifically record the targeted domain name or e-mail address, which is a significant clue for identifying spams.On the other hand, in this paper, the detailed definition of near-duplicate is given as follows.

Definition3. (Near-Duplicate). Let I ¼ ft1 ; t2 ; ... ; ti ; ... ; tn; <mytext=>; <anchor>g be the set of all valid HTML tagswith two types of newly created tags, <mytext=> and<anchor>, included. An e-mail abstraction derived from procedure SAG is denoted as <e1 ; e2 ; ... ; ei ; ... ; em >, which is an ordered list of tags, where ei 2 I. The definition of near- duplicate is: "Two e-mail abstractions ¼ <a1 ; a2 ; .. . ; ai ; ... ; an > and ¼ <b1 ; b2 ; .. ; bi ; .. ; bm > are viewed as near-duplicate if 8ai ¼ bi and n ¼ m."

Definition4. (Tag Length). This tag length of an e-mail abstraction is defined as the number of tags in an e-mail abstraction we strictly define that two e-mail abstractions are near-duplicate only if they are exactly identical to each other. The major reason is that there are numerous HTML tag patterns appearing commonly and frequently. Partial matching of HTML tag sequences will cause much higher rate of false positive error, and the complexity will be too high to achieve efficient matching. In addition, for further speed-up, while the tag length of an e-mail abstraction is longer, we even apply a looser matching criterion, which does not degrade detection results

Related Works

The e-mail spam problem is increasingly serious nowadays, various techniques have been explored to relieve the problem. Based on what features of e-mails are being used, previous works on spam detection can be generally classified into three categories: 1) content-based methods,2) noncontent-based methods, and 3) others. Initially, researchers analyze e-mail content text and model this problem as a binary text classification task. Representatives of this category are Naive Bayes and Support Vector Machines (SVMs) methods.In general, Naive Bayes methods train a probability model using classified e-mails, and each word in e-mails will be given a probability of being a suspicious spam keyword. As for SVMs, it is a supervised learning method, which possesses outstanding performance on text classification tasks. While above conventional machine learning techniques have reported excellent results with static data sets one major disadvantage is that it is cost-prohibitive for large-scale applications to constantly retrain these methods with the latest information to adapt to the rapid evolving nature of spams. The spam detection of these methods on the e-mail corpus with various language has been less studied yet. In addition, other classification techniques, including markov random field model neural network and logic regression and certain specific features, such as URLs and images have also been taken into account for spam detection.The other group attempts to exploit noncontent information such as e-mail header, e-mail social network and e-mail traffic to filter spams. Collecting notorious and innocent sender addresses from e-mail header to create black list and white list is a commonly applied method initially. MailRank examines the feasibility of rating sender addresses.

Proposed System

E-Mail Abstracttion System

In This E-mail abstract system is introduced by producing SAG. SAG is presented to depict the generation process of an E-mail abstraction. The devised data structures SPTable and SPTree are explained.

SAG (Structure Abstraction Generation)

The SAG is generated byE- mails abstraction generation SAG process is out let of E-mail and SAG is composed of three major ypes 1.TagExtraction Type:This is Extract The every E-mail to Tag formate and tag attributes and attribute values,2.TagReordering Type: This order is based on Tag Extraction process then after arrange in rearanged the Tag process and 3.<anchor>Appending Type:This is linking to another web sites example:spam.com.

```
Procedure SAG
Input: the email with text/html content-type,
       the tag length threshold (Lth_short) of the short email
Output: the email abstraction (EA) of the input email
1   // Tag Extraction Phase
2   Transform each tag to <tag.name>;
3   Transform each paragraph of text to <mytext/>;
4   AnchorSet = the union of all <anchor>;
5   EA = the concatenation of <tag.name>;
6   Preprocess the tag sequence of EA;
7   // Tag Reordering Phase
8   for (each tag of EA) // pn: position number
9     tag.new_pn = ASSIGN_PN (EA.tag_length, tag.pn);
10    Put the tag to the position tag.new_pn;
11  EA = the concatenation of <tag.name> with new_pn;
12  // <anchor> Appending Phase
13  if (EA.tag_length < Lth_short)
14    Append AnchorSet in front of EA;
15  return EA;
End
```

Design Of Sptable And Sptree

This major Aim of thes work to design the innovative data structure to facilitate the process of near-duplicate matching. SpTable and SpTrees are helps to store large amounts of the e-mail abstractions in reported spams. several SpTrees are maintain of the database, and the e-mail abstractions of collected spams are maintained in the corresponding SpTrees. two e-mail abstractions are possible to be near-duplicate only, when the numbers of their tags are identical. if e-mail abstractions extract to tag lengths into diverse SpTrees, the quantity of spams required to be matched will decrease. then SpTree is only maching to one single tag length, Then this SPTree chacking from left root node to left chiled node this Three is easy to catching and effectively finding words, it is too much work is reduced server to maintain spams. SpTable is created to record overall information of SpTrees. The i_{th} column of SpTable links to the root of SpTree_i by a pointer, and e-mail abstractions with tag lengths ranging from 2^i to $2^{i+1}-1$ the corresponding nodes from low levels to high levels. As such, an e-mail abstraction is stored in one path from the root node to a leaf node of SpTree, and hence the matching between a testing E-mail and known spams is processed from root to leaf. The primary goal of applying the tree data structure for storage is to reduce the number of tags required to be matched when processing from root to leaf. Since only subsequences along the matching path from root to leaf should be compared, the matching efficiency is substantiallyincreased.

Robustness Issue

In The main difficulty of near-duplicate spam detection is to withstand malicious attack by spammers. Prior approaches generate e-mail abstractions based mainly has-based content based.

Random Paragraph Insertion

This type of spammer attack is commonly used nowa- days. normal contents without any advertisement keywords are inserted to confuse text-based spam filtering techniques. It is noted that our scheme transforms each paragraph into a newly created tag <mytext=>, and consecutive empty tags will then be transformed to <empty=>. As such, the representation of each random inserted paragraph is identical, and thus our scheme is resistant to this type of attack.

Random Tml Tag Insertion

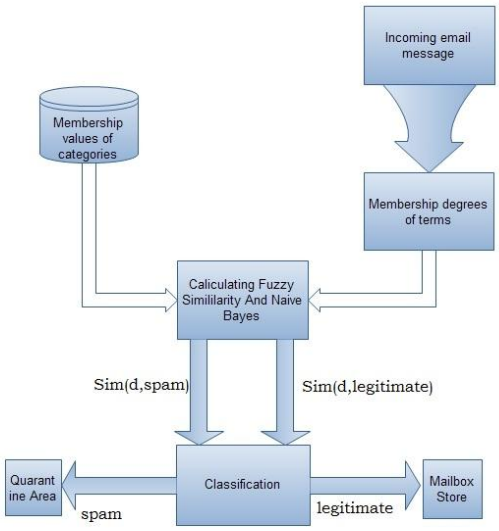
Then spammers know that the how to entering with HTML tag sequences, random HTML tags will be inserted rather than random paragraphs. arbitrary tag insertion will cause syntax errors due to tag mismatching most random inserted tags will be removed, and thus the effectiveness of the attack of random tag insertion is limited.

Sophisticated Html Insertion

The spammers are more sophisticated, then spamers are insert legal HTML tag patterns. if tag patterns that do conform to syntax rules are inserted, they will not be eliminated. However, although some crafty tricks may be conceivable, it is not intuitive for spammers to generate a large number of spams with completely distinct e-mail layout structure.

Naïve Bayes Classifier

A Naïve Bayes classifier is a simple probabilistic classifier based on applying Bayes' theorem with strong (naive) independence assumptions. A more descriptive term for the underlying probability model would be "independent feature model". Naive Bayes belongs to a group of statistical techniques that are called 'supervised classification'



Byes Algorithm

This concept of conditional probability is introduced in elementary statistics. the conditional probability of event is a probability obtained we used P(B/A) to denoted

$$P(B \mid A) = \frac{P(AandB)}{P(A)}$$

This condition probability of B given A can be found by assuming that event A has occurred and working under that Assumption, calculating the probability that event B will occur.

This probability calculation form is comparing with in side mail, when mail box getting mails that mail is divided to tags, then that tags are arranged in preorder, then after this Naives Bayes using a Bayes theorem this Bayes theorem is give some values to every tag then finaly that tag is find the spam or not then spam also having a more then 70 dad word came mean it will show to spam other wise not a spam.

Conclusion

In This filed Of A Fuzzy Ssimilarity Approach For Automated Spam Filtering And Naïve Bayes Classifier Filtering By Near Duplicate Detection, a superior e-mail abstraction scheme Is required to more certainly catch the evolving nature Of spam.com this paper explore more sophisticated and robost e-mail abstraction. The specific procedure Sag is proposed to generate the e-mail abstraction using html content in e-mail, and This newly-devised abstraction can more effectively capture the near-duplicate phenomenon of Spams.this Is designed to efficiently process the near duplicate matching and up dating.

References

[1]. E.Blanzieri and A.Bryl, "Evaluation of The highest Probability SVM Nearest Neighbor Classifier with Variable Relative Error Cost"Proc.Fourth Conf. Email and Anti Spam(CEAS),2007

[2]. M.T.Chang, W.T.Yih and C.Meek, "Partitioned Logistic Regression for Spam Filtering,"Poc 14th ACM-SIGKDD int'l Conf Knowledge Discovery and Data Mining (KDD) pp.97-105,2008

[3]. S.Chhabra, W.S.Yerazunis, and C.Siefkes, "Spam Filtering Using a Markov Random Field model with Variale Weighting Schemas" Proc. Fourth IEEE Int'l Conf. Data Mining (ICDM),pp.347-350,2004.

[4]. P.-A.Chirita, J.Diederich and W.Nejdl, "Mailrank: Using Ranking for Spam Detection" Proc. 14th ACM Int'l Conf.Information and Knowlede Management (CIKM), pp.373-380,2005.

[5]. E.Damiani, S.D.C.di Vimercanti, S.paraboschi, and P.Samarati, "P2P Based Collaborative Spam Detection and Filtering,"Proc.Fourth IEEE Int'l Conf Peer to Peer Computing pp 176-183,2004.

[6]. S.Hershkop and S.J.Stolfo, "Combining Email Models for False Positive Reduction" Proc. 11th ACMSIGKDD Int'l Conf. knowledge Discovery and Data Mining (KDD), pp.98-107,2005.

[7]. J.Hovold, "Naïve Bayes Spam Filtering Using Word-Position-Based Attributes," Proc. Second Conf.Email and Anti-Spam(CEAS),2005.

[8]. A.Kolcz and J.Alspector, "SVM-Based Filteringof Email SpamwithContent-Specific Misclassification Costs"Proc.ICDM Work-shop Text mining,2001.
[9]. A.akolcz, A.Chowdhury, and J.Alspector, "The impact of Feature Selection on Signature-Driven Spam Detection,"Proc.Frist Conf.Email and Anti-Spam(CEAS),2004.
[10]. V.Metsis,I.Androutsopoulos, and G.Paliouras, "Spam Filtering with Naïve Bayes,"Proc.Third Conf.Email and Anti-Spam(CEAS)2006.