利用者からの迷惑メール報告情報の利用

2009SE050 林 幸佑 2009SE109 川地 智久
指導教員：後藤 邦夫

1 はじめに

近年，広告などの目的で，見ず知らずの不特定多数ユーザに送信されるメールが増えている．迷惑メールの相談業務などを行っている日本データ通信協会 [7] によると生産面への被害は約 7,300 億円にも上がることが判明した．具体的な被害の内容は，PC がウイルスに感染してしまうことや迷惑メールを大量に送られることが原因でサーバの処理能力の低下である．

迷惑メールの対策として spamhaus.org[4] や RBL.jp(Realtime Black List)[2] といったブラックリストのデータを共有し，対策をする例がある．先行研究の，青山 尚樹 電子メイルヘッダの調査による spam メイル判定の提案 [5]，丹羽 清志，藤田 公孟，山内 裕太 spam 検知情報の XML による共有 [6] では自動判定を用いて spam 判定をしたが，メールを格納するプログラムが未完であり，構成のみで実験までに至っていないという課題があった．また，自動判定の内容では SPF レコードを判定材料にしているがまだ普及段階にあるので，どのメールにも付いてあるわけではない．そこで本研究ではプログラムによる自動判定ではなく，人の目で実際にメールを読み迷惑メールであるか否かを判断する方法，つまり目視による手動判定によって迷惑メールか否か判定する．

評価基準はメールの格納が正しくされているか，spam メールを spam と正しく判定できているかである．評価方法は，受信したメールの情報とデータベースに格納されている情報を比較し，同一判定と類似判定により評価する．その際に主にヘッダ情報の messageID と本文，subject を使用する．同一判定とは，利用者の受信メール情報とデータベースサーバに格納されている情報が全く同じことを意味し，類似判定は利用者のメール情報がデータベースに格納されている情報に含まれていること，つまり部分的に一致していることを意味する．

予想される成果は先行研究より spam 判定の精度が上がることと，また利用者はメールを受信した際に，その判定を見ることにより本文を読む必要がなくなり仕事の効率化があがると推測できる．したがって，実用性があると考えられる．本研究の利用想定は学校や会社といった組織内であり，少しでも迷惑メールの対処をする時間を省くために利用者がメールを読む際に一目で迷惑メールと判断できるシステムの実現を目的とする．また，エンドユーザへのメリットはこの判定結果を見ればそのメール内容を読まなくてよい点である．本研究で林は主にプログラム作成，川地は主に実験環境と文章作成を担当した．

2 システムの概要

この節では本研究におけるシステムの処理手順を説明する．

2.1 システム全体図

システムの構成を図 1 に示す．

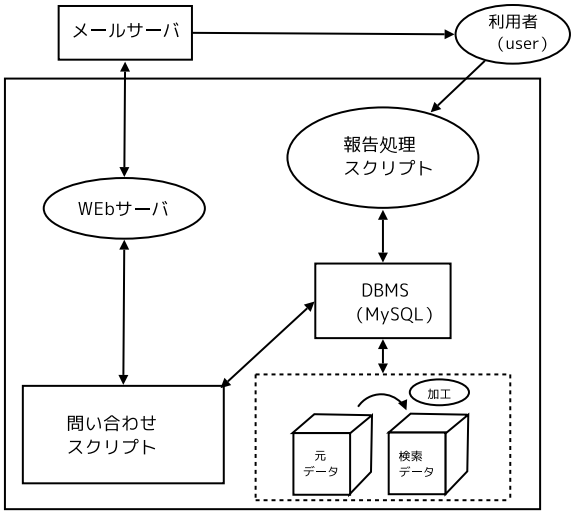

図 1 DB への登録の流れ

1. DB に spam 情報を格納するため，ゼミ生と後藤教授から収集したメールを林と川地が目視し spam と判定したものを DB に格納する．
2. メールサーバが受信したメールの情報をファイル形式で問い合わせ CGI スクリプトに送信する．問い合わせ CGI スクリプトで DB に格納されている情報と一致するものがあるか spam 判定する．
3. spam 判定結果をメールヘッダに追加し，メールサーバに返す．
4. ユーザはその結果が付けられたメールを目視し，万が一 spam と判定したメールは DB に報告し格納情報の更新をする．

2.2 作成するプログラムの仕様

本研究で使用するプログラムは php で作成した．php を使用した理由は，データベースに用いた mysql を使用するための関数が豊富でメール処理においても関数が多く用意されているからである．

- 報告情報処理スクリプト

  受け取ったメール情報を mime decode しメールヘッダ，本文を抽出する．抽出することで格納ができる状態にする．本研究では多くの spam メールを実験で使用するため，データを速く処理でき検索の速度が優れている MySQL が必要であると考えた．本研究ではこれからの実用性を想定し普及している MySQL を使用し，データベースを設計する．工夫点は DB に情報を格納する際に primary key が重複

している場合は Insert(格納) できないため, その項目を count した点である.

- 問い合わせ用スクリプト
  使用言語はシェルスクリプトである．利用者がメールサーバから問い合わせ HTTP 受付スクリプトにメール情報を POST(送信) するためのプログラムである.
- 問い合わせ受付 CGI スクリプト
  こちらも受け取ったメール情報を mime decode し情報ごとに抽出する．その後，MySQL に接続し一致する情報があるか select 文を用い比較する．工夫点は LIKE 構文を使用することで類似判定を可能にした点である.

2.3 サイト内での共有

サイト内共有での手順を図 2 に示す.

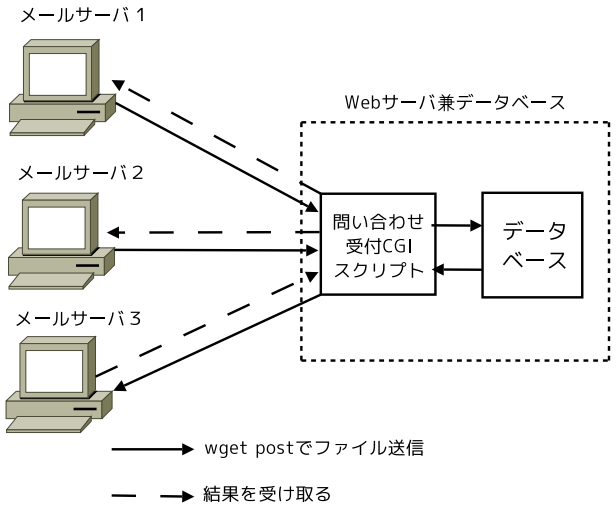

図 2 サイト内での共有のしくみ

1. 利用者はメールサーバが wget POST することでメール情報を送信する.
2. 問い合わせスクリプトが DB サーバにメール情報の検索をする.
3. 判定結果を受け取る.
   1 から 3 の手順をメールサーバ 1，2，3 がそれぞれ実行する．DB を 1 つにすることで，他者との情報の共有を可能にした.

2.4 spam 判定に有効なメール情報

次にメールヘッダの実例を示す．spam 判定に有効なメールの情報 [1][3] は以下である.

1. Received: from 送信元 IP アドレスを表す.
   [133.29.15.21] が送信元 IP アドレス
   by 受信サーバを表す.
2. Date:報告された日付
3. from: 発信者アドレス
4. To: 受信者のアドレス
5. messageID:
   そのホスト中で絶対重複することのない文字列.
   MessageID から世界中のメールを一意に識別することができる.
6. Content-type: ファイルの種類
7. 添付ファイル (multipart/mixed と記述してある):
8. Body:メールの本文．目視に欠かせない重要な項目
9. Subject:メールの件名．目視に欠かせない重要な項目

実例

```
Received: from putter.nanzan.ac.jp([192.168.244
.244])
～中略～
Received: from mail.seto.nanzan-u.ac.jp (mail.s
eto.nanzan-u.ac.jp[133.29.15.21])
by isvw3.ic.nanzan-u.ac.jp (Postfix) with ESMT
P id 35DB41AD13A
for <09se050@nanzan-u.ac.jp>; Thu,6 Dec 2012 1
3:11:27 +0900(JST)  ......(1)
Received: from kosuke-laptop (09se050.G406.set
o-private [10.64.6.11])
by mail.seto.nanzan-u.ac.jp (8.9.3p2/3.7Wpl2)
with SMTP id NAA02179
for <09se050@nanzan-u.ac.jp>; Thu,6 Dec 2012 1
3:11:27 +0900
Date: Thu,6 Dec 2012 13:11:26 +0900   .....(2)
From:09se050 <09se050@nanzan-u.ac.jp> .....(3)
To: 09se050@nanzan-u.ac.jp    ..........(4)
Message-Id: <20121206131126.48c8a838.09se050@n
anzan-u.ac.jp> .........(5)
Subject: hello!!     ................(6)
Mime-Version: 1.0
Content-Type: multipart/mixed;   ......(7)
boundary="Multipart=_Thu__6_Dec_2012_13_11_26_
+0900_t/jYsUmm2XEw.0X"
X-UIDL: 092678b79e33a49dcae572a4f145aa9c
hellohellohellohello!!!!! ..............(8)
--Multipart=_Thu__6_Dec_2012_13_38_41_+0900
_.LAYvkIYMDuEob=1--}
```

(1) の Received 行から重要な箇所を抜き出す.

1. from mail.seto.nanzan-u.ac.jp(mail.seto.nanzan-u.ac.jp[133.29.15.21])
2. by isvm3.ic.nanzan-u.ac.jp(postfix)with ESMTP id 35DB41AD13A
3. for 09se050@nanzan-u.ac.jp: Thu 6 Dec 2012 13:11:27 +0900(JST)

- 1 は送信元 IP アドレスが分かる.
  この場合 [133.29.15.21] である.
- 2 は受信サーバが分かる.
  isvm3.ic.nanzan-u.ac.jp(postfix) がサーバの宛先である.
- 3 は宛先のメールアドレス，処理時刻である.
  09se050@nanzan-u.ac.jp 宛てに 12 月 6 日木曜日 13:11 に届いていると分かる.

本研究ではmessageIDと目視には欠かせない本文，subject を用いる．

2.5 目視による判定

目視とは簡単にいえばメールを読むことである．本文と Subject の内容から迷惑メールか否かを目視して判定する．目視による判定は，

1. DB に情報を前もって格納する
2. 利用者が受信メールを読む

場合である．1 は DB には情報をあらかじめ格納しておく必要があるので，我々の受信したメールや研究室の仲間が受信したメールをもらい目視し spam 判定をする．2 は spam 判定後に利用者が正常なメールを受信した場合である．したがって 1 は我々による目視，2 は利用者による目視を表す．

2.6 テーブル定義

本研究で使用するテーブル定義を次に示す．

テーブル定義

```
messageID varchar(200) 主キーに指定
mail1 varbinary(8000) binary 型で格納
count int(20)
Received text
from varchar(100)
To  varchar(100)
Date varchar(100)
subject varchar(200)
honbun varbinary(400)
```

ここからメール情報をもとに select 文で検索する．select messageID from mailbox を実行すると登録されている messageID の一覧が出る．条件を指定する場合は select * from mailbox where 条件 =’ 文字列’ とすることでデータの中にある条件にあったデータを抽出することができる．

3 spam 判定

この章では spam 判定方法の同一判定と類似判定について説明する．この判定は問い合わせスクリプトがメール情報の抽出をし DBMS に接続した後，DBMS に受信メールの情報と一致する情報がないか select 文で検索する際に用いられる．spam 判定の流れを図 3 に示す．

1. spam と判定するのに重要な情報の messageID を比較する．一致した場合 spam と返す．
2. そこで一致する情報がなければ次に subject を比較する．一致した場合 spam と返す．
3. 本文 (部分文字列) の同様に比較する．一致した場合 spam と返す．
4. すべてにおいて一致しない場合は正常なメールです，と結果を返す．

3.1 同一判定（完全一致判定）

ヘッダ情報の messageID に対してこの判定方法を用いる．同一判定は一字一句相違なく一致することであるので，1 文字でも違っていれば同一ではない．messageID に対して同一判定を行う理由は，本文や subject と違い単語など含まれていない上に，形が決まっているからである．

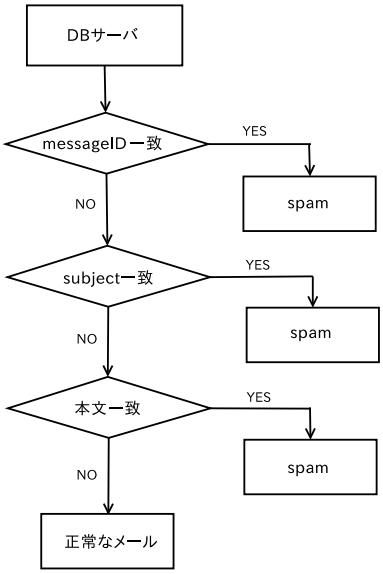

図 3 同一判定の流れ

3.2 類似判定（部分一致判定）

次に類似判定の方法を説明する．ヘッダ情報の subject，本文に対してこの判定方法を用いる．類似判定は部分的に一致している場合の判定だが同時に同一判定も兼ねている．DB に格納されているデータはその 1，受信メールはその 2 に示す．

その 1

```
subject:高額当選おめでとうございます
Date:Mon,17 Dec 2012 09:40:17 +0400
MIME-Version:1.0
Body:1 億円
```

その 2

```
Subject:高額当選
Date:Mon,17 Dec 2012 01:28:38 -0300
MIME-Version: 1.0
Body:1 億円当たりました
```

次に用意した SQL 文を示す．

SQL 文

```
select * from テーブル名
where subject LIKE ’%変数 1%’
select * from テーブル名
where 本文    LIKE ’%変数 2%’
```

この SQL 文では変数 1 は高額当選，変数 2 は 1 億円である．変数 1 の場合，新着メールの Subject の文字列が DB に格納されているデータの Subject の文字列に含まれているので部分一致となり，変数 2 の場合は本文の文字列が DB に格納されているデータの本文の文字列に含

まれていないので部分一致ではない．

4 実験

それぞれの実験を順に示す．実験の際の利用者 (user) は林と川地が担当する．学内の PC(spamproc)1 台を使用する．

- 実験 1
  目視により spam と判定したメールのヘッダ部分を添付ファイルに入れ，spam-report@h303s0.sd.nanzan-u.ac.jp に送信する．その際．報告処理スクリプトで必要なヘッダ情報を抽出し DB に格納する．そして抽出した情報が DB に格納されているかを確認する．
- 実験 2
  メールサーバから DB サーバへの問い合わせと応答をする．DB に格納してある情報と受信メールを比較するために DB サーバに問い合わせスクリプトで情報を検索する．検索した情報と受信メールを比較し spam 判定する．その際に正常メールの場合は No plobrem，spam の場合は spam と付けてメールサーバに返す．川地，林が spam 判定の結果が正しいかどうかを目視し確認する．
- 実験 3
  連携サイト内での共有をする．

5 実験結果と評価

OS はメールサーバ用と利用者用共に Ubuntu10.04 を使用する．

- 実験 1 の結果と評価
  研究用のメールはゼミ生と後藤教授が収集した 300 通を使用する．そのメールを林と川地 (利用者) がまず目視して spam 判定する．そして spam と判断したメールを報告しプログラムに渡して 200 通を DB に登録できた．
- 実験 2 の結果と評価
  wget POST で問い合わせスクリプトにファイル送信成功した．メールは 100 通を用意した．

問い合わせ判定結果は次に示す．

実行結果

```
messageID 完全一致:0 件
subject 完全一致:12 件
subject 部分一致:4 件
本文完全一致:12 件
本文部分一致:17 件
正常メール:55 件
```

この結果が正しいことを目視して確認した．messageID 完全一致 0 件と本文完全一致 12 件，本文部分一致 17 件，そして subject 完全一致 12 件のメール内容を確認した結果，spam メールと判断できる内容であった．しかし，subject 部分一致のメールの内容を確認した結果，4 件のうち 3 件は spam メールだと判定した．残りの 1 件を確認した結果，部分一致はしているが，正常な内容であった．また正常メール 55 件を確認した結果，50 件は spam であった．その具体例を次に示す．subject が “招待状” の受信メールと “招待状が高木さんから届いています” の DB 格納情報と部分一致した．しかし受信メールの内容を確認すると，友人から届いた誕生日パーティーの招待メールであった．本文での判定と subject 完全一致判定では一致件数は少なかったが，spam メールだけを判定できた．subject 部分一致ではより多くの件数で一致したが，同時に正常なメールも spam と判定した．

この結果から，格納しているメール報告情報データの数を増やせば精度が上がると予想される．また自動判定より目視による手動判定が信頼性があると考えられる．

- 実験 3 の結果と評価

ゼミ生が 2.3 の手順 1，2，3 を実行した結果，spam 判定結果が返ってきた．

6 おわりに

本研究はデータベースに格納されている情報が 200 通分だったが，格納情報を増やすことでより多くの種類の受信メールを spam 判定できるようになるので実用性があると考えられる．また今後の課題として

- 問い合わせプログラムの品質の向上
- 部分一致判定の精度向上
- 格納テーブルの見直し，改善
- メールサーバを設置しての実験

という点があげられる．これらの課題を解決することで spam 判定と共有の精度が上がると考えられる．

参考文献

[1] Crocker, D. H.: Standard for the format of apra internet text messages RFC822 (accessed Dec. 2012).

[2] Hart Computer, Inc. and Volunteers: RBL.JP プロジェクト, (accessed Aug. 2012). http://www.rbl.jp/.

[3] MEMORVA: メールのヘッダ情報の意味・見方・調べ方 (accessed Dec. 2012). http://memorva.jp/internet/spam_virus/mail_header.php.

[4] spamhaus: spamhaus: The Spamhaus Project, (accessed Aug. 2012). http://www.spamhaus.org.

[5] 青山 尚樹：電子メイルヘッダの調査による spam メイル判定の提案，南山大学 数理情報学部 情報通信学科 2011 年度卒業論文 (2012).

[6] 丹羽 清志，藤田 公孟，山内 裕太：spam 検知情報の XML による共有，南山大学 数理情報学部 情報通信学科 2011 年度卒業論文 (2012).

[7] 日本データ通信協会：迷惑メール相談センター (accessed Aug. 2012). http://www.dekyo.or.jp/soudan/report/effective.html.