

Detecção de *Spams* Utilizando Conteúdo Web Associado a Mensagens

1

Abstract. *In this paper we propose a strategy of spam classification that exploits the content of the Web pages linked by e-mail messages. We describe a methodology for extracting pages linked by spam and we characterize the relationship among those pages and the spam messages. We then use a machine learning algorithm to extract features found in the web pages that are relevant to spam detection. We demonstrate that the use of information from linked pages can significantly outperforms current spam classification techniques, as portrayed by Spam Assassin. Our study shows that the pages linked by spams are a very promising battleground, where spammers do not hide their identity, and that this battleground has not yet been used by spam filters.*

Resumo. *Neste trabalho propomos uma estratégia de detecção de spams que explora o conteúdo das páginas Web apontadas por mensagens. Descrevemos uma metodologia para a coleta dessas páginas, caracterizamos a relação entre as páginas e as mensagens de spam e, em seguida, utilizamos um algoritmo de aprendizado de máquina para extrair as informações relevantes para a detecção de spam. Mostramos que a utilização de informações das páginas mencionadas melhora significativamente a classificação de spams e hams, gerando um baixo índice de falsos positivos. Nosso estudo revela que as páginas apontadas pelos spams ainda são um campo de batalha não explorado pelos filtros, onde os spammers não se preocupam em esconder a sua identidade.*

1. Introdução

Spam é um problema que tem acompanhado o desenvolvimento e popularização da Internet e tem sido um meio usual de enviar mensagens relacionadas à obtenção de dados pessoais com objetivos ilícitos (*phishing*) e para a disseminação de códigos maliciosos [Milletary 2005]. Os servidores de correio eletrônico têm que lidar com o fato de que entre 82% e 92% das mensagens recebidas são *spam* [MAAWG 2009]. O prejuízo que essa prática acarreta à sociedade é avaliado em bilhões de dólares [Sipior et al. 2004].

O problema do *spam* é análogo a uma corrida armamentista (chamada comumente de *spam arms race* [Guerra et al. 2010]). Isso significa que há uma evolução constante tanto de técnicas de detecção de mensagens indesejadas como da sofisticação das tecnologias adotadas pelos *spammers*. Nesta corrida, cada um tenta se sobrepor ao outro e a mudança na estratégia de um lado induz a mudanças na estratégia do adversário. Os filtros anti-*spam* adotam, comumente, estratégias baseadas em filtragem de conteúdo de mensagens, como o *Spam Assassin* [SpamAssassin 2008] e listas de bloqueio [Cook et al. 2006]. As duas estratégias são complementares, uma vez que a primeira trata do conteúdo da mensagem em si e a segunda trata das estratégias que o *spammer* utilizou para disseminar a mensagem. Vale notar que o objetivo final do *spammer* é

ser atrativo o suficiente para o receptor tomar alguma ação — seja esta comprar algum produto ou seguir algum elo de navegação. Os filtros baseados em conteúdo obrigam o *spammer* a ofuscar suas mensagens, de forma que o *spammer* tem um compromisso entre manter o e-mail legível (e atingir menos caixas de entrada) e comprometer a legibilidade, possivelmente atingindo mais usuários. As estratégias dos *spammers* para evitar as listas de bloqueio, por outro lado, não comprometem a “qualidade” das suas mensagens.

Com o advento e popularização de técnicas de contaminação de máquinas por códigos maliciosos que podem transformar qualquer máquina de usuário em um *bot*, uma ferramenta para redistribuição de *spam* (entre outros usos), estratégias baseadas em listas de bloqueio têm se tornado menos eficientes [Ramachandran et al. 2006]. A iminente troca de versão do protocolo IP (da versão 4 para a versão 6) provavelmente criará dificuldades ainda maiores para o sucesso das listas de bloqueio, uma vez o aumento da faixa de endereços disponíveis tornará mais difícil manter as listas de bloqueio atualizadas.

Em um trabalho recente, Guerra et al. (2010) caracterizaram a adaptação dos filtros em relação às estratégias utilizadas por *spammers* e mostraram como certas características são exploradas ao longo do tempo. Um aspecto do *spam* que ainda não é explorado pelos filtros e, conseqüentemente, não é ofuscado pelos *spammers*, é o conteúdo das páginas Web apontadas pelas *URLs* contidas nos *spams*. Muitas vezes essas páginas estão até mesmo fora do controle dos *spammers*, pertencendo a empresas externas que os contratam para divulgar seus produtos. Pu e Webb (2006) mostraram que pelo menos uma *URL* aparece em 85% a 95% dos *spams* presentes em todos os meses analisados por eles. Já Guerra et al. (2008) reportam que 96,5% das campanhas de *spam* observadas por eles continham pelo menos uma *URL*. Esse número indicam que técnicas que considerem o conteúdo das páginas como evidência para detecção e mitigação do *spam* pode ter impacto bastante positivo. Neste trabalho, mostramos que essas páginas podem oferecer informações valiosas acerca da natureza dos *spams*.

Utilizamos duas bases de dados históricas de *spams* e mensagens legítimas (também chamadas *hams*), *SpamArchive* [Guenter 2010] e *Spam Assassin*¹, para construir uma base de dados que relaciona mensagens e páginas. Utilizando essa base como estudo de caso, mostramos que a utilização das páginas melhora a detecção de *spam* em aproximadamente 10%, sem causar um aumento no índice de falsos positivos. As contribuições deste trabalho, portanto, são (i) a disponibilização de uma base de dados que relaciona mensagens de *spam* às páginas apontadas por elas e (ii) a proposta de uma metodologia para a detecção de *spam* através do conteúdo das páginas apontadas pelas mensagens. Mostramos que as páginas apontadas por mensagens são um campo de batalha promissor e que a informação proveniente das páginas ainda não é explorado pelos filtros.

Este artigo descreve os principais resultados de um projeto de iniciação científica realizado no segundo semestre do ano de 2010. O aluno de iniciação científica, autor deste artigo, foi o idealizador e principal responsável pela realização deste projeto de pesquisa, que foi orientado por tres professores.

2. Trabalhos Relacionados

Em [Upasana and Chakravarty 2010], é apresentado um *survey* sobre técnicas de identificação de *spam* baseadas em classificação de texto. Um filtro que faz uso de grande

¹Disponível em <http://spamassassin.apache.org/publiccorpus/>

parte das estratégias conhecidas atualmente é o *Spam Assassin* [SpamAssassin 2008], que utiliza filtros bayesianos e listas de bloqueio DNS. Além disso, o *Spam Assassin* também conta com um conjunto de regras, geralmente representadas por expressões regulares, que são comparadas com os campos *body* ou *header* de cada mensagem. Ou seja, o *Spam Assassin* é um filtro que lida tanto com características do corpo da mensagem quanto características de rede.

Em [Ramachandran et al. 2006], foi feito um estudo sobre a efetividade de listas de bloqueio baseadas em *DNS* em relação a *botnets*. Os resultados preliminares indicam que apenas 5% de todos os *IPs* dos bots estudados apareciam na lista de bloqueio utilizada. Em [Sinha et al. 2008], é feita uma avaliação de várias listas de bloqueio, e mostra-se que as listas de bloqueio apresentam um número significativo de falsos negativos e falsos positivos. Devido aos problemas potenciais de listas de bloqueio, é necessário a descoberta de novas técnicas.

Em [Webb 2006] construiu-se uma base de dados com páginas apontadas por *spams* da base de dados *Spam Archive* [Gunter 2010] no período entre novembro de 2002 e janeiro de 2006. Porém, essa base de dados tem como foco *Web Spam*, e não relaciona cada página a uma mensagem de *spam*, de forma que não pudemos utilizá-la.

Dessa forma, nosso trabalho é o primeiro, até onde sabemos, a empregar o conteúdo de páginas como evidência para identificação de *spams*, e o primeiro a disponibilizar uma base de dados que relaciona *spams* a páginas *Web*.

3. Detecção de Spams utilizando conteúdo Web associado a mensagens

A técnica para detecção de *spam* que propomos se baseia nas páginas apontadas por *URLs* em mensagens de *spam*. Apesar de o acesso a essas páginas implicar em um custo extra, em uma implementação em produção nossa técnica poderia funcionar de forma complementar a outras estratégias de classificação de *spam*. Ao se analisar uma mensagem, carrega-se as páginas identificadas por *URLs* contidas na mesma e verifica-se se essas páginas possuem conteúdo que seja associado com campanhas de *spam* — da mesma forma que um filtro de conteúdo avalia o corpo da mensagem, mas nesse caso considerando o conteúdo da página. Esse conteúdo é então combinado com as outras características da mensagem (dadas pelo *Spam Assassin*) e o par (mensagem, página) é classificado para identificar *hams* e *spams*. Nesta seção descrevemos as operações em cada uma dessas etapas. O processo é ilustrado pelo diagrama apresentado na Figura 1 e pelo exemplo apresentado ao final.

3.1. Processamento da página

Após a identificação das *URLs* nas mensagens e o acesso às páginas por elas identificadas, removemos todas as *tags* HTML e *scripts*, deixando apenas as palavras que o usuário veria se o mesmo carregasse a página.

3.2. Classificação da página

Para classificar a página, utilizamos um algoritmo de aprendizado associativo sob demanda [Veloso et al. 2006]. Optamos por esse algoritmo por (*i*) ter bom desempenho para utilização em tempo real (o algoritmo consegue classificar, em média, 111 páginas

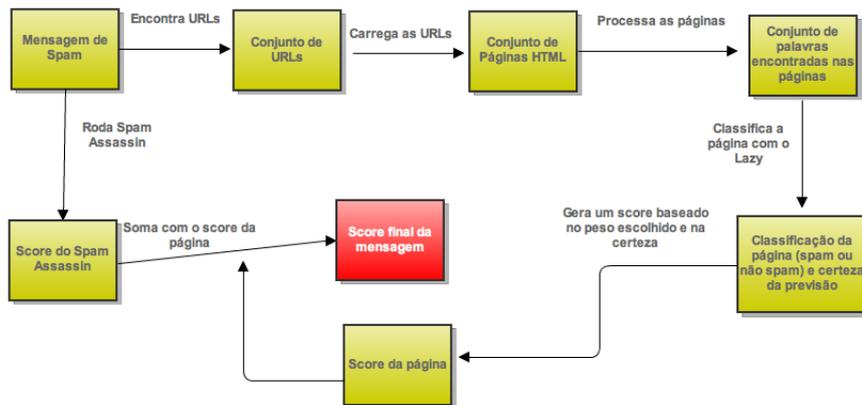


Figura 1. Diagrama ilustrativo do processo de detecção de spams proposto.

por segundo), (ii) gerar um modelo de boa legibilidade (que pode ser facilmente transformado em um conjunto de expressões regulares, como as do *Spam Assassin*) e (iii) ser bem calibrado [Veloso et al. 2008]. Essa última característica significa que o algoritmo gera uma probabilidade de cada previsão estar certa, ou seja, as previsões com mais certeza são mais confiáveis. O algoritmo produz regras do tipo $\chi \rightarrow c$, onde χ é um conjunto de termos e c é a classe (*spam* ou *ham*). Cada uma dessas regras tem uma certa frequência (que chamamos de suporte) e uma confiança, que é dada pelo número de instâncias que são classificadas corretamente pela regra dividido pelo número de instâncias que contém o conjunto de termos χ . O resultado final da classificação de cada página é a classe predita pelo algoritmo e a certeza da predição, medida entre 0 e 1. Quanto maior o valor da certeza, mais confiável é a predição.

Uma das dificuldades da detecção de *spam* é a assimetria entre o custo associado aos diferentes tipos de erro. Um falso negativo simplesmente causa alguma irritação, *i.e.*, o usuário recebe uma mensagem indesejada. Por outro lado, um falso positivo pode ser crítico: uma mensagem importante pode nunca chegar à caixa de entrada do usuário, se for filtrada pelo servidor [Fawcett 2003]. Em virtude do alto custo de se classificar uma mensagem legítima como *spam*, empregamos também a noção de *custo de classificação*. O custo de uma classe mede o quão caro é classificar incorretamente uma instância dessa classe. Ao ponderar todas as regras obtidas para uma determinada instância, o algoritmo faz uma soma ponderada das regras levando em conta a confiança e o custo de cada classe, de forma a valorizar mais regras que apontam para classes de custo mais alto. Isso implica que com o aumento do custo, mais indícios são necessários para que o algoritmo classifique uma página como *spam*.

Como o classificador associativo necessita de instâncias de ambas as classes (*spam* e *ham*), utilizamos uma base de dados pública de *hams*, fornecida pelo próprio *Spam*

Assassin, como já citado. Para instâncias da classe *spam*, utilizamos a base de dados descrita na seção anterior.

3.3. Classificação da mensagem

Associamos um peso p à classificação da página (que será somado à classificação da mensagem depois). A pontuação final da página é dada por $p * c$, onde c é a certeza da predição, de forma que páginas com maior chance de serem identificadas como *spam* acarretam em pontuações mais altas para as suas mensagens. Em seguida, a pontuação da página é somada com a pontuação da mensagem (dada pelo *Spam Assassin*). Vale notar que se o classificador associativo classifica uma página como *ham*, a pontuação da página, que é somada à pontuação do *Spam Assassin*, é negativa.

3.4. Exemplo ilustrativo

Apresentamos um exemplo da aplicação da nossa técnica passo a passo. Escolhemos uma mensagem de *spam* obtida do *Spam Archive* no mês de outubro de 2010. A Figura 2 mostra o corpo da mensagem (omitimos grande parte dos cabeçalhos por questão de espaço). Percebe-se que a mensagem é bem concisa e ofuscada.

```

From: Discount Rolex_Etc. <8-hef@hef.fr>
To: ----
Subject: 75% Off on Gucci, Rolex, And Louis Vuitton Handbags and Various Other items
Date: Tue, 26 Oct 2010 19:20:16 +0300
MIME-Version: 1.0
Content-Type: multipart/alternative;
  boundary="-----_hnhlkay_21_61_99"
X-Priority: 3
X-Mailer: guddj 10
Message-ID: <4281462511.JBNTUZ0E381515@hwxuglmwirrh.fnezaq.biz>

-----_hnhlkay_21_61_99
Content-Type: text/plain;
  charset="windows-1250"
Content-Transfer-Encoding: quoted-printable

Stop paying more than you have to!
http://migre.me/1JHmb
-----_hnhlkay_21_61_99
Content-Type: text/html;
  charset="windows-1250"
Content-Transfer-Encoding: quoted-printable

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML><HEAD>
<META http-equiv=3DContent-Type content=3D"text/html; charset=3Dwindows-1-
250">
<STYLE></STYLE>
</HEAD>
<BODY>
<a href=3D"http://su.pr/6uua5y">Stop paying more than you have to!</a>
</BODY></HTML>

-----_hnhlkay_21_61_99--

```

Figura 2. Mensagem de *spam* extraída do *Spam Archive*

O *Spam Assassin*, com listas de bloqueio ativadas, encontra as seguintes regras:

Regra	Significado	Pontuação
HTML_MESSAGE	Há HTML na mensagem	0.001
RCVD_IN_BRBL_LASTTEXT	Lista de bloqueio DNS BRBL	1.644
URIBL_BLACK	URL em lista de bloqueio	1.775

A pontuação resultante é 3,4 – insuficiente para classificar a mensagem como *spam*. Um excerto da página apontada pelas *URLs* dessa mensagem é ilustrado pela Figura 3.



Figura 3. Pagina de spam apontada pela mensagem extraída do Spam Archive

Percebe-se, neste caso, que o conteúdo da mensagem e o conteúdo da página são totalmente diferentes. O conteúdo da página é transformado, então, em um conjunto de palavras (através do navegador *lynx*), que é entregue ao classificador associativo, que já dispunha de um conjunto de páginas de *spam* e não-*spam* como treino (no caso, as outras páginas armazenadas do *Spam Archive* e a base de dados de *ham* do *Spam Assassin*). O classificador associativo encontra um conjunto de regras, das quais alguns exemplos são:

Regra	Suporte	Confiança
<i>viagra</i> → <i>Spam</i>	36.70%	99.84%
<i>levitra</i> → <i>Spam</i>	34.01%	99.90%
<i>rather</i> → <i>Ham</i>	2.97%	67.30%

Por fim, o resultado final do classificador associativo é que a página é *spam*, com 90% de certeza. Supondo que tenhamos pré-definido que o peso das páginas seria 4, a pontuação da página seria $4 * c$, sendo c a certeza do classificador associativo. Percebe-se que o valor de c é determinante na pontuação final da mensagem, de forma que as páginas que o classificador associativo tem menos certeza recebem uma pontuação menor. Essa página, portanto, teria pontuação igual a 3,6. Somando-se a pontuação obtida pelo *Spam Assassin* com a pontuação da página, temos uma pontuação igual a 7,0 – mais do que suficiente para classificar a mensagem como *spam*.

4. Coleta da Base de Dados de Páginas de Spam

Em [Pu and Webb 2006], mostrou-se que pelo menos uma *URL* aparece em 85% a 95% das mensagens de *spam* no *Spam Archive* no período entre 2004 e 2006, enquanto [Guerra et al. 2008] reportaram que 96,5% de suas campanhas continham pelo menos uma *URL*. Apesar disso, a coleta de páginas de *spam* ainda é uma tarefa desafiadora. [Anderson et al. 2007] mostra que poucas páginas têm um tempo de vida maior do que 13 dias, ou seja, a coleta das páginas tem que ser feita em um período próximo do instante que a mensagem foi disseminada.

Entre julho e dezembro de 2010 nós ocoletamos as mensagens de *spam* da base de dados *Spam Archive* diariamente (a base também é atualizada diariamente), de forma a obter as mensagens de *spam* mais recentes. Em seguida, extraímos as *URLs* do corpo das mensagens e utilizamos expressões regulares simples para remover imagens e executáveis. Em seguida, carregamos e armazenamos as páginas. No caso de mensagens que continham múltiplas *URLs*, todas as *URLs* foram carregadas e armazenadas. Várias *URLs* continham redirecionamentos; nesse caso, seguimos todos os redirecionamentos e armazenamos o conteúdo final da página.

As características da base de dados obtidas são mostradas na Tabela 1. Percebe-se que a maioria das mensagens contém poucas *URLs*. Vale notar que só consideramos como parte da base de dados as mensagens para as quais pelo menos uma página foi baixada.

Tabela 1. Descrição da base de dados obtida

Número de mensagens	63.034
Número de páginas	157.114
Número médio de páginas baixadas por mensagem	2,49

5. Resultados e Discussão

Para avaliar a aplicabilidade de se construir filtros anti-*spam* a partir do conteúdo das páginas, selecionamos todas as páginas únicas da base de dados. Optamos por avaliar apenas as páginas únicas para impedir que uma campanha de mensagens apontando para a mesma página enviesasse os nossos resultados. Quando várias mensagens diferentes apontavam para a mesma página, uma delas foi selecionada aleatoriamente para a avaliação, de forma que apenas uma instância de cada página permanecesse na avaliação. Ao final, portanto, avaliamos a nossa técnica em 32929 páginas *spam*, apontadas por 12111 mensagens de *spam* e 11134 páginas *ham*, apontadas por 4927 mensagens retiradas da base de *ham* do *Spam Assassin*. Utilizamos validação cruzada para a avaliação, dividindo as páginas em 5 partições. Nas subseções seguintes mostramos o impacto da variação dos parâmetros peso e custo, além da comparação da nossa técnica com o *Spam Assassin* puro. O classificador associativo foi executado com confiança 0.3, o custo das duas classes foi igual e o peso escolhido foi 4, exceto quando indicado diferente. Esses valores foram ajustados na validação cruzada.

5.1. Impacto do parâmetro peso na detecção de *spam*

Mostramos na Figura 4 o impacto de diferentes valores de peso (que é multiplicado com a certeza da previsão do classificador associativo) no índice de falsos negativos e falsos positivos. Mostramos também na figura o índice de falsos positivos e falsos negativos gerados através da utilização do *Spam Assassin* sem a nossa técnica, para fins de comparação. Percebe-se que com um peso de até 4, o índice de falsos positivos permanece praticamente igual ao índice de falsos positivos do *Spam Assassin*, embora o índice de falsos negativos seja consideravelmente mais baixo. Selecionamos para os experimentos seguintes, portanto, o valor de peso 4, que representa o menor índice de falsos negativos sem aumentar o índice de falsos positivos.

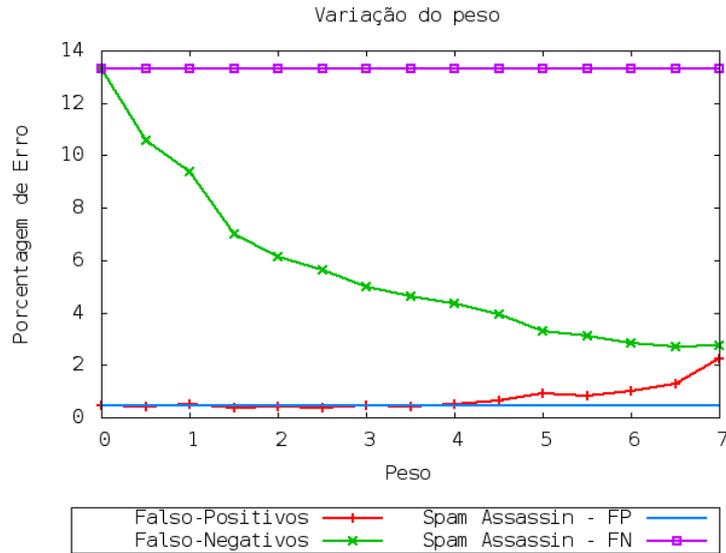


Figura 4. Falsos positivos e falsos negativos x Peso

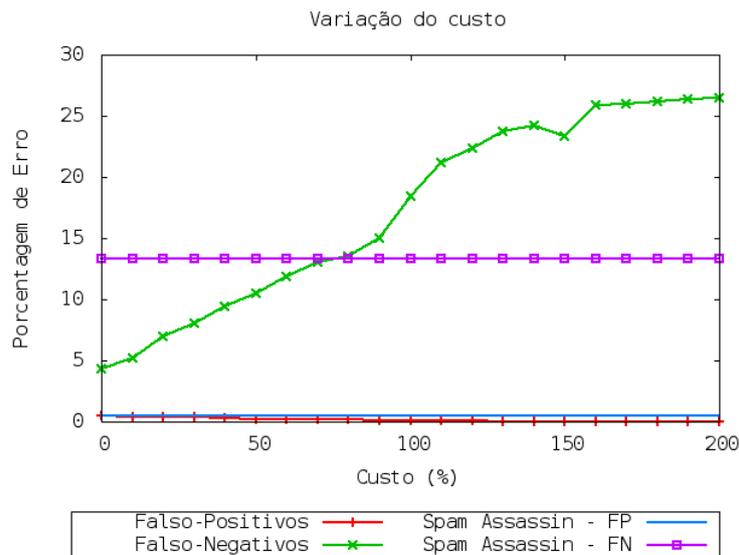


Figura 5. Falsos positivos e falsos negativos x Custo

5.2. Impacto do parâmetro custo na detecção de *spam*

Mostramos na Figura 5 o impacto da variação do custo no índice de falsos positivos e falsos negativos da nossa técnica. Os valores do eixo x representam a diferença entre o custo de se classificar um *ham* como *spam* e o custo de se classificar um *spam* como *ham*. Portanto, se o valor no eixo x é 50%, isso significa que é 50% mais custoso classificar um *ham* como *spam* do que vice-versa. Naturalmente, um aumento no custo gera uma redução do índice de falsos positivos e um aumento no índice de falsos negativos. Com um custo maior do que 70%, nossa técnica passa a classificar *spams* com menos eficácia do que o *Spam Assassin*, embora o número de falsos positivos chegue a 0. É interessante notar que esse compromisso é ajustável na nossa técnica, através do parâmetro custo. Cabe ao usuário da técnica definir o custo de acordo com a sua necessidade.

6. Conclusões e Trabalhos Futuros

Neste trabalho, mostramos que as páginas *Web* apontadas por mensagens de *spam* podem ser utilizadas com sucesso para a classificação dessas mensagens. Nossa proposta consiste em utilizar as páginas como complemento a outras estratégias já utilizadas de classificação de mensagens de *spam*.

Mostramos na seção de trabalhos relacionados que estratégias de filtragem de *spam* convencionais não fazem uso das páginas. A grande maioria dos *spams* contém *URLs* [Pu and Webb 2006], e portanto podem ser filtrados pela nossa técnica. Mostramos também que uma das estratégias mais comuns para a filtragem de *spams*, o uso de listas de bloqueio, esta perdendo sua efetividade [Ramachandran et al. 2006] [Sinha et al. 2008], e portanto é necessário que novas técnicas de filtragem sejam estudadas e utilizadas. Neste trabalho propomos uma técnica que explora um aspecto no qual os *spammers* ainda não escondem a sua identidade. Além disso, as páginas muitas vezes não pertencem aos *spammers*, e portanto são um campo de batalha no qual os *spammers* estão em desvantagem.

Avaliamos o uso de um algoritmo de aprendizado de máquina sob demanda [Velooso et al. 2006] para a classificação das páginas, e propomos uma forma de se agregar a classificação das páginas com a classificação tradicional das mensagens com o filtro *Spam Assassin* [SpamAssassin 2008]. Mostramos que a utilização da nossa técnica melhora a filtragem de *spam* em mais de 10%, sem inserir um número significativo de falsos positivos. Mostramos também que a quantidade de falsos positivos pode ser ajustada com a variação do parâmetro custo.

Acreditamos que o trabalho abre novas possibilidades para o desenvolvimento de estratégias de filtragem de *spam*, introduzindo um aspecto totalmente novo e ainda não explorado na literatura. Em outras palavras, as páginas apontadas pelas mensagens constituem-se em um novo campo de batalha, com o qual hoje os *spammers* não precisam se preocupar. Neste novo campo de batalha, diferentes algoritmos podem ser utilizados para a classificação das páginas, e o resultado da classificação pode ser combinado com outras técnicas. Por fim, servidores de correio eletrônico poderiam utilizar as técnicas descritas em [Guerra et al. 2008] para agrupar as mensagens em campanhas, de forma a diminuir o número de páginas a serem classificadas.

Referências

- Anderson, D. S., Fleizach, C., Savage, S., and Voelker, G. M. (2007). Spamscatter: Characterizing Internet Scam Hosting Infrastructure. pages 135–148.
- Cook, D., Hartnett, J., Manderson, K., and Scanlan, J. (2006). Catching spam before it arrives: domain specific dynamic blacklists. In *ACSW Frontiers '06: Proceedings of the 2006 Australasian workshops on Grid computing and e-research*, pages 193–202, Darlinghurst, Australia, Australia. Australian Computer Society, Inc.
- Fawcett, T. (2003). "in vivo"spam filtering: a challenge problem for kdd. *SIGKDD Explor. Newsl.*, 5:140–148.
- Guenter, B. (2010). Spam archive. <http://untroubled.org/spam/>.
- Guerra, P. H. C., Guedes, D., Jr., W. M., Hoepers, C., and Steding-Jessen, K. (2008). Caracterização de estratégias de disseminação de spams. In *26o Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, Rio de Janeiro, RJ.

- Guerra, P. H. C., Guedes, D., Wagner Meira, J., Hoepers, C., Chaves, M. H. P. C., and Steding-Jessen, K. (2010). Exploring the spam arms race to characterize spam evolution. In *Proceedings of the 7th Collaboration, Electronic messaging, Anti-Abuse and Spam Conference (CEAS)*, Redmond, WA.
- MAAWG (2009). Email Metrics Program: Report #5 – Third and Fourth Quarter 2008. http://www.maawg.org/about/MAAWG_2008-Q3Q4_Metrics_Report.pdf.
- Millettary, J. (2005). Technical trends in phishing attacks. Technical report, CERT Coordination Center, Carnegie Mellon University. http://www.cert.org/archive/pdf/Phishing_trends.pdf.
- Pu, C. and Webb, S. (2006). Observed trends in spam construction techniques: a case study of spam evolution. *Proceedings of the 3rd Conference on Email and Anti-Spam (CEAS)*.
- Ramachandran, A., Dagon, D., and Feamster, N. (2006). Can dns-based blacklists keep up with bots? In *In Proceedings of the 3rd Conference on Email and AntiSpam (CEAS) (Mountain View)*.
- Sinha, S., Bailey, M., and Jahanian, F. (2008). Shades of grey: On the effectiveness of reputation-based blacklists. In *Malicious and Unwanted Software, 2008. MALWARE 2008. 3rd International Conference on*, pages 57–64.
- Sipior, J. C., Ward, B. T., and Bonner, P. G. (2004). Should spam be on the menu? *Commun. ACM*, 47(6):59–63.
- SpamAssassin (2008). <http://spamassassin.apache.org>.
- Upasana and Chakravarty, S. (2010). A survey on text classification techniques for e-mail filtering. In *Machine Learning and Computing (ICMLC), 2010 Second International Conference on*, pages 32–36.
- Veloso, A., Jr., W. M., and Zaki, M. J. (2006). Lazy associative classification. In *ICDM*, pages 645–654. IEEE Computer Society.
- Veloso, A., Jr., W. M., and Zaki, M. J. (2008). Calibrated lazy associative classification. In de Amo, S., editor, *SBB*, pages 135–149. SBC.
- Webb, S. (2006). Introducing the webb spam corpus: Using email spam to identify web spam automatically. In *In Proceedings of the 3rd Conference on Email and AntiSpam (CEAS) (Mountain View)*.