DETECÇÃO DE MENSAGENS NÃO SOLICITADAS UTILIZANDO MINERAÇÃO DE TEXTOS

Marcio Succar Moreira

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Engenharia Civil, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia Civil.

Orientador: Nelson Francisco Favilla Ebecken

Rio de Janeiro
Setembro de 2010

DETECÇÃO DE MENSAGENS NÃO SOLICITADAS UTILIZANDO MINERAÇÃO DE TEXTOS

Marcio Succar Moreira

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA CIVIL.

Examinada por:

_____

Prof. Nelson Francisco Favilla Ebecken, D.Sc.

_____

Prof[a]. Beatriz de Souza Leite Pires de Lima, D.Sc.

_____

Prof[a]. Myrian Christina de Aragão Costa, D.Sc.

_____

Prof. Elton Fernandes, Ph.D.

RIO DE JANEIRO, RJ - BRASIL
SETEMBRO DE 2010

Moreira, Marcio Succar

Detecção de Mensagens não Solicitadas Utilizando Mineração de Textos/ Marcio Succar Moreira. – Rio de Janeiro: UFRJ/COPPE, 2010.

XI, 56p.: Il.; 29,7 cm.

Orientador: Nelson Francisco Favilla Ebecken

Dissertação (mestrado) – UFRJ/ COPPE/ Programa de Engenharia Civil, 2010.

Referências Bibliográficas: p 54-56

1. Mineração de textos. 2. Spams. 3. Mineração de Dados. 4.SQL Server 2008. I Ebecken, Nelson Francisco Favilla II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Civil. III. Título

AGRADECIMENTOS

Terminar o mestrado é o encerramento de uma etapa muito importante da minha vida. Mais que um título acadêmico, a conclusão deste desafio é a vitória da força de vontade, do amor e da amizade frente à adversidade. Mas essa vitória não é só minha. Tenho que agradecer a todas as pessoas que me ajudaram ou me apoiaram para que eu terminasse o mesmo.

- Primeiro tenho que agradecer ao meu anjo da guarda chamado Marcela. Obrigado meu amor por ter me impulsionado em direção a esta conclusão. Obrigado por ter aberto mão de finais de semana, por ter me cobrado quando era necessário e por ter me ajudado de todas as formas possíveis. Obrigado pela paciência e pelo amor inabalável.
- Tenho que agradecer também ao meu antigo chefe Vitor Bellia, ele que foi o estopim, o pontapé inicial dessa caminhada. Obrigado Vitor por ter ensinado que eu deveria ir dormir todos os dias sabendo mais do que eu sabia ao acordar.
- Quero agradecer o meu orientador Nelson Ebecken, por ter me ajudado e me apoiado, e principalmente, por ter acreditado em mim.
- Ao meu amigo do mestrado, Cristian Klen, que me apoiou e me ajudou inúmeras vezes.
- Ao meu amigo Fábio Couto por ser verdadeiramente um amigo. Obrigado por dizer o que os outros não dizem e obrigado pela ajuda.
- Ao meu antigo professor e amigo Sérgio Serra, que me abriu as portas do mestrado, me orientando na iniciação científica e na publicação do meu primeiro pôster.
- E a todas as pessoas que me dão amor e carinho, tão necessários no decorrer de nossas vidas como: meu filho Pedro, à minha mãe, ao meu primo Paulo, e à toda minha "*famíly-in-law*": Maurício, Patrícia, Sara, Fernando, Solange, Sandra, Adilson, Luiz e principalmente à Júlia e ao Davi pelos momentos de amor, alegria e diversão.

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

DETECÇÃO DE MENSAGENS NÃO SOLICITADAS UTILIZANDO MINERAÇÃO DE TEXTOS

Marcio Succar Moreira

Setembro/2010

Orientador: Nelson Francisco Favilla Ebecken

Programa: Engenharia Civil

O recebimento de mensagens não solicitadas (Spams) é um problema conhecido de todos os usuários da Internet e gera um custo anual de milhares de dólares para empresas, governos, provedores e usuários.

Mais importante do que não receber os Spams é garantir que uma mensagem autêntica não será classificada como Spam (falso positivo) e será entregue ao usuário. Ao contrário dos Spams, que podem ser facilmente apagados quando recebidos, as mensagens autênticas não entregues podem gerar prejuízos maiores, pois em alguns casos, são vitais para as partes envolvidas e, quando não são entregues, podem afetar uma relação comercial, um tratado uma ação judicial ou até uma relação afetiva.

O objetivo deste estudo é avaliar a metodologia de mineração de textos na detecção de spams utilizando duas ferramentas comerciais, os dados utilizados na classificação serão exatamente os mesmos que foram recebidos nas contas de e-mails. O único tratamento aplicado às mensagens será a aplicação das técnicas que fazem parte do pré-processamento dos dados na mineração de textos.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfilment of the requirements for the degree of Master of Science (M.Sc.)

DETECTION OF UNSOLICITED INTERNET MESSAGES USING TEXT MINING

Marcio Succar Moreira

September/2010

Advisor: Nelson Francisco Favilla Ebecken

Department: Civil Engineering

The receipt of unsolicited messages (Spam) is a known problem for all Internet users and generates an annual cost of thousands of dollars to companies, governments, providers and users.

Besides that, it's even more important to ensure that an authentic message will not be classified as Spam (false positive) and will be delivered to the user. Unlike Spam, which can easily be erased when received, authentic undelivered messages may generate greater losses because in some cases are vital for the parties involved, and when not delivered they can affect a business relationship, a treaty, a lawsuit or even a loving relationship.

The aim of this study is to evaluate the text mining methodology in spams detecting using two commercial tools, the data used in the classification will be exactly the same that were received in the e-mail accounts. The only treatment applied to the messages will be the application of the data mining pre-processing techniques.

SUMÁRIO

ÍNDICE DE FIGURAS

ÍNDICE DE TABELAS

ÍNDICE DE GRÁFICOS

1. INTRODUÇÃO

Este trabalho vai avaliar o comportamento da metodologia da mineração de textos na classificação de mensagens eletrônicas não solicitadas, os Spams. Por estes causarem anualmente prejuízos de milhões de dólares [1], é necessário que existam ferramentas de classificação eficazes para mitigar os danos causados pelos mesmos.

1.1. CONTEXTO

Com a popularização da Internet e do aumento da utilização da mesma como meio de comunicação, aumentou também o número de pessoas que tentam, de alguma forma, obter lucro financeiro utilizando este meio. Uma das formas mais utilizada para se chegar às pessoas na Internet é o envio de mensagens eletrônicas não solicitadas, os Spams. Apesar de existirem vários tipos diferentes de Spams, as utilizações mais comuns destes são: para obter informações do destinatário através de programas maliciosos ou para anunciar algum produto.

Quase todo mundo já recebeu um Spam de anúncio de remédios, com um pedido para renovar a senha do banco ou com um link um para um site com fotos pessoais. Em todos os casos o emissor deseja, na verdade, obter algum lucro ilícito.

Estes emissores de Spams, popularmente conhecidos como Spammers, aproveitam a ingenuidade e a falta de conhecimento em informática da maioria dos usuários, para de alguma forma obter vantagens sobre os mesmos. A fim de evitar que estas mensagens sejam recebidas o usuário pode instalar uma ferramenta anti-spam que tenta, utilizando diversas técnicas diferentes, classificar as mensagens entre autênticas e Spams.

Tabela 1 - Evolução do número de ocorrências e do valor perdido de 2005 a 2009 nos Estados Unidos [1].

Ano	Reclamações	% de reclamações	US$ milhões	% custo	US$ / reclamação
2009	**336.655**	**145%**	**559.70**	**306%**	**1.662,53**
2008	**275.284**	**119%**	**265.00**	**145%**	**962,64**
2007	**206.884**	**89%**	**239.09**	**131%**	**1.155,67**
2006	**207.492**	**90%**	**198.44**	**108%**	**956,37**
2005	**231.493**	**100%**	**183.12**	**100%**	**791,04**

Na tabela 1 pode ser observado o número de reclamações de crimes da Internet nos Estados Unidos [1] que ocorreram entre os anos de 2005 e 2009, assim como o valor, em dólares, gastos em função destas reclamações. As colunas ***% de reclamações*** e ***% custo*** apresentam a variação destas (custo e reclamações) tendo como base o ano de 2005. Desta forma, no ano de 2007 o número de reclamações foi 89% do número de reclamações de 2005, mas o valor gasto foi 31% maior que o gasto em 2005. Outro ponto interessante apresentando na tabela 1 é o fato do custo por reclamação ter tido um aumento de mais de 100% do ano de 2005 para 2009.

No gráfico 1 pode ser observado que das reclamações existentes 6,2% são causadas por Spams. No relatório não existe o custo, em dólares, causado por cada divisão apresentada no gráfico 1. Isso quer dizer que não tem como saber se o prejuízo de cartão de crédito é maior por reclamação que o custo por spam. Sabendo que o percentual de spams no ano passado foi de 6,2% e que a perda total foi de 559.70 milhões de dólares, podemos inferir que o envio de Spams foi responsável por uma perda financeira de 19,6 milhões de dólares somente nos Estados Unidos. Conforme pode ser observado em na tabela 1, o número de crimes de Internet aumentou 22,3% de 2008 para 2009 e representaram uma perda de 559,7 milhões de dólares no mesmo ano. De acordo com o gráfico 1, 16,6% (55.884) das ocorrências foram por causa de

programas maliciosos, 11.9% (40.061) pela não entrega de falsas propagandas e 6,2% (20.872) foram por causa de Spams.

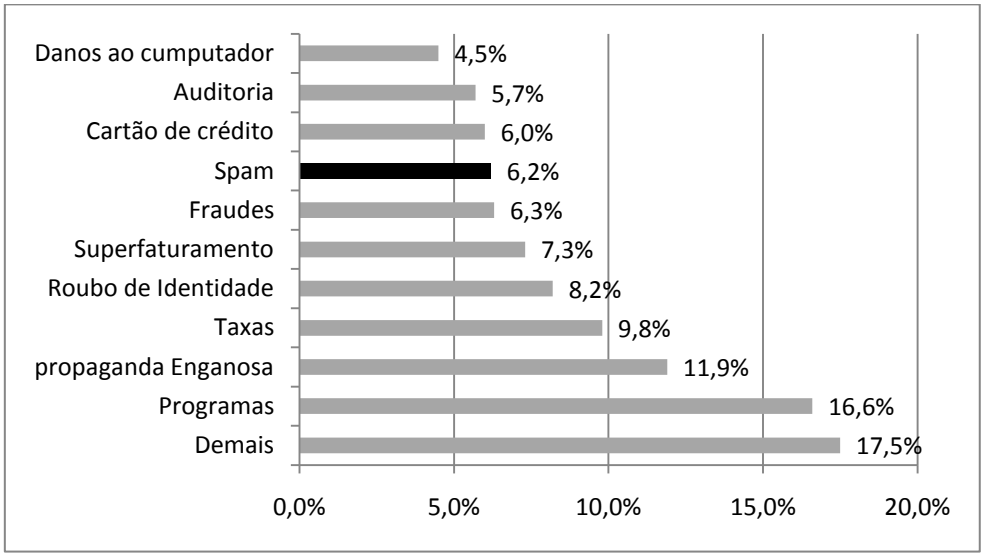

Gráfico 1 - Distribuição dos 10 mais comuns crimes de Internet em 2009 [1]

Atualmente no mercado algumas ferramentas já fazem utilização de algoritmos de mineração de dados para classificar mensagens eletrônicas. O Gmail, por exemplo [2], utiliza diversas ferramentas nesta finalidade. Independente da técnica utilizada na detecção de mensagens não solicitadas, este processo é muito complexo, pois ao contrário dos outros textos, as mensagens não solicitadas são criadas de forma que elas não sejam classificadas como Spams para serem enviadas para a caixa de entrada do destinatário. O fato das mensagens serem criadas em um formato que dificulte a sua classificação cria um conjunto de dados que, ao contrário dos outros documentos, é construído para forçar a classificação errada dos mesmos.

1.2. MOTIVAÇÃO

Como visto anteriormente, o envio de mensagens não solicitadas causou no ano de 2009 um prejuízo de milhões de dólares. Quando um Spam é recebido e o anti-spam o classifica como válido esta mensagem é considerada um falso-negativo. Por outro

lado, quando um e-mail válido é recebido e o sistema anti-spam o classifica como Spam isso é considerado um falso-positivo.

Ao contrário do que pode ser imaginado, um falso-positivo pode ter um impacto maior para uma organização ou uma pessoa que um falso-negativo. Quando uma pessoa recebe um Spam e este vai para a caixa de entrada (falso-negativo) a pessoa pode simplesmente apagar esta mensagem, e continuar o seu trabalho como se nada tivesse acontecido. Por outro lado, quando uma empresa ou pessoa não recebe uma mensagem legítima, ou demora para ler uma mensagem que foi colocada na pasta de Spams, isso pode acarretar uma perda muito maior que o tempo perdido para se apagar uma mensagem não solicitada. O custo de um falso-positivo não lido pode ser o custo do não fechamento de um negócio, o custo da perda de um emprego, ou um custo imensurável como o final de um relacionamento amoroso [3].

Os sistemas de detecção de Spams utilizam várias técnicas para evitar que mensagens eletrônicas sejam erroneamente classificadas, porém mesmo utilizando várias técnicas diferentes, eles continuam classificando as mensagens eletrônicas de forma equivocada [4].

Alguns destes sistemas de classificação de mensagens eletrônicas existentes no mercado utilizam o SQL Server da Microsoft como Sistema Gerenciador de Banco de Dados (SGBD). Apesar do SQL Server possuir algumas ferramentas para mineração de dados, o mesmo não é utilizado com esta finalidade pelos sistemas anti-spam, sendo utilizado apenas como um SGBD normal. Ou seja, os sistemas anti-spam apenas armazenam os e-mails no SGBD. Por causa desta possível sub-utilização do SQL

Server[1] os dados utilizados neste trabalho serão classificados pelo mesmo, a fim de se averiguar a funcionalidade deste como ferramenta de classificação de dados não estruturados.

1.3. ORGANIZAÇÃO DO TRABALHO

O capítulo 1 deste trabalho tem o objetivo de contextualizar o leitor, e facilitar o entendimento do problema aqui abordado. O capítulo 2 aborda o Spam propriamente dito, neste estão descritos os tipos de Spams, a história de seu surgimento, e a história do termo Spam. No capítulo 3 estão descritas algumas técnicas de invasão utilizadas, técnicas de defesa e a motivação dos spammers. No capítulo 4, é explicado como os dados foram obtidos, o pré-processamento que foi aplicado aos mesmos e um breve resumo da construção e importância da BoW (bag-of-words). No capítulo 5 são demonstrados os resultados obtidos e a comparação entre o SQL Server 2008 e o Statistica 8 e no capítulo 6 é apresentada a conclusão do trabalho e as propostas de trabalhos futuros.

Vale ressaltar, que o objetivo principal deste trabalho não é o de descrever minuciosamente o processo de mineração de textos, assim como explicar o funcionamento dos algoritmos utilizados na classificação dos mesmos. O objetivo deste trabalho é demonstrar a aplicação da metodologia de classificação de dados não estruturados, utilizando duas ferramentas conhecidas pelo mercado em um problema atual e de grande importância para grande parte da sociedade.

[1] Será utilizado o SQL Server 2008, que é a ultima versão comercializada durante a realização deste trabalho.

2. MENSAGENS NÃO SOLICITADAS (SPAMS)

Desde o primeiro spam conhecido a motivação principal foi à obtenção de algum tipo de retorno financeiro. Seja promovendo algum tipo de produto ou serviço ou tentando obter informações para causar uma fraude. O fato é que atualmente o Spam é o meio mais barato de se enviar informação para um vasto número de usuários [5].

De acordo com o relatório de segurança da Microsoft [6], aproximadamente 720 bilhões de spams foram bloqueados pelo serviço de detecção on-line de Spams da Microsoft, o FOPE (*Microsoft **F**orefront **O**nline **P**rotection for **E**xchange*), que é um serviço utilizado por empresas do mundo todo sendo a primeira porta de bloqueio de Spams para as empresas que o utilizam.

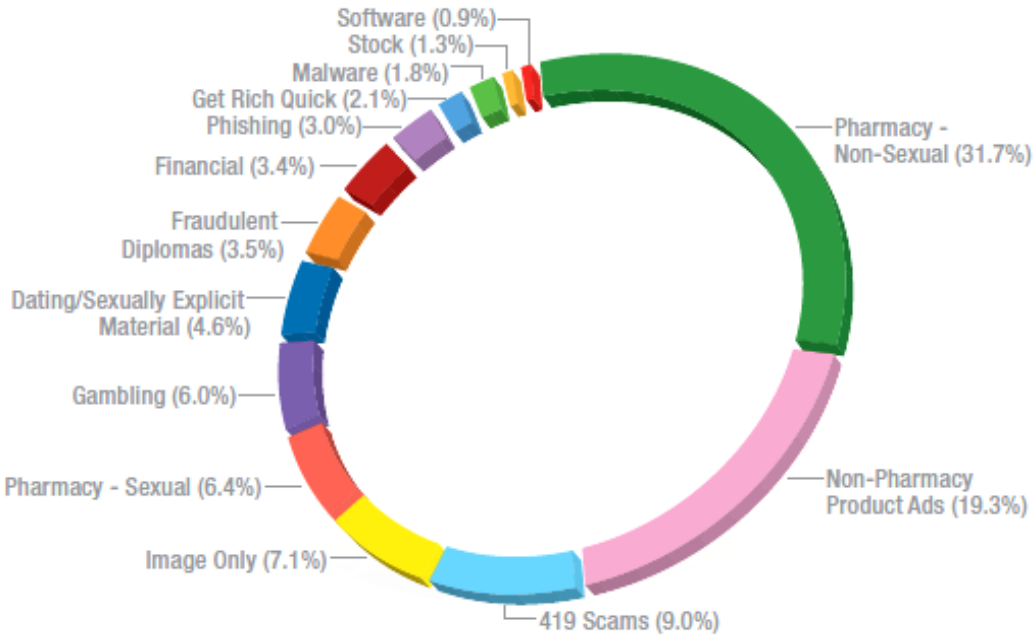

Gráfico 2 - Distribuição dos Spams por tipo de mensagem [6]

Destes 720 bilhões de Spams recebidos, 97,8% foram bloqueadas antes de chegar à empresa, os 2,2% restantes, tiveram que ser analisados por sistemas de detecção de Spam que analisam o assunto e a mensagem em si. Apesar de

percentualmente o valor ser pequeno, 2,2% de 720 bilhões perfazem um total de 15.840.000.000 Spams/ano.

Deve ser observado que este número é apenas dos servidores do FOPE. Estima-se que em 2009 foram enviados 90 trilhões de e-mails, e que destes 81% foram spams, o que daria uma media de 200 bilhões de Spams por dia [7]. Se for levado em consideração que estas mensagens ocupam banda da Internet, espaço de armazenamento dos provedores, poder de processamento, mão de obra para operar estes equipamentos, tempo de leitura de cada Spam etc. será fácil compreender por que o Spam causou um prejuízo de milhares de dólares no ano de 2009 [1], conforme pode ser observado na tabela 1, e por que deve ser combatido o mais próximo de sua origem. Um Spam que passa pelas primeiras barreiras de proteção, é um Spam que ocupará mais recursos de uma forma geral. Segundo [8] anualmente uma empresa gasta aproximadamente 712 dólares por empregado por causa dos Spams.

2.1. A ORIGEM DO SPAM

A origem do Spam é um pouco controversa, existem duas primeiras histórias de "primeiro Spam" no mundo. O primeiro, de acordo com o site Wired, foi enviado por Gary Thuerk. Este foi enviado em 1978 para quase 400 pessoas e era um convite para o lançamento de novos produtos da DEC (Digital Equipment Corporation).

A segunda versão para o primeiro Spam é a mesma que o comitê Gestor de Internet no Brasil [4] trata como a história oficial do primeiro Spam. Esta diz que o primeiro Spam foi enviado em cinco de março de 1994. Este foi enviado por dois advogados que enviaram a mensagem para um grupo da USENET a fim de promover

uma loteria. Este fato foi agravado 27 dias depois quando os mesmos enviaram a mesma mensagem utilizando um sistema de envio automático que enviou esta para diversos grupos de discussão da USENET[2].

2.2. O TERMO SPAM

O termo Spam veio de um *sketch* do grupo de comediantes o Monty Python que passou na televisão em 1970. Neste sketch um casal perguntava quais comidas haviam no restaurante e escutava da funcionária do bar que todos os pratos levavam Spam (Uma marca de presunto[3]). Mesmo que a pessoa pedisse o prato sem Spam, o bar não venderia, pois só venderia comida com Spam [9]. Nesta sátira, da mesma forma que o presunto era indesejado pela cliente do bar, as mensagens também são por aqueles que as recebem. Este sketch pode ser visto em http://www.youtube.com/watch?v=anwy2MPT5RE.

2.3. TIPOS DE SPAMS

Atualmente os Spams são divididos em nove tipos distintos, [4]:

- ✓ Correntes (chain letters)

Os Spams do tipo Correntes normalmente solicitam que o leitor repasse a mensagem para todos os seus amigos e parentes. Aparentemente inofensivas estas mensagens podem congestionar servidores de e-mail.

Exemplos de assuntos de Spams do tipo Corrente segundo:

---

[2] USENET (Unix User Network) é um meio de comunicação criado em 1979 e era utilizada para troca de mensagens, artigos e códigos binários em vários grupos existentes [6]. Seria um análogo aos grupos do Google, Yahoo etc. Na época de sua criação, o USENET utilizava um protocolo próprio UUCP e a conexão era direta entre a máquina do usuário e um dos servidores do grupo.

[3] Uma marca de presunto enlatado que ficou famosa durante a segunda guerra mundial, pois era nutritivo e não precisava de refrigeração.

- O Projeto de Lei 5476/2001 existe e propõe o cancelamento da assinatura do telefone fixo
- Trapaça eletrônica. Historiadeamor.com manda cartão romântico.
- Promoção Brasil Telecom

✓ Boatos (hoaxes) e lendas urbanas

Os Boatos são muito parecidos com as Correntes, pois tentam convencer o leitor a passar para todos os seus conhecidos. A diferença das correntes é o formato dos e-mails. Os Boatos normalmente contas estórias alarmantes e falsas enquanto os do tipo corrente dizem para você passar para x pessoas que conhece para ganhar uma bênção ou prêmio.

Exemplos de Spams do tipo boatos:

- **Difamatórios**: Espalham inverdades pela rede. Denigrem a imagem de empresas ou pessoas dentre outras formas de difamações.
- **Filantrópicos**: Utilizam a desgraça alheia para obter lucro financeiro. Falam que estão captando recursos para as vítimas de enchentes, terremotos etc..

✓ Propagandas

Os Spams que divulgam algum produto são conhecidos como UCE (Unsolicited Comercial e-mail). Existem empresas que fazem uso deste meio apesar de existirem outros meio de se divulgar produtos e serviços. Empresas que utilizam o Spam acabam perdendo credibilidade. Também existem casos de anuncio de produtos que não existem. Caso uma pessoa compre um produto que não existe, a mesma pode procurar a justiça a fim de reaver seu dinheiro.

✓ Ameaças, brincadeiras e difamação

Quando uma mensagem difamadora é enviada para muitas pessoas isso é considerado Spam. No caso de uma difamação ou ameaça a empresa ou pessoa difamada podem procurar a justiça para processar o divulgador da mesma.

- ✓ Pornografia

Essa é a mais antiga modalidade de Spam. A possibilidade de recebimento de material pornográfico por crianças ou de crianças (pedofilia) é um dos motivos para os usuários utilizarem programas que detectem spams automaticamente.

Em caso de pedofilia, o usuário deve notificar a polícia Federal pelo e-mail dcs@dpf.gov.br.

- ✓ Códigos maliciosos

O remetente destes e-mails insere em seu código um programa com código malicioso e/ou um link para um programa na Internet a fim de fazer com que o destinatário execute o mesmo.

Os cinco tipos mais comuns de código são:

- Backdoor – Programa que permite que o emissor da mensagem invada o programa infectado.
- Spyware – Estes são feitos para monitorar a utilização do computador e enviar tudo para um terceiro.
- Keylogger – Estes armazenam tudo que é digitado no computador e enviam os dados para um servidor na Internet ou os armazena em disco. Com isso ele captura senhas,
- Screenlogger – Capturam a tela ou uma área ao redor do mouse no momento do click.

- ✓ Fraudes

São spams que têm a intenção de enganar os usuários a fim de obter vantagens sobre os mesmos. Esses podem ser através de falsos sites de cartão de crédito ou falsos sites bancários.

- ✓ Spit e spim
    - É a modalidade de Spam através de telefones com Internet (Spam Internet Mobile ou Telefone). Eles podem ser enviados pelo ICQ ou Messenger dos dispositivos móveis.
- ✓ Spam via redes de relacionamentos
    - São as mensagens não solicitadas enviadas via sites como o www.orkut.com.br. Estas podem conter links para códigos maliciosos ou simplesmente ter caráter difamatório ou comercial.

Como pode ser visto acima existem Spams, como os de Propaganda, que não representam muito risco aos seus destinatários, pois não vão instalar nenhum código malicioso como um vírus. E existem os perigosos, que podem instalar um código malicioso, ou levar o cliente a um falso site de banco para que o cliente digite a sua senha.

3. COMO "TRABALHAM" OS SPAMMERS

Para alcançar o maior número de pessoas o Spammer pode adquirir uma lista de e-mails de uma empresa que venda endereços de e-mails válidos, ou pode utilizar um aplicativo a favor dos seus objetivos. Abaixo serão apresentadas algumas técnicas utilizadas pelos Spammers.

3.1. VARREDURA DE SITE

Montar um robô que varra vários sites a fim de encontrar endereços de e-mail válidos é muito utilizado pelos Spammers. Esta técnica, conhecida como *harvesting* [10], é composta por duas etapas. Exemplificando que o Sr. X, um Spammer em início de carreira, precisa de 100.000 endereços válidos. Como o mesmo não quer comprar isso, ele fabrica um "Robô" que começará a abrir vários sites e procurará endereços de e-mail válidos nos mesmos afim de cadastrá-los em um banco de dados de endereços de e-mails. Ele vai encontrar, por exemplo, na página de contato do PEC da COPPE da UFRJ[4] os endereços de e-mail academica@coc.ufrj.br e support@coc.ufrj.br, uma vez que o mesmo esteja com esta informação ele pode, e normalmente faz, duas coisas:

1. Mandar um Spam com outro programa, que pode transformar o computador da pessoa em um servidor de Spams;
2. Começar a enviar para o domínio encontrado coc.ufrj.br vários Spams utilizando uma das técnicas a ser explicada neste trabalho, o ataque de dicionário.

[4] http://www.coc.ufrj.br

Uma forma de se burlar este tipo de ataque é a colocação dos endereços de e-mail em figuras (figura 1)[5], a criação de páginas onde a pessoa envie sua mensagem pelas mesmas, ou, por exemplo, a substituição da @ por at. Ao invés de fulano@dominio.org se coloca fulano*at*dominio.org.

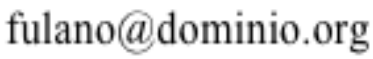

Figura 1 - endereço de e-mail como figura

3.2. IDENTIFICANDO ENDEREÇOS VÁLIDOS

Quando um Spammer começa a varrer um domínio, fazendo o ataque de dicionário em busca de endereços de e-mail válidos, ele basicamente testa se as mensagens foram abertas ou não. Uma forma muito simples de testar isso é incluindo o link de uma imagem, um *web bug*[6], neste e-mail e verificando se a mesma foi aberta. Quando um computador faz uma requisição para um servidor web o servidor pode pegar informações da máquina que requisita a mesma. Se, por exemplo, uma mensagem foi enviada para fulano@dominio.org e este usuário for válido e abrir oSpam, um servidor do Spammer pode confirmar que este endereço de e-mail é válido. Veja abaixo algumas formas de se utilizar o *web bug* [10]:

- Criar a imagem do *web bug* com o nome do endereço de e-mail que quer validar, Exemplo: fulano@dominio.org.png;

[5] Apesar de ser mais seguro, Spammmers mais avançados podem incluir em seu robô um ocr para varrer o conteúdo das imagens.

[6] Link de imagem inserido no e-mail que quando exibido "informa" ao Spammer que o destinatário é válido.

- Hospedar o *Web bug* em um servidor web onde o Spammer tenha acesso às informações de acesso que serão geradas quando o *Web bug* for visualizado;
- Criar uma mensagem de e-mail no formato HTML, que tenha em seu conteúdo a URL completa da imagem correspondente ao *Web bug*, Exemplo: http://www.dominio.example.org/fulano@dominio.org.png;
- Enviando a mensagem criada para o endereço de e-mail a ser validado. Exemplo: fulano@dominio-atacado.org e verificando se o servidor retorna um erro.

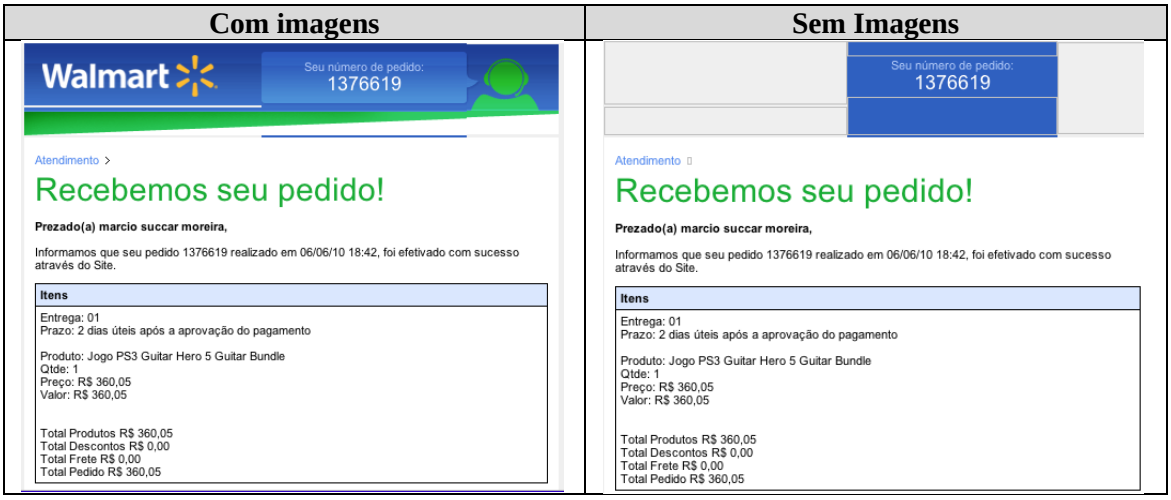

Figura 2 – Exemplo de como o Gmail apresenta um e-mail sem figuras e com figuras.

Assim que o usuário abrir a mensagem, esta vai fazer a requisição para abrir a imagem ou endereço web e o servidor do Spammer registrará que foi feita a requisição para abrir a imagem fulano@dominio.org.png. Depois um programa varrerá o log do servidor Web procurando o nome destas imagens e conseqüentemente os endereços válidos. Para evitar que este tipo de técnica possa surtir algum efeito, o usuário pode configurar seu servidor de e-mail ou seu programa de e-mail para que só receba mensagens em formato txt ou formatado sem figuras. Isso faria com que os links automáticos ou aparecessem apenas como um endereço web ou que as figuras não fossem exibidas, evitando o *Web bug*.

3.3. O ATAQUE DE DICIONÁRIO

Agora que já foi explicada uma das formas como um Spammer identifica um e-mail válido, pode-se perguntar como ele escolhe o nome de usuário a ser testado. Aproveitando o exemplo anterior, onde o endereço de e-mail a ser testado e atacado era fulano@dominio.org, o Spammer testará, digamos, 10.000 nomes por domínio. Para isso, ao invés dele tentar aleatoriamente a parte antes da @ (o id do usuário). Ele utilizará um conjunto de palavras utilizadas no ramo de trabalho do domínio a ser atacado, como se fosse um dicionário específico, e testará cada uma dessas palavras como parte do endereço eletrônico. Por exemplo:

Tabela 2 - exemplo de nomes no dicionário e e-mails a serem testados

Palavra no dicionário	e-mail a ser testado
atendimento	atendimento@dominio.org
compras	compras@dominio.org
Suporte	suporte@dominio.org
Support	support@dominio.org
sac	sac@dominio.org

3.4. CRIANDO O SPAM

Uma vez que o Spammer possua a lista de destinatários, ele precisa criar uma mensagem que desperte o interesse de seus destinatários e que não seja detectada pelos anti-spams. Todo e-mail pode ser enviado como texto ou como HTML. Quando um e-mail é enviado como texto, este não pode ter nenhum tipo de formatação, link para a Internet (pode ter um endereço, mas não um link), figura etc. Um e-mail que vai como texto não tem nenhum apelo visual e não pode esconder “informações” (como web bugs) de quem o lê. Desta forma, os Spams, normalmente, são enviados como HTML, por serem mais atrativos e mais difíceis de serem detectados.

Os e-mails em formato HTML podem ser quase um web site. Existem e-mails promocionais que enviam imagens de produtos e links, como se fossem uma página web. Um e-mail em formato texto normalmente não é bloqueado pelo anti-spam, pois não fornece perigo ao destinatário. Porém um Spam que não tem link e não é atrativo dificilmente vai servir aos interesses do Spammer.

Independente de ser HTML ou não, o e-mail será analisado pelo anti-spam. Uma forma de burlar a análise de texto do e-mail é inserir algumas figuras na mensagem de forma que estas, juntas e alinhadas, formem a mensagem desejada. Como o texto estará em uma imagem à análise de textos não encontrará o mesmo e classificará a mensagem como e-mail válido. Existe uma técnica que insere partes de mensagens autênticas no spam de forma que o programa anti-spam não os classifique corretamente, conhecida como *good word attack* (ataque de palavras "boas"), [11].

3.5. ENVIANDO O SPAM

Quando um anti-spam recebe muitas mensagens classificadas como spam de um mesmo servidor, o programa bloqueia este emissor de spams e, dependendo do programa anti-spam, envia o IP do mesmo para uma lista pública de servidores bloqueados (Blacklist[7]). Uma forma de se driblar este tipo de bloqueio e de distribuir o trabalho do envio é a utilização de servidores de e-mail invadidos para que estes sejam os emissores dos Spams. Isso viabiliza que os Spams passem pela verificação de servidores SMTP e desta forma consigam chegar ao servidor de e-mail do destinatário da mensagem.

---

[7] *Blacklist* - http://en.wikipedia.org/wiki/DNSBL - Lista de servidores bloqueados. O termo pode ser utilizado para pessoas também. É uma lista negra. O *blacklist* de servidores pode ser implementado em DNS ou em provedores que divulgam esta lista para empresas, de forma que estas bloqueiem as mensagens provenientes de servidores bloqueados.

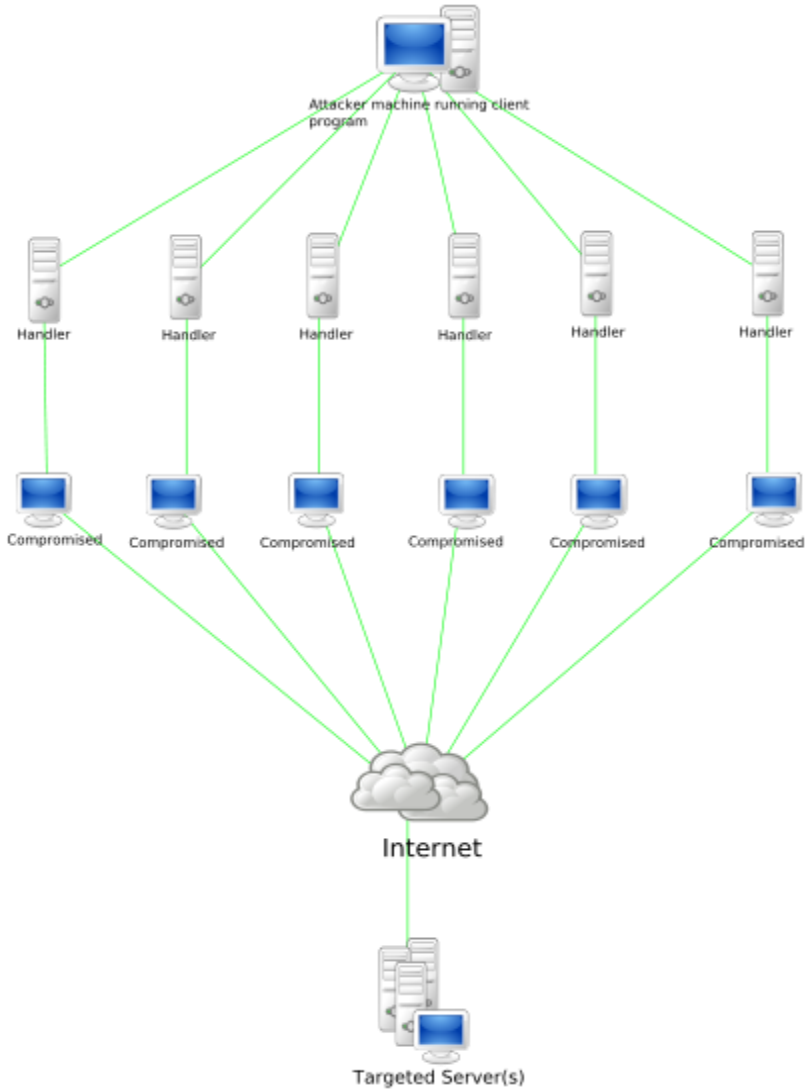

Figura 3 - Exemplo de um esquema de servidores executando o ataque de DOS

Existem várias técnicas de invasão de servidores. A mais comum é o ataque de força bruta (*DOS Denial-of-service*[8]), que derruba um servidor fazendo mais requisições do que este pode atender. Uma vez que a máquina foi invadida, o Spammer instala um aplicativo que torna esta máquina em um escravo. Esta pode ser utilizada para simplesmente enviar spams, ou pode fazer parte do grupo de máquinas que ataca outros servidores, fazendo o uso do ataque de força bruta.

[8] http://en.wikipedia.org/wiki/Denial-of-service_attack -

Na figura 3 pode ser observado um primeiro computador rodando uma aplicação cliente, que controla os computadores invadidos, e utiliza os mesmos para invadir um novo servidor através da solicitação maciça de algum serviço (e-mail, pagina web etc) em funcionamento neste.

Outra forma de se enviar e-mails a partir de outros servidores é a utilização de servidores de e-mail com o "relay aberto".

Na figura 4, pode ser observado o caminho normal de um e-mail sendo enviado de uma empresa pela linha pontilhada, e o e-mail sendo enviado pelo Spammer pela linha tracejada. O Spammer utilizará o servidor válido e conhecido da Internet para enviar suas mensagens. Provavelmente este servidor entrará em uma *blacklist* trazendo problemas para a empresa e seu administrador.

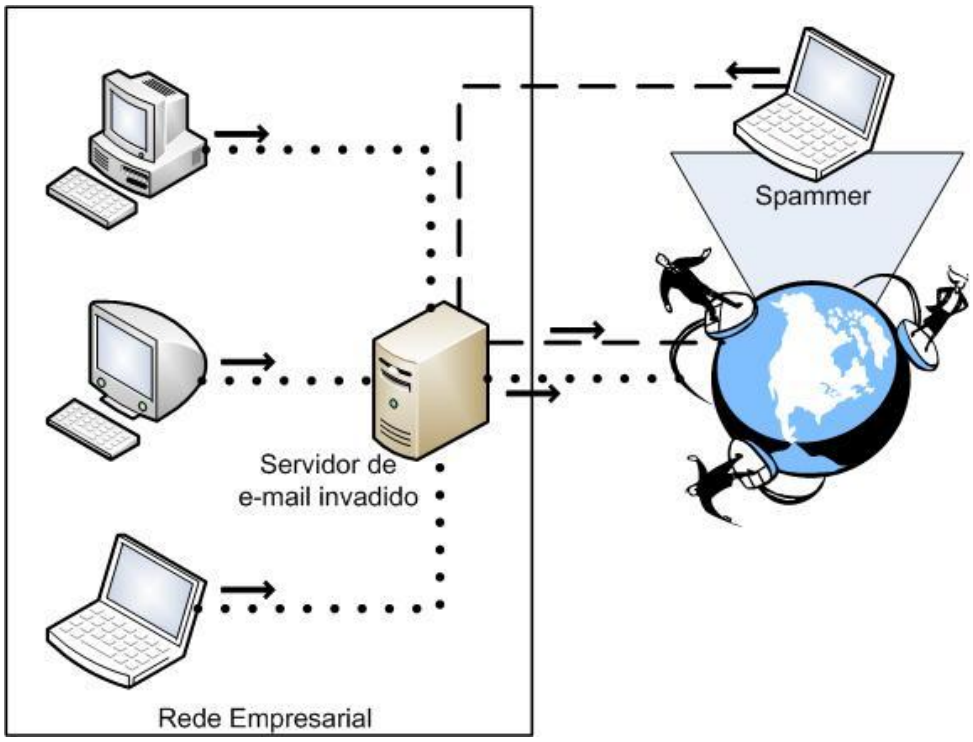

Figura 4 - A rota "normal" de um e-mail em verde, e a roda de um e-mail enviado por um Spammer em vermelho.

Utilizando servidores invadidos o Spammer não pode ser localizado e não necessita manter a infraestrutura necessária para exercer suas atividades. Ele não precisa de um link veloz, pagar a energia necessária, refrigerar as máquinas etc. Para ele é como estar trabalhando em uma *cloud*[9] sem custos. O Spammer pode estar em um café em São Paulo ou numa praia do Rio de Janeiro[10] enquanto servidores invadidos e espalhados pelo mundo enviam e-mails para uma empresa na Índia.

3.6. FORMAS DE PROTEÇÃO

Por quais e quantos "filtros" um Spam passa até chegar à caixa de e-mail do destinatário? Como usuário, não é possível saber esse tipo de informação, até por que isso depende do provedor, sistema operacional dos servidores, como os servidores foram configurados, sistemas instalados etc. O que pode ser afirmado é que atualmente todo o Spam passa em algum filtro. Abaixo serão listados alguns filtros e como estes funcionam.

A jornada de uma mensagem começa no servidor de e-mail que acaba de receber um Spam. Se o destinatário dos e-mails tiver uma ferramenta anti-spam esta verificará se o IP do emissor está bloqueado em alguma blacklist. Não estando, ele verificará se o IP do servidor emissor responde pelo domínio na Internet que está sendo utilizado na mensagem. Por exemplo, no DNS do endereço www.gmail.com existe o

---

[9] C*loud computing* - O conceito de computação em nuvem refere-se à utilização da memória e das capacidades de armazenamento e cálculo de computadores e servidores compartilhados e interligados por meio da Internet, seguindo o princípio da computação em grade. - http://pt.wikipedia.org/wiki/Computa%C3%A7%C3%A3o_em_nuvem

[10] O Governo do Estado do Rio de Janeiro criou um programa que visa, dentre outras coisas, disponibilizar Internet gratuita para todo o Rio de Janeiro. O projeto Rio Estado Digital, é um exemplo clássico do tipo de conexão que pode ser utilizada por um Spammer a fim de executar suas tarefas sem ser encontrado. http://www.governo.rj.gov.br/noticias.asp?N=59203

nome do servidor MX[11] desse domínio. Nesse caso o servidor mail.google.com é o servidor MX do gmail, o servidor destinatário verificará se o e-mail recebido foi enviado pelo IP do servidor MX do gmail.

Ou seja, se o Spamer enviou um Spam com o endereço suporte@banco.com.br e o servidor que está enviando a mensagem não for o responsável por enviar e-mail do banco, o servidor do destinatário recusará a mensagem.

Uma vez que tenha sido confirmado que o servidor que está enviando a mensagem é o servidor correto, o e-mail entra na fila do anti-spam para ser analisado. Uma vez que o mesmo seja analisado pelo sistema anti-spam, ele é liberado ou não para a caixa de correio eletrônico do destinatário.

[11] MX – Mail Exchange – é o apelido dos servidores de recebimento de mensagens no DNS. http://www.mxtoolbox.com e http://en.wikipedia.org/wiki/MX_record

4. OS DADOS

Nesta parte do trabalho será relatado todo o processo de obtenção dos dados (mensagens eletrônicas, e-mails) assim como o tratamento que os mesmos sofreram antes de serem analisados.

4.1. OBTENÇÃO DOS E-MAILS

Os e-mails utilizados neste trabalho foram obtidos de diversas fontes. Várias pessoas contribuíram com seus e-mails, Spams ou não. Um aplicativo foi desenvolvido, o *Spam Miner*, para receber as mensagens via protocolo POP3[12] e armazená-las em uma base de dados do SQL Server 2008.

O *Spam Miner* foi desenvolvido no Visual Studio 2008 da Microsoft na linguagem de programação C# .NET. O mesmo faz uma conexão, utilizando POP3, com os provedores cadastrados na base de dados e realiza posteriormente a importação das mensagens para outra tabela na mesma base de dados do SQL Server 2008. Nas figuras 5 e 6 são apresentados o DER[13] da base de dados e a interface do *Spam Miner*.

Para este trabalho foram obtidas 71.857 mensagens de 12 contas diferentes, sendo 65.141 (90.65%) Spams e 6.716 (9.35%) e-mails autênticos. Para que o classificador fosse treinado e como a origem das mensagens era conhecida, estas eram importadas e classificadas como Spam (1) ou Autênticas (0) através de um campo binário na base de dados do sistema.

---

[12] Protocolo utilizado para descarregar sequencialmente mensagens eletrônicas para um programa no computador do usuário. Após estas serem descarregadas, o usuário pode ler as mesmas sem precisar estar conectado à Internet [20].

[13] Diagrama que descreve o modelo de dados. É utilizado para representar o modelo conceitual de um banco de dados [21].

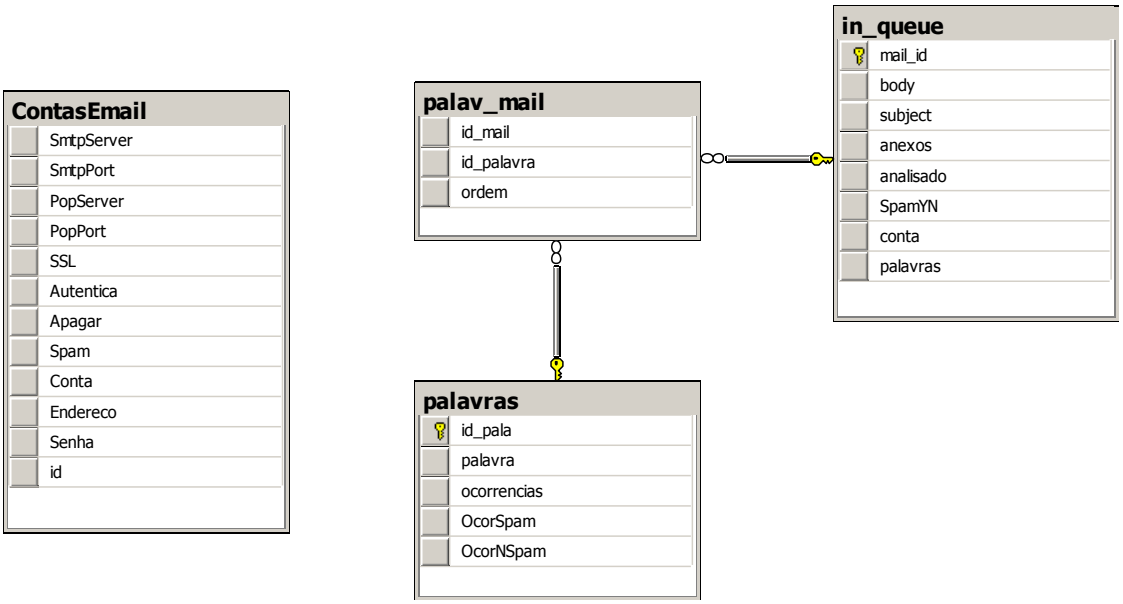

Figura 5 - Diagrama Entidade Relacionamento da base de dados utilizada pelo Sistema Spam Miner

Durante o recebimento, as mensagens foram armazenadas em sua forma original na tabela *in_queue* da base de dados do sistema. Após três meses apenas executando a importação, os dados começaram a ser analisados a fim de se verificar se os mesmos poderiam ser utilizados no experimento.

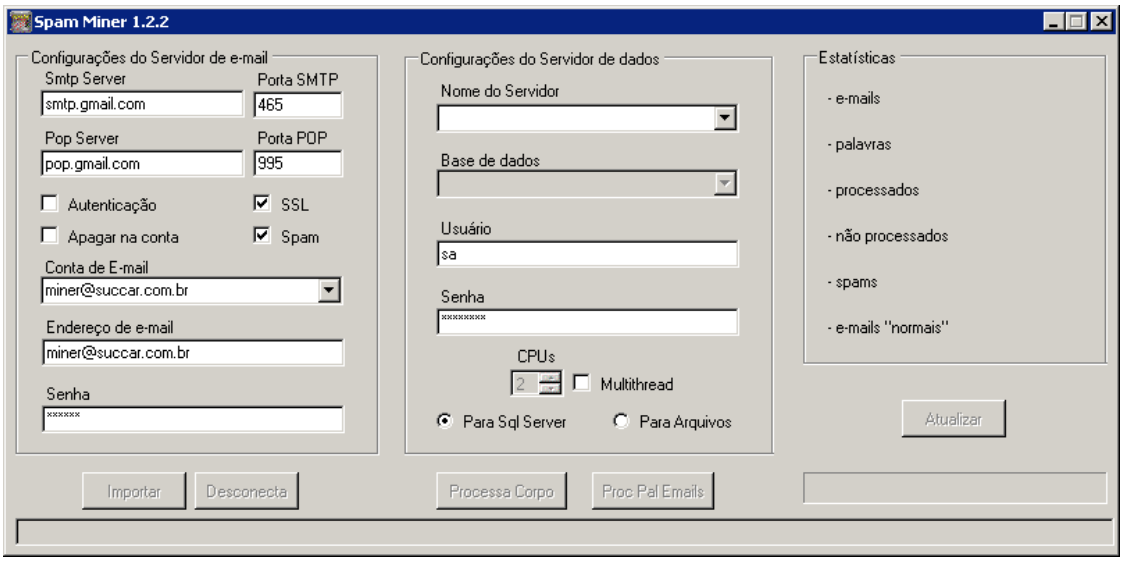

Figura 6 - Imagem da Interface do Spam Miner

Todas as mensagens eletrônicas utilizadas neste trabalho foram obtidas de contas de e-mail de pessoas que se dispuseram a fornecer as mesmas. Para treinar o

classificar, o anti-spam do gmail foi utilizado como padrão de classificação. Para tal, todas as mensagens seguiam o caminho demonstrado na figura 7.

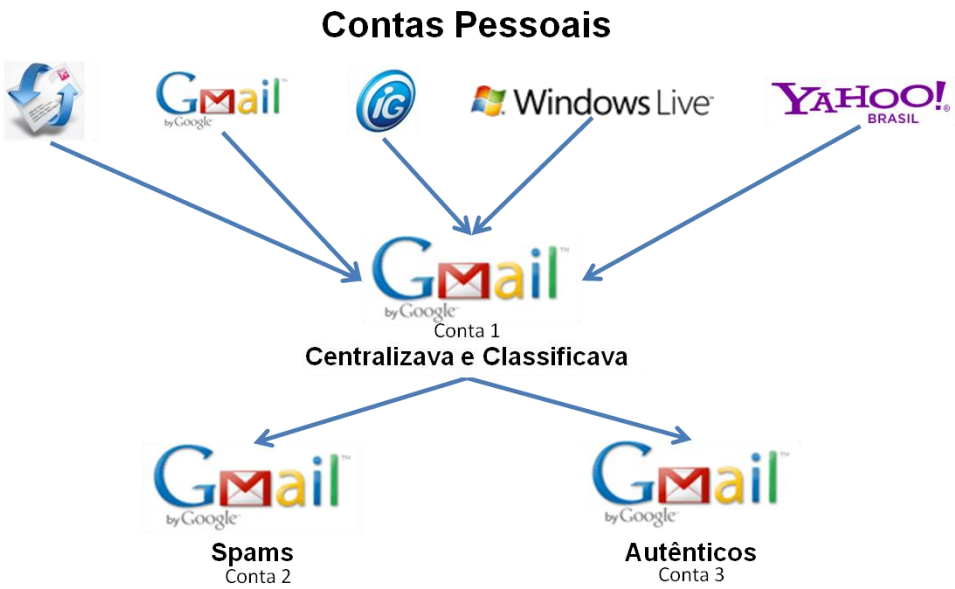

Figura 7 - Processo de importação e classificação das mensagens recebidas

Todos as mensagens destas pessoas eram recebidas pela conta 1 do Gmail. Uma vez nesta conta, as mensagens eram recebidas pela conta 3 do Gmail se fossem autênticas ou eram recebidas pela conta 2 do Gmail se fossem Spams. Nesse processo os Spams eram retirados da caixa de Spam e colocados na caixa de mensagens para que fossem importados pela outra conta ou pelo Spam Miner.

Após todo esse processo, as mensagens autênticas eram importadas diretamente para a base do SQL Server 2008 e marcadas como 0 no campo binário, e as da conta 2 também eram importadas para a mesma base do SQL Server sendo marcadas como 1 no campo binário.

As contas 2 e 3 da figura 7 foram mantidas até o final do estudo para garantir a possibilidade de re-importação pelo SQL Server 2008 caso ocorresse algum problema com as mensagens no banco de dados.

Tabela 3 – amostra dos dados guardados pelo importador de e-mails na base do SQL Server 2008

ID	Cabeçalho	Corpo	Anexos	Spam	Conta	Palav
1	Coleção Paulo	Para sair desta lista, por	0	1	miner@succar.com.br	35
2	Every woman can	facebook Hi, 3dfabio@newave	0	1	miner@succar.com.br	82
3	If you feel that your	[image: Click Here!] *About this	0	1	miner@succar.com.br	226
4	The prices of these	You won't believe your eyes	0	1	miner@succar.com.br	57
5	If watering your	facebook Hi, magroove@altstad	2	1	miner@succar.com.br	82
6	Enlarging your	Pharmacist's Letter: July 2009;	0	1	miner@succar.com.br	143
7	With us your	marcio@succar.com.br	1	1	miner@succar.com.br	532
8	Receive a	GET YOUR DIPLOMA	0	1	miner@succar.com.br	92
9	Need a diploma?	GET YOUR DIPLOMA	1	1	miner@succar.com.br	92
10	Be the macho-per	facebook Hi, ltang@pigmentia.c	0	1	miner@succar.com.br	117

Além de marcar a mensagem como Spam ou Não, o sistema contava o número de palavras por mensagem, o número de anexos, o corpo, o assunto e guardava o endereço da conta da qual as mensagens foram importadas. A tabela 3 mostra uma consulta com 10 mensagens e seus respectivos os campos da tabela com os e-mails.

4.2. PRÉ-PROCESSAMENTO DOS DADOS

No processo de mineração de textos é necessário que sejam executadas algumas tarefas antes de realizar a mineração propriamente dita. Em um sistema de mineração que funciona em tempo real, essas tarefas são executadas em memória. Para cada nova mensagem ou documento recebido pelo sistema, este executará as etapas de pré-processamento, classificação e "treino" do classificador. Como neste estudo as mensagens estão armazenadas em banco de dados, cada uma das etapas da mineração de textos foi executada em todo o conjunto das mensagens antes da próxima etapa.

O esquema abaixo é usualmente empregado e servirá para exemplificar as etapas citadas as quais as mensagens foram submetidas antes de serem classificadas.

4.2.1. FILTRO

- Foram removidos todos os caracteres não alfabéticos;
- Remoção das mensagens escritas com caracteres de origem Asiática, árabe etc;
- Remoção dos cabeçalhos das mensagens[14] (campos De: Para: cc: etc);
- Remoção de: espaços duplos, tabulações, linhas vazias, acentos;
- Remoção das mensagens em branco[15].

4.2.2. PADRONIZAÇÃO

- Todas as mensagens tiveram seus caracteres convertidos para minúsculo (caixa baixa). Independente de ser início de parágrafo, frase ou ser um símbolo.

4.2.3. ATOMIZAÇÃO

- No *Spam Miner* todas as palavras foram separadas e colocadas em uma tabela de palavras na base de dados do sistema. Nesta etapa foi calculada a freqüência de cada palavra separadamente nos dos conjuntos, Spams e não Spams.

4.2.4. ELIMINAÇÃO DAS *STOPWORDS*[16]

Neste trabalho existem mensagens em inglês e português. Por causa disso foi criado um conjunto de stopwords misto. Foram removidas as palavras sem valor semântico como: artigos, preposições, pronomes etc.

4.2.5. *STEMMING* OU REDUÇÃO DO RADICAL

- Redução de todas as palavras ao seu radical.

[14] O algoritmo utilizado para obter as mensagens colocava o cabeçalho das mesmas no topo do corpo. Isso fazia com que os endereços dos e-mails fossem utilizados pelo algoritmo pra classificar estas.

[15] Após as etapas anteriores algumas mensagens ficaram sem assunto e corpo e foram excluídas.

[16] Palavras sem valor semântico.

4.2.6. BAG-OF-WORDS, BOW

Como explicado interiormente o pré processamento tem a finalidade de tornar o texto capaz de ser analisado por um dos algoritmos utilizados na mineração de textos. Uma das etapas que ocorrem após o pré processamento é a construção da *bag-of-words* (BoW). A BoW é uma coleção desordenada de palavras que mantém o número de ocorrências dessas palavras no conjunto de documentos. Por exemplo, se uma BoW for montado com os dois assuntos das mensagens eletrônicas abaixo, será apresentado o seguinte resultado:

Assunto mensagem 1: Compre Viagra sem sair de casa

Assunto mensagem 2: Compre sem pressa a sua casa

Após o pré-processamento os assuntos ficariam:

Assunto mensagem 1: compr viagra sai cas

Assunto mensagem 2: compr cas

Tabela 4 - exemplo de como é contruída uma BoW

compr	viagra	sai	cas
1	1	1	1
1	0	0	1
2	**1**	**1**	**2**

Como pode ser observado, a pequena BoW da tabela 4 tem a capacidade de mostrar o número de ocorrências de um termo na coleção de documentos. A mesma pode ser utilizada na classificação das mensagens em função de algumas distribuições de probabilidade de uma palavra em mais de uma BoW. Na classificação de mensagens eletrônicas a fim de determinar a existência de Spams, existirão duas BoW. Uma com a distribuição das palavras das mensagens autênticas, e outra com a distribuição das palavras dos Spams. Desta forma podemos definir que uma BoW é uma matriz $X \in$

R^{nxm} de documentos por termos. Onde os elementos X^{ij} representam a freqüência com que um termo (*j*) aparece no documento (*i*).

$$x = \begin{pmatrix} x_{11} & \cdots & x_{1m} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{nm} \end{pmatrix}$$

Figura 8 - representação matemática de uma BoW, onde o termo x é o numero de vezes que o termo m aparece no documento j.

Outra forma mais utilizada de se representar a freqüência dos termos é fazendo a ponderação destes levando-se em conta a importância destes na coleção de documentos, no documento sendo analisado pelo valor normalizado desta freqüência. Desta forma cada termo da BoW pode ser representado pela a função de ponderação $\pi(i,j) = \pi_L(i,j) \times \pi_G(j) \times \pi_N(i,j)$ onde: $\pi(i,j)$

- $\pi_L(i,j)$ é o fator de ponderação local;
- $\pi_G(j)$ é o fator de ponderação global;
- $\pi_N(i,j)$é o fator de normalização do vetor.

Enquanto o fator de ponderação local favorece os termos mais freqüentes no documento, o fator de ponderação global penaliza os termos muito freqüentes em todos os documentos uma vez que estes não ajudam a classificar um documento por estarem muito presentes em toda a coleção.

Se forem analisadas as palavras *Viagra*, *compre* e *watch* veremos que as mesmas serão muito mais freqüentes na BoW dos Spams, do que na BoW das mensagens solicitadas. Na classificação Bayesiana, a BoW pode ser utilizada para determinar a probabilidade de uma palavra estar em uma BoW ou em outra, e assim, classificar o conjunto de termos analisado como sendo de um Spam ou de uma mensagem verdadeira.

Umas das funções embutidas do Spam Miner é a contagem de termos em spams e e-mails desejados e suas freqüências. Apesar de bem custoso esse processamento possibilitava que, uma vez que todas as mensagens tenham sido processadas e armazenadas na base de dados, várias técnicas de classificação pudessem ser testadas sem haver a necessidade de refazer grande parte do processamento.

O processamento de todos os e-mails coletados dura aproximadamente 36 horas no Spam Miner. Este tempo se ao fato de serem mais de 71.800 mensagens. O que daria aproximadamente 2.000 mensagens por horas ou aproximadamente 33 mensagens por minuto.

Tabela 5 - resultado de uma consulta na tabela que armazena a frequência das palavras em cada grupo.

Id_pala	Palavra	Ocorrencias	OcorSpam	OcorNSpam
503052	sair	234	90	144
503054	lista	1337	280	1057
503055	favor	3654	775	2856
503056	clique	2693	999	1694

Na tabela 5 pode ser observada uma consulta à tabela que mantém a frequência das palavras em todos os documentos. Nesta existe um identificador numérico para a palavra (id_pala), existe o número total de ocorrências (Ocorrencias), o número de ocorrências em Spams (OcorSpam) e o número de ocorrências em mensagem autênticas (OcorNSpam). Com isso é possível montar duas BoW uma de Spams e outra de mensagens autênticas.

Como a base de dados utilizada estava desbalanceada, algumas mensagens marcadas como não solicitadas (SPAMs) foram apagadas. É sabido que no treinamento de um algoritmo de classificação, [12] a utilização de uma base de dados desbalanceada faz com que os dados tendam a ser erroneamente classificados como pertencentes da

maior classe. Neste caso se o algoritmo for treinado utilizando a base completa, que é composta de 9.35% de mensagens solicitadas e 90.65% de SPAMs, haverá um número muito grande de falsos positivos. Neste caso, um falso positivo é uma mensagem autêntica que foi classificada como uma mensagem não solicitada.

Para fazer a análise de mensagens em formato HTML, muitos cuidados seriam necessários além de ser necessário o desenvolvimento de um algoritmo que analisasse apenas a mensagem propriamente dita. Mensagens em HTML podem ter vários dos campos de um HTML convencional e o algoritmo teria que analisar, por exemplo, apenas o *body* da mensagem.

Apesar de ter o mesmo nome, o body de uma mensagem de e-mail não tem relação com o body de um HTML. O primeiro representa o corpo da mensagem de e-mail e o segundo representa a parte do HTML que será exibida ao usuário. Um HTML pode conter campos de formatação de texto, scripts dentre outros campos.

Em uma mensagem autêntica, o campo body contém o texto e figuras da mensagem propriamente dita. Desta forma, um anti-spam que analisa HTML precisaria apenas analisar o conteúdo do body do HTML e classificar a mensagem como Spam ou autêntica. Porém quando um spam é construído, o mesmo é feito para ser classificado como uma mensagem autêntica, desta forma o conteúdo do HTML não é colocado no campo correto fazendo com o classificador não detecte o seu verdadeiro conteúdo.

Desta forma foi necessário avaliar se o custo, de se construir um sistema capaz de detectar em que parte do HTML estaria o verdadeiro texto do e-mail para só após detectar a verdadeira mensagem, era menor que simplesmente eliminar as mensagens em HTML e trabalhar com mensagens em texto puro. Além do problema do HTML havia o problema das classes estarem muito desbalanceadas.

Foi verificado que haviam 28.167 mensagens em HTML e que essas eram distribuídas de forma que, se estas fossem eliminadas, o desbalanceamento entre as classes seria suavizado. Desta forma, para suavizar o problema do desbalanceamento e para eliminar a necessidade de ter que tratar as mensagens em HTML, todas estas mensagens foram apagadas da base de dados. A tabela 6 mostra a distribuição do número de Spams e de mensagens autênticas antes e após a exclusão das mensagens em HTML. Apesar desta ação não ter solucionado o problema do desbalanceamento, ela suavizou o mesmo e retirou 28.167 e-mails em HTML da amostra.

Tabela 6 - Distribuição mais equilibrada das mensagens após a exclusão das mensagens em HTML.

Base	**Não Solicitadas**		**Solicitadas**	
Antes	65.141	90.65%	6.716	9.35%
Depois	38.227	87.50%	5.463	12.50%

4.2.7. ELIMINAÇÃO DE TERMOS MUITO FREQÜENTES.

Após todo o processamento anteriormente descrito, foram retirados os termos que ocorrem em mais de 95% de todos os documentos. Ou seja, os termos presentes em 95% das mensagens autênticas e em 95% dos Spams foram eliminados, pois os mesmos poderiam ter o mesmo efeito de uma StopWord na coleção. Como o número de Spams representa 87,50% de todas as mensagens, uma palavra que estivesse em todos os Spams só precisaria estar presente em 40% das mensagens autênticas. Para evitar este tipo de desvio foram eliminadas as palavras que estivessem em 95% dos Spams e em 95% das mensagens autênticas. Ou seja, os termos que estivessem ao mesmo tempo em pelo menos 36.315 Spams e em 5.190 mensagens autênticas. A eliminação de palavras pouco freqüentes não foi realizada, pois palavra poucos freqüentes podem identificar um grupo distinto dentro de uma coleção.

Deve-se ter em mente que em um sistema anti-spam, uma cópia da mensagem é submetida a todo este processo, e a classificação resultante seria aplicada à mensagem propriamente dita e não à todo o conjunto de dados como neste trabalho. A função de classificar mensagens eletrônicas não desejadas é uma tarefa muito complexa, pois além de conter toda a complexidade da classificação de informação não estruturada, ela é a análise de uma informação, que em sua essência é feita para não ser classificada [13]. Outro ponto é que as mensagens podem ser escritas em vários idiomas, o que forçaria o sistema a ser um sistema preparado para esta situação. Na tabela 7 pode ser verificado um exemplo das etapas do pré-processamento sedo aplicadas a um exemplo hipotético.

Tabela 7 - Exemplo de resultados das etapas do pré-processamento

Etapa	Exemplo
Texto Original	A customização dos carros de corrida do Estado.
Filtro	A customizacao dos carros de corrida do Estado
Padronização	a customizacao dos carros de corrida do estado
Atomização	Sem efeito no exemplo
StopWords	customizacao carros corrida estado
Stemming	customiz carr corr estad

Durante a etapa de coleta de dados foram obtidas 71.857 mensagens eletrônicas, destas 65.141 (90.65%) eram Spams, e 6.716 (9.35%) eram mensagens autênticas. Após todas as etapas do pré-processamento o número de mensagens que foram utilizadas no estudo ficou em 43.690 sendo 38.227 (87,50%) Spams e 5.463 (12,50%) mensagens autênticas. Desta forma, nas etapas que se seguem serão analisadas 43.690 mensagens. Se em algum momento, como no gráfico 3, a análise tiver sido realizada em cima de todas as mensagens obtidas (71.857), isso será explicitamente comentado.

5. EXPERIMENTOS

Os experimentos estão divididos em três partes. A primeira apresenta algumas estatísticas básicas sobre os dados. Estas foram baseadas nas informações processadas pelo Spam Miner. A segunda parte mostra os resultados obtidos com o SQL Server 2008 e a terceira mostra, para compararmos, os resultados do Statsoft Statistica 8.

Vale ressaltar que o número de mensagens utilizadas nos experimentos foi de 43.690, sendo que destas eram exatamente iguais e alguns algoritmos, após classificar a primeira ocorrência destas, não reclassificava uma mensagem idêntica ignorando a mesma. O número de mensagens distintas é 10.622 distribuídas em 9.298 Spams e 1.324 autênticas. Esse foi o número de mensagens classificadas por todos os algoritmos com exceção do SVM, que classificou todas as 43.690 mensagens.

5.1. ESTATÍSTICAS BÁSICAS

Como mencionado anteriormente, durante o pré-processamento das mensagens o Spam Miner realizou a contagem de cada termo em todas as mensagens classificadas. Em função do exposto uma análise estatística básica pôde ser realizada a fim de demonstrar algumas características de cada grupo das mensagens eletrônicas.

O gráfico 3 mostra a distribuição das palavras nos e-mails (Spams ou não). Apesar de esta ser uma estatística baseada nos dados coletados (71.857 mensagens) neste experimento, ela deve expressar algo bem próximo da realidade, se levarmos em consideração a Lei dos Grandes Números [14], ao menos no que tange os Spams, uma vez que esta é composta por um número significativo de mensagens. O número de palavras nas mensagens não solicitadas pode ser utilizado como um dos parâmetros que definem se uma mensagem é ou não Spam.

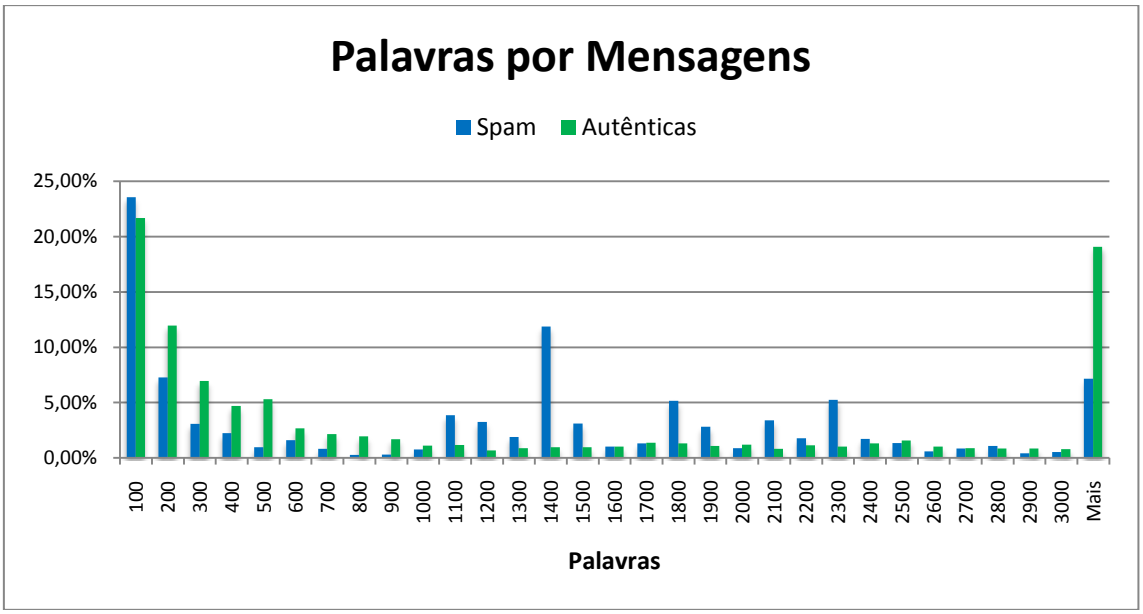

Gráfico 3 - Histograma com a distribuição das palavras por Spams e e-mails autênticos. Até 3.000 palavras

Se observarmos o gráfico 3, 23% dos Spams têm entre 100 e 200 palavras. Também pode ser observado que entre 1.200 e 2.400 palavras estão compreendidos 43% dos Spams. Esta simples análise mostra que se um provedor focar seus esforços de análise nos e-mails com até 200 palavras e entre 1300 e 1400 palavras, ele tem chance de filtrar grande parte dos spams sem muito esforço computacional. Este fenômeno dos Spams serem mais expressivos entre 1.200 e 2.400 palavras se deve ao fato de muitos spams serem enviados em formato HTML para, desta forma, conseguir mapear as características da rede do destinatário da mensagem e/ou colocar figuras, links etc.

No Gráfico 4 foi analisada a distribuição dos e-mails com até 100 palavras a fim de se verificar as características destes mais detalhadamente. Como pode ser observado, e-mails com menos de 10 palavras tem mais de 25% de chance de ser autêntico, assim como e-mails com 50 palavras tem praticamente a mesma chance de ser spam. Todas essas análises podem ser utilizadas a fim de pontuar os e-mails como prováveis spams ou não.

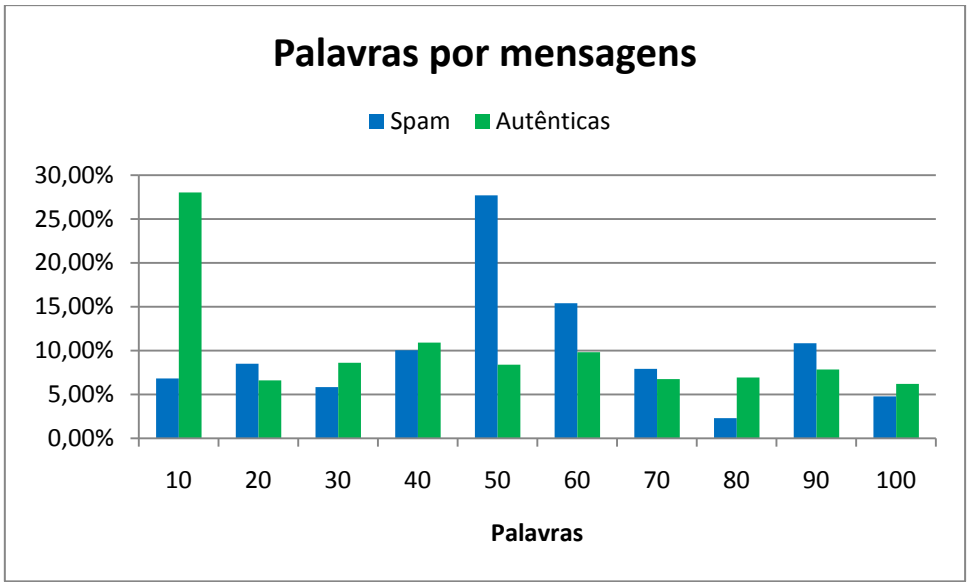

Gráfico 4 - Distribuição de palavras por e-mails com até 100 palavras

Na tabela 8 é apresentada a distribuição de todos os e-mails utilizados neste trabalho. Esta mostra o número de mensagens em função do numero de palavras e os percentuais utilizados no gráfico 3.

Tabela 8 - Percentual de mensagens, por centena de palavras, no conjunto de dados utilizados

Palavras	SPAMS		AUTENTICOS		Palavras	SPAMS		AUTENTICOS	
	Mensagens	%	Mensagens	%		Mensagens	%	Mensagens	%
100	15346	24%	1456	22%	**1700**	854	1%	92	1%
200	4738	7%	803	12%	**1800**	3353	1%	87	1%
300	2000	3%	466	7%	**1900**	1835	1%	72	1%
400	1452	2%	315	5%	**2000**	564	1%	80	1%
500	629	1%	357	5%	**2100**	2205	1%	54	1%
600	1044	2%	179	3%	**2200**	1160	1%	76	1%
700	527	1%	145	2%	**2300**	3422	1%	69	1%
800	170	0%	130	2%	**2400**	1122	1%	87	1%
900	193	0%	114	2%	**2500**	868	2%	105	2%
1000	489	1%	74	1%	**2600**	373	1%	69	1%
1100	2509	4%	78	1%	**2700**	543	1%	59	1%
1200	2122	3%	46	1%	**2800**	698	1%	56	1%
1300	1232	2%	58	1%	**2900**	267	1%	56	1%
1400	7734	12%	65	1%	**3000**	342	1%	52	1%
1500	2027	3%	65	1%	**Mais**	4654	19%	1282	19%
1600	670	1%	68	1%	**----**	----	----	----	---

Na classificação de mensagens eletrônicas podem ser utilizadas várias técnicas para pontuar uma mensagem como sendo autêntica ou não. A utilização de diferentes técnicas pode diminuir a ocorrência de um falso-positivo, e principalmente, diminuir a ocorrência de um falso-negativo.

5.2. MINERAÇÃO DE TEXTOS COM O SQL SERVER 2008

5.2.1. CONSTRUÇÃO DOS MODELOS

O SQL Server 2008 possui muitos recursos para a mineração de dados estruturados. Por ser um gerenciador de banco de dados, o mesmo permite que sejam utilizados os dados armazenados em suas bases e ele possui algumas ferramentas de conversão de dados além de outros recursos, que podem ser utilizados antes dos dados serem carregados pelo algoritmo de mineração de dados.

O SQL Server 2008 possui uma linguagem própria para se trabalhar com data mining, o *Data Mining Extensions* (DMX). Esta linguagem permite que seja criada a estrutura que define o modelo de mineração de dados, assim como é utilizada para: apagar, criar, modificar e executar tarefas e modelos de mineração de dados. Ao mesmo tempo que o SQL Server 2008 disponibiliza esta linguagem, que pode ser utilizada em um aplicativo, ele disponibiliza as mesmas opções em sua interface facilitando o trabalho do analista.

Após a criação da estrutura de mineração, o analista pode rodar um comando SQL *insert* com um *select* passando como parâmetros a estrutura de mineração e a base com os dados a serem processados. Apesar de ser muito prático, pois dá a possibilidade de se adicionar o código DMX ao código de uma aplicação. Este código não é a melhor opção para ser utilizado em trabalhos acadêmicos quando o objetivo do processo é obter resultados gráficos.

Abaixo estão alguns códigos DMX que foram utilizados na criação dos modelos de mineração de dados utilizados na classificação realizada pelo SQL Server 2008.

Tabela 9 - Códigos DMX utilizados no estudo

Código DMX	Aplicação
CREATE MINING STRUCTURE [EmailsProcessados] ([CorpoDoEmail] TEXT DISCRETE, [AssuntoDoEmail] TEXT DISCRETE, [SpamYN] BOOLEAN DISCRETE, [eMailID] LONG KEY) WITH HOLDOUT (30 PERCENT or 1000 CASES)	Cria a Estrutura de mineração *EmailsProcessados* com 4 atributos. No treinamento do modelo que utilizará esta estrutura, serão utilizados no máximo 30% 1.000 casos dos dados.
ALTER MINING STRUCTURE [EMailsProcessados] ADD MINING MODEL [NAIVE BAYES] (eMailID, [CorpoDoEmail], [AssuntoDoEmail], [SpamYN] PREDICT_ONLY) USING MICROSOFT_NAIVE_BAYES	Altera a estrutura anterior adicionando o modelo de mineração de dados Naive Bayes, indicando quais os dados devem ser utilizados e qual é, no caso, o que deve ser utilizado no treinamento do modelo.
ALTER MINING STRUCTURE [EMailsProcessados] ADD MINING MODEL [Arvore] (eMailID, [CorpoDoEmail], [AssuntoDoEmail], [SpamYN] PREDICT_ONLY) USING MICROSOFT_DECISION_TREES	Como o código acima, este adiciona um modelo de classificação dos textos baseado em árvore de decisão.
INSERT INTO MINING STRUCTURE [EMailsProcessados] ([CorpoDoEmail], [AssuntoDoEmail], [SpamYN], [eMailID]) OPENQUERY([Base_Limpa_MSC], 'SELECT body, subject, SpamYN, mail_id FROM dbo.v_in_queue ')	Este comando treina o modelo. Em DMX para se treinar um modelo de dados basta que seja realizado um insert no modelo de mineração existente.

Apesar das estruturas e modelos aqui apresentados terem sido criados com o DMX, as saídas apresentadas foram obtidas através do processamento do modelo na ferramenta de BI do SQL Server 2008, o *SQL Server Business Intelligence Development Studio* que na verdade é uma customização do *Visual Studio 2008* para o BI do *SQL Server 2008*. Ou seja, é uma interface gráfica para o DMX.

5.2.2. RESULTADOS

Foi verificado que os algoritmos da *Microsoft Naive Bayse* e o *Microsoft decision trees* analisam todo o corpo ou assunto do e-mail como um único elemento. Por exemplo, os assuntos "*compre viagra agora*" e "*compre agora viagra*" são completamente diferentes para estes algoritmos da Microsoft. Infelizmente estes aparentemente não foram preparados para analisar cada termo do texto como um item em uma coleção. Desta forma, os resultados obtidos não foram satisfatórios e os algoritmos não conseguiram classificar os e-mails como esperado.

Tabela 10 - Matriz de confusão da classificação do MICROSOFT_NAIVE_BAYES

	Spam	**Não Spam**
Spam	9.298 (100,00%)	0 (0,00%)
Não Spam	1.291 (97,51%)	33 (2,49%)

Tabela 11 - Matriz de confusão da classificação do MICROSOF_DECISION_TREES

	Spam	**Não Spam**
Spam	9.298 (100,00%)	0 (0,00%)
Não Spam	1.288 (97,28%)	36 (2,72%)

A tabela 10 é a matriz de confusão da classificação do Microsoft Naive Bayes, nesta as linhas representam o valor real e as colunas o valor predito. Como pode ser observado na mesma, os Spams foram classificados com uma precisão de 100%

enquanto os não Spams foram classificados com uma precisão de aproximadamente 2%. O mesmo comportamento pode ser observado na tabela 11, que é a matriz de confusão da classificação utilizando o Microsoft Decision Tree. Esta tendência de classificar todas as mensagens como Spam torna inviável a utilização destes algoritmos da Microsoft para classificar mensagens eletrônicas.

O gráfico 5 é o Gráfico de Ganho dos dois modelos de mineração. Apesar de parecer razoável, o mesmo apresenta este resultado, pois classifica corretamente todos os spams. Como os Spams representam 87,53% da amostra o gráfico mostra a classificação correta deste percentual dos dados. Podemos observar que o modelo criado pelo Naive Bayse classificou com menos precisão os dados que o de árvore de decisão. Enquanto o Naive Bayse teve um erro mais aparente a partir de 40% da amostra, a árvore de decisão começa a se afastar do modelo ideal em 60%.

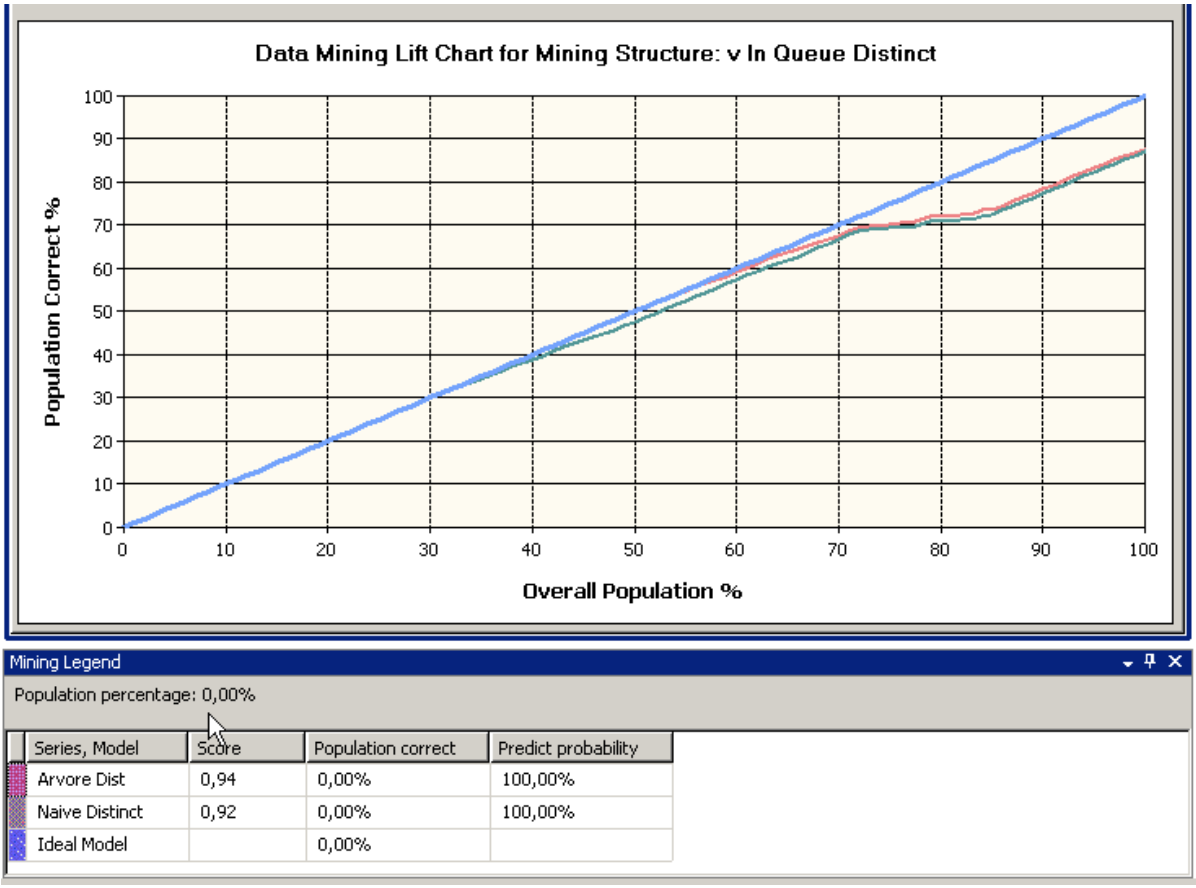

Gráfico 5 - Gráfico de Ganho dos dois modelos de classificação

Foram feitos alguns testes com os dados. Os classificadores foram treinados apenas com o assunto, apenas com o corpo e com ambos. Alem destes, foram feitos testes com assuntos e corpos de mensagens distintos. Ou seja, foram classificadas amostras sem mensagens repetidas. Em todos estes casos o acerto do classificador não variou mais de 0,5% frente os resultados aqui apresentados. Com a exceção do SVM do Statística 8, foi observado que as mensagens repetidas foram ignoradas pelos outros algoritmos classificadores.

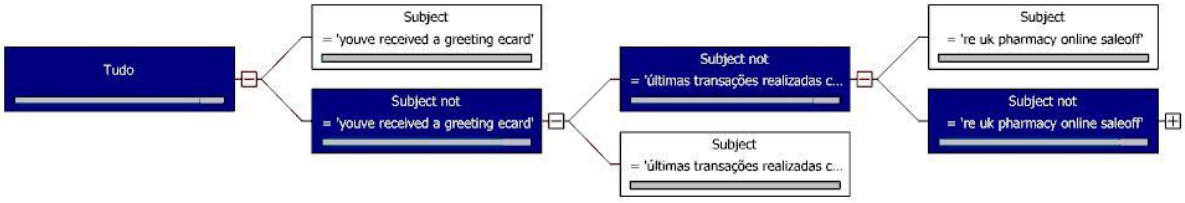

Figura 9 - Árvore construída com o Microsoft Decision Trees utilizando apenas o assunto como entrada.

Uma ferramenta que pode ser utilizada para montar o bag of words automaticamente é o serviço de integração de dados do SQL Server2008. Atualmente apenas funcional para o Inglês, esta ferramenta é muito prática de se configurar e de se trabalhar. Como pode ser observado na figura 10, para se montar um bag of words basta configurar quatro componentes deste. (origem dos dados, conversão de dados, extração de termos e base de destino.

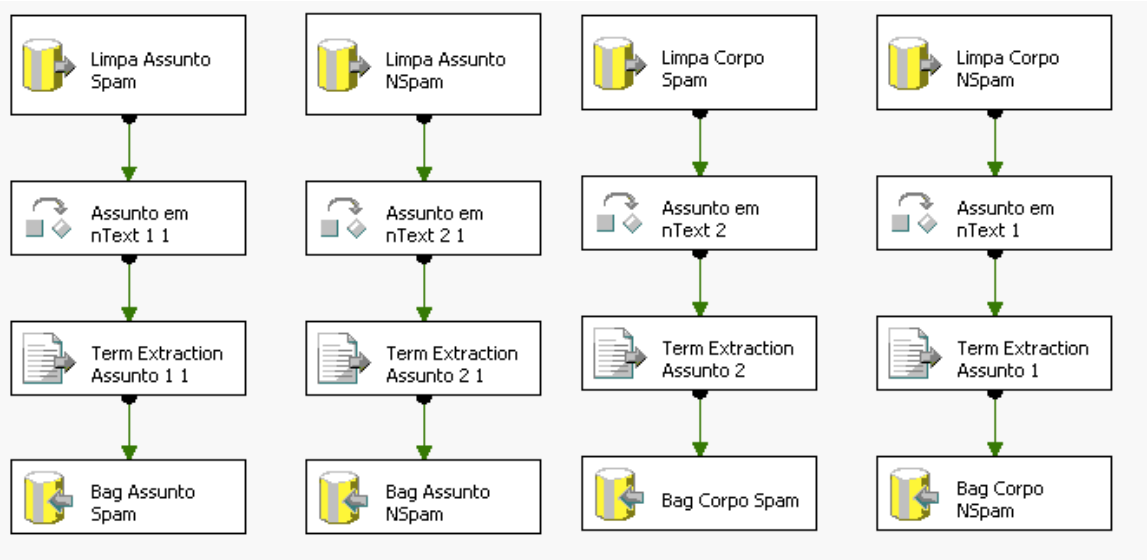

Figura 10 - Projeto de Integração de dados do SQL Server 2008

Esta etapa pode fazer parte de um conjunto maior de operações, que estariam ocorrendo em um nível mais abstrato. Ou seja, a etapa acima, poderia ser um módulo de um projeto de integração como pode ser observado no exemplo da figura 11.

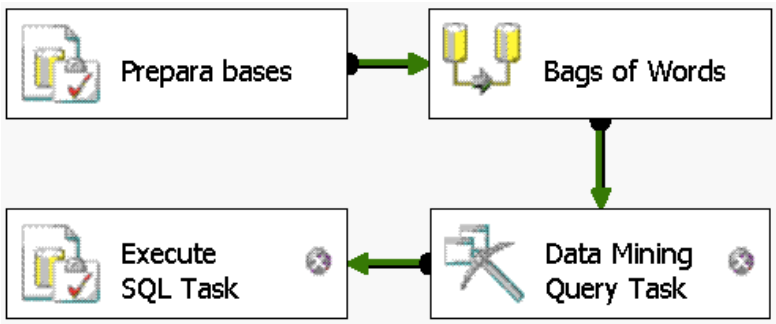

Figura 11 - Projeto de integração contendo a etapa de construção do Bag of Words

5.3. MINERAÇÃO DE TEXTOS COM O STATISTICA®

A fim de comparar os resultados obtidos com o SQL Server 2008, alguns testes foram realizados com o Statistica® 8.0 da StatSoft, Inc. Para realizar a classificação dos textos, o Statistica® faz a conversão do texto em números. Apos a redução da palavra ao

seu radical, o estatística converte cada um destes radicais para um número, através de um dicionário de termos, criando, desta forma, uma cópia da mensagem em formato numérico, ou melhor dizendo, em um array de números.

Com isto, os algoritmos de classificação do Statistica® podem ser utilizados sem que para isso os mesmos lidem com o texto propriamente dito.

As mensagens utilizadas foram classificadas no Statistica® por três algoritmos distintos: o Naive Bayes, o KNN (K-Nearest-Neighbors) e o SVM (Suport Vector Machine). Em todos os casos o resultado do Statistica® foi significativamente melhor que o resultado obtido com o SQL Server 2008. É válido ressaltar, que nenhum tratamento diferente do utilizado com o SQL Server 2008 foi realizado com as mensagens antes destas serem classificadas pelo Statistica®.

5.3.1. IMPORTAÇÃO E CLASSIFICAÇÃO DOS DADOS

O Statistica® 8 é habilitado à importar os dados diretamente de uma base do SQL Server utilizando uma conexão via ODBC. Após a mesma ser realizada o sistema coloca os dados em uma planilha que será utilizada como repositório para os algoritmos.

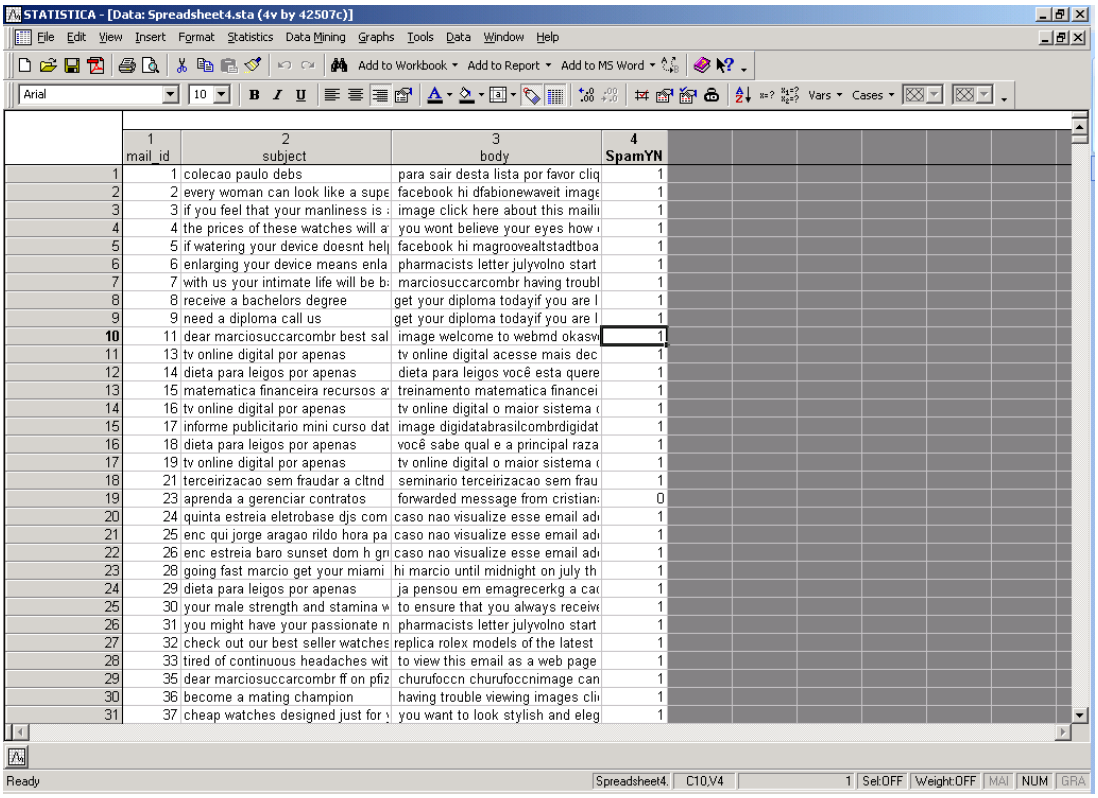

Figura 12 - Dados importados para o Statistica® a partir do SQL Server

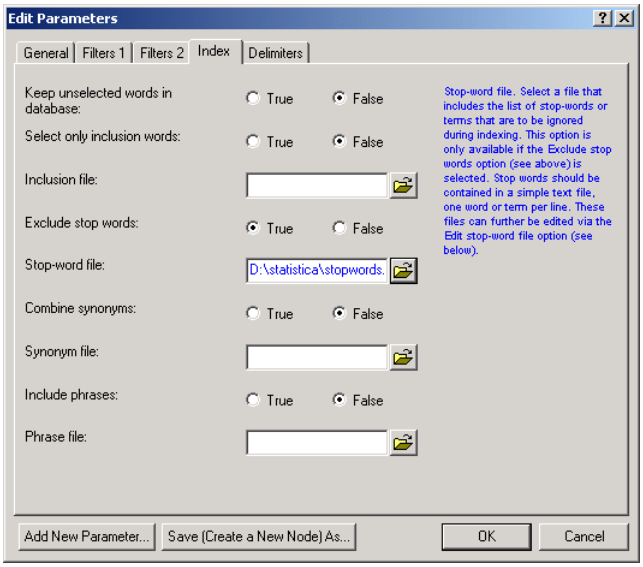

Figura 13 - Exemplo de janela de configuração específica para *text mining*

Como pode ser verificado na figura 13 o sistema possui campos específicos para se trabalhar com mineração de textos. Pode-se importar a lista das *stop words*, sinônimos etc. Após o processamento do texto e de sua conversão em uma matriz numérica, o sistema está habilitado a realizar a classificação do texto, utilizando, para isto, um dos algoritmos de classificação existentes no mesmo.

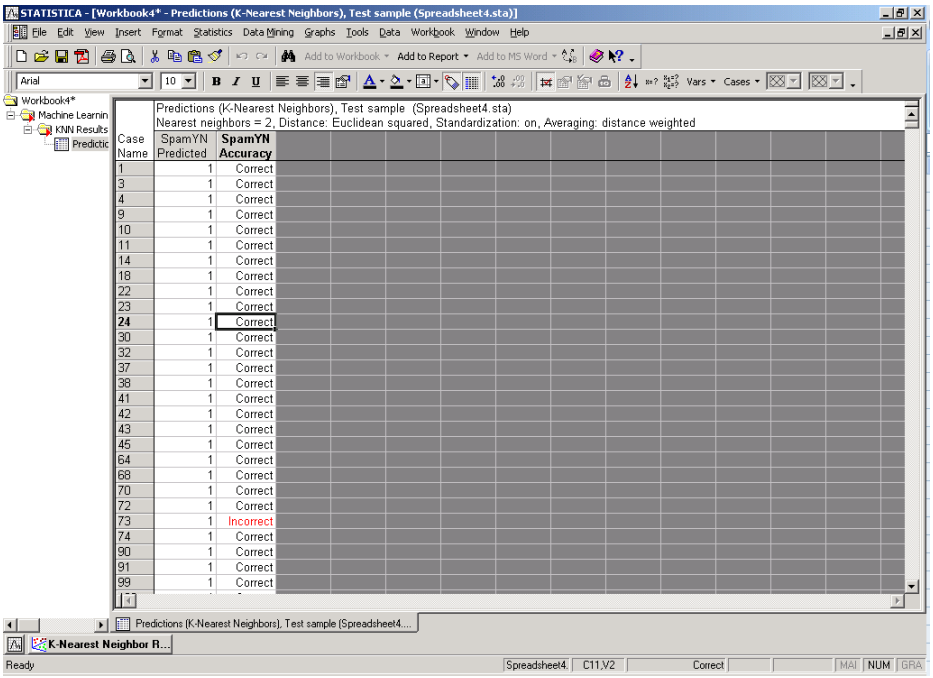

Figura 14 - Resultado do Classificador utilizando KNN

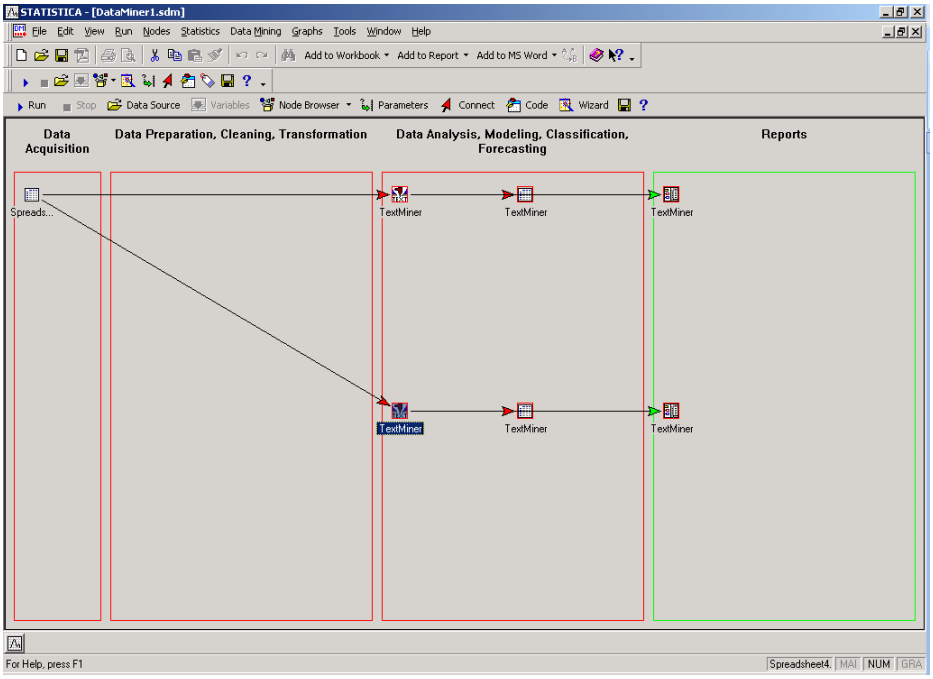

Figura 15 - Passos do processo de Mineração de Textos

Abaixo podem ser observados os resultados obtidos pelos três classificadores utilizados na classificação de textos do Statistica®: o Naive Bayes, o KNN e o SVM.

Independente do algoritmo utilizado os resultados dos três algoritmos do Statistica® foram muito melhores que os resultados obtidos com o SQL Server 2008.

5.3.2. NAIVE BAYES

Como pode ser observado na matriz de confusão abaixo o Naive Bayes do Statistica® obteve um bom resultado muito melhor que o Text Mining do SQL Server 2008.

Tabela 12 - resultado do Naive Bayes do Statistica®

	Spam	**Real**
Spam	7.173 (77,15%)	2.125 (22,85%)
Real	544 (41,09%)	780 (58,91%)

A classificação correta de aproximadamente 77% dos Spams e de aproximadamente 59% dos autênticos não é um resultado satisfatório para a classificação de Spams. Porém, este resultado mostra que o "mesmo" algoritmo da Microsoft precisa ser aprimorado quando o assunto a ser classificado é composto por textos. Pode ser observado que este algoritmo desconsiderou as mensagens iguais classificando apenas 10.622 e-mails.

Tabela 13 - resultado do Naive Bayes do SQL Server 2008 e do Statistica 8

	Microsoft Naive Bayes		**Statistica Naive Bayes**	
	Spam	**Não Spam**	**Spam**	**Não Spam**
Spam	9.298 (100,00%)	0 (0,00%)	7.173 (77,15%)	2.125 (22,85%)
Não Spam	1.291 (9751%)	33 (2,49%)	544 (41,09%)	780 (58,91%)

No Microsoft Naive Bayes, o resultado "surpreendente" de classificar 100% dos Spams corretamente é derrubado pelo fato deste algoritmo ter classificado apenas 2% das mensagens autênticas corretamente O que torna, na classificação de mensagens

eletrônicas, a utilização deste algoritmo inviável. Por outro lado, o resultado do Naive Bayes do Statistica apresenta um resultado mais equilibrado. Apesar de estar utilizando a mesma base de dados, o Statistica conseguiu classificar ambas as classes, apresentando um erro de aproximadamente 23% nos Spams e de aproximadamente 41% nas mensagens autênticas.

5.3.3. KNN

O resultado do KNN foi o melhor encontrado entre todos os classificadores utilizados. O erro de apenas 1,81% das mensagens autênticas e de 7,75% nos Spams mostra que este algoritmo, do Statistica 8, poderia ser utilizado para fazer a classificação de mensagens eletrônicas.

Tabela 14 - resultado do KNN do Statistica®

	Spam	**Autêntico**
Spam	8.577 (92,25%)	721 (7,75%)
Autêntico	24 (1,81%)	1.300 (98,19%)

Várias opções do KNN foram utilizadas e todos os resultados foram melhores que os dos demais algoritmos. Na tabela 14 é apresentado o resultado do KNN com um k=5 e com mais "peso" para os vizinhos mais próximos. Curiosamente, este resultado é o mesmo obtido com k=2 sem um maior peso para os vizinhos mais próximos.

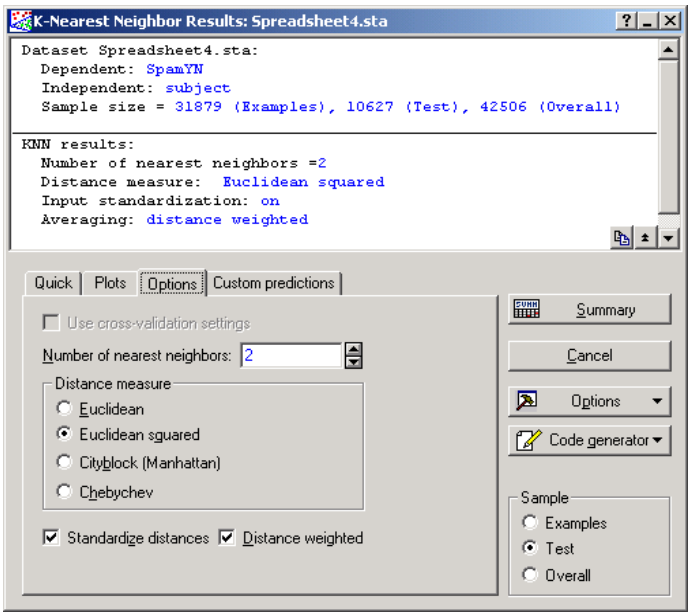

Figura 16 - parâmetros do KNN do Statistica®

Foram testados diferentes valores de k que variaram de 1 a 5. O valor de k que classificou melhor os e-mails foi o de k=2. Na tabela 15 são apresentados os diferentes resultados em função da variação do valor de k. Os percentuais apresentados representam o percentual de mensagens classificadas em cada classe em função do k utilizado. Ou seja, para k=5 aproximadamente 1% das mensagens autênticas foram classificadas como Spam e aproximadamente 99% das mensagens foram classificadas corretamente.

Tabela 15 - resultados obtidos com a variação do valor de k

		Spam		Autêntica	
		Mensagens	Percentual	Mensagens	Percentual
K=1	Spam	8.575	92,22%	723	7,78%
	Autêntica	25	1,89%	1.299	98,11%
K=2	Spam	8.577	92,25%	721	7,75%
	Autêntica	24	1,81%	1.300	98,19%
K=3	Spam	8.435	90,72%	863	9,38%
	Autêntica	23	1,74%	1.301	98,26%
K=4	Spam	8.436	90,73%	862	9,27%
	Autêntica	21	1,59%	1.303	98,41%
K=5	Spam	8.376	90,08%	922	9,92%
	Autêntica	14	1,06%	1.310	98,94%

Outra possível variação no KNN é o método de calculo da distância utilizada entre os elementos que estão sendo classificados. As opções são: Euclidiana, Euclidian Squared, Cityblock (Manhattan) e Chebychev. Foi realizado um teste com todas as distâncias com cinco vizinhos (k=5) e, em todos os casos o resultado foi o mesmo do K=5 da tabela 15.

Além dessa opção, existe no Statistica a possibilidade de se utilizar as opções abaixo:

- *Standardize Distance* que só deve ser utilizada quando os valores são de origem diferente (temperatura, pressão e umidade). Apesar dos valores serem todos textos esta opção foi selecionada e a mesma não influenciou no resultado.
- *Distance Weighted* dá um peso maior aos valores mais próximos. A utilização desta opção fez com que o algoritmo desse um resultado mais apurado. A tabela 16 apresenta esta diferença.

Tabela 16 - diferença do resultado com e sem o *Distance Weighted* com k=5

	Sem *Distance Weighted*		Com *Distance Weighted*	
	Spam	**Não Spam**	**Spam**	**Não Spam**
Spam	8.376 (90,08%)	922 (9,93%)	8.577 (92,25%)	721 (7,75%)
Não Spam	14 (1,06%)	1.310 (98,94%)	24 (1,81%)	1.300 (98,19%)

Como dito anteriormente k=5 com *Distance Weighted* apresenta o mesmo resultado de k=2 com ou sem *Distance Weighted*.

5.3.4. SVM

O SVM foi o único classificador que analisou 43.690 mensagens. Apesar de ter analisado todas as mensagens que foram disponibilizadas para o Statistica, o resultado do SVM foi pior que o resultado do KNN. A característica de se classificar ou de se ignorar mensagens iguais não é muito relevante neste estudo, pois isso pode ser facilmente implementado em uma ferramenta de classificação de mensagens eletrônicas. Mais importante que classificar efetivamente as repetidas é classificar as mesmas corretamente.

Tabela 17 – Resultado da classificação do SVM do Statistica®

	Spam	**Autêntica**
Spam	38.201 (99,93%)	26 (0,07%)
Autêntica	4.440 (81,27%)	1.023 (18,73%)

Segundo [15], um dos maiores problemas do SVM é a definição da superfície de decisão que maximiza as margens entre as classes. Isso torna o processo de classificação muito custoso. Além de apresentar uma classificação ruim, o SVM levou em média

quatro vezes mais tempo do que os outros classificadores, provavelmente pela dificuldade do mesmo em criar a citada superfície de decisão.

Tabela 18 - Tabela comparativa dos resultados do KNN e do SVM

	KNN		SVM	
	Spam	**Autêntico**	**Spam**	**Autêntica**
Spam	8.577 (92,25%)	721 (7,75%)	38.201 (99,93%)	26 (0,07%)
Autêntico	24 (1,81%)	1.300 (98,19%)	4.440 (81,27%)	1.023 (18,73%)

Como pode ser observado na tabela 18 o KNN foi muito mais preciso que o SVM na classificação de mensagens autênticas. O SVM aparentemente tendeu para a classe mais numerosa melhorando, em função disso, o acerto de Spams, mas piorando a classificação de mensagens autênticas. Classificar corretamente apenas 18,73% das mensagens autênticas faz com que este algoritmo seja menos interessante que o KNN.

6. CONCLUSÃO

Este trabalho de pesquisa teve como objetivo avaliar o comportamento da metodologia de mineração de textos em um conjunto de mensagens eletrônicas. Estas deveriam ser classificadas como Spams ou mensagens autênticas. As mensagens eletrônicas (e-mails) foram obtidas a partir de várias contas pessoais. As mesmas foram importadas em algumas contas do Gmail a fim de se classificar estas mensagens para que esta classificação fosse utilizada no treinamento dos algoritmos utilizados.

Após serem classificadas pelo Gmail, as mensagens eram importadas por um aplicativo, desenvolvido durante este trabalho, que marcava as mensagens como Spam ou não Spam, de acordo com a conta do Gmail (figura 7) da qual as mensagens estavam sendo importadas. As mensagens foram armazenadas no banco de dados relacional do SQL Server 2008. Após armazená-las foi realizado o pré-processamento dos textos das mensagens para que fosse possível a execução da mineração das mesmas.

Após serem pré-processadas as mensagens foram classificadas pelos algoritmos do SQL Server 2008 e pelos algoritmos do Statistica 8. Ambos os sistemas possuem qualidades distintas e complementares. O SQL Server é um dos melhores sistemas gerenciadores de banco de dados e, por causa disso, consegue processar os dados com muita eficiência e segurança. Enquanto o Statistica®, por ser uma ferramenta específica de análises estatísticas e de mineração de dados estruturados ou não estruturados, apresentou uma interface mais amigável e ao final apresentou um resultado mais preciso.

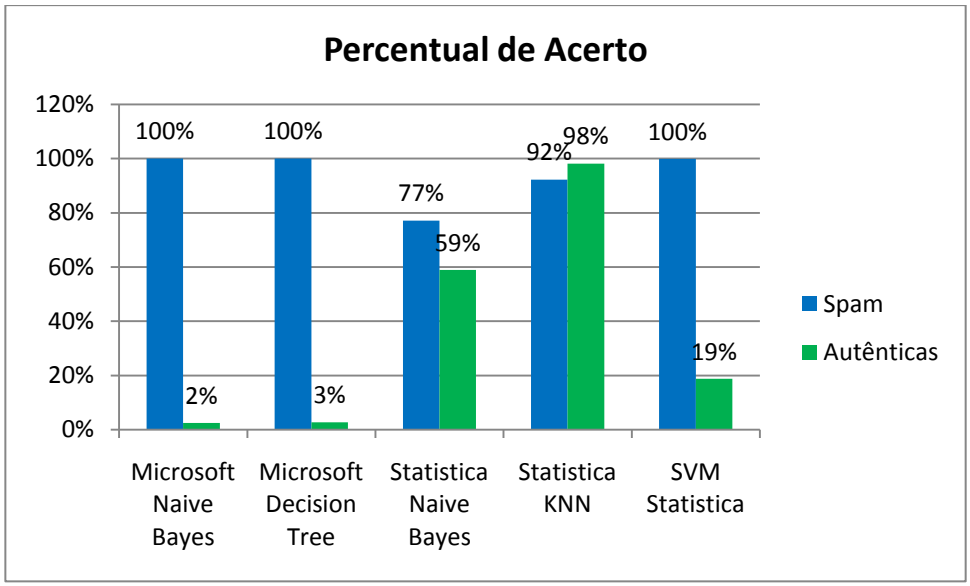

Gráfico 6 – Percentual de acerto dos algoritmos

No gráfico 6 pode ser observado o percentual de acerto dos algoritmos analisados. O KNN do Statistica apresentou o melhor resultado de classificação na classe mais importante, classificando 98% das mensagens autênticas corretamente, e obteve um resultado aceitável na classificação da classe menos importante, a classe dos Spams. Os algoritmos da Microsoft e o SVM do Statistica classificaram corretamente 100% dos Spams, porém classificaram muito mal as mensagens autênticas, logo classificaram muito mal a classe mais importante dos dados.

Para que o SQL Server tenha o mesmo funcionamento, ou melhor, o mesmo desempenho, do Statistica®, é necessário que seja desenvolvido um aplicativo que faça o pré-processamento do texto, explicado no capítulo 4 e o transforme em dados numéricos para que o classificador do SQL Server 2008 de dados estruturados possa ser utilizado. Este aplicativo deverá detectar o idioma da mensagem e classificá-la em função do resultado de uma média de avaliação dos classificadores do SQL utilizados.

Sistemas de detecção de Spam corporativos têm o valor de mercado bem elevado, pois devem ser capazes de filtrar todas as mensagens coorporativas muito

rapidamente. O SQL Server é um dos sistemas de banco de dados mais utilizados do mercado e se o classificador de textos do mesmo for utilizado isso pode baratear muito a implantação de um sistema de detecção de mensagens eletrônicas não solicitadas que, utilizando os serviços do SQL Server poderia detectar Spams de forma rápida e eficiente.

Para se classificar dados não estruturados no SQL Server 2008 é necessário que parte da ferramenta de classificação de dados não estruturados seja desenvolvida, utilizando para isso uma linguagem de programação como C# .Net (capítulo 5.2), e é necessário que sejam utilizados códigos DMX na implementação da parte de mineração de dados do SQL Server.

Independente da ferramenta utilizada foi demonstrado neste trabalho que é possível se classificar mensagens eletrônicas utilizando a mineração de textos. A falta de precisão na classificação por parte da maioria dos algoritmos mostra que esta é uma tarefa muito árdua, e que a escolha dos algoritmos de classificação deve ser feita com de forma a minimizar os erros de classificação.

6.1. TRABALHOS FUTUROS

O desenvolvimento deste trabalho mostrou que a classificação de mensagens eletrônicas é um processo muito árduo e que é necessário que os algoritmos utilizados sejam muito treinados e tenham seus parâmetros configurados com muita precisão. Abaixo estão algumas sugestões que podem ser implementadas e testadas em trabalhos futuros:

- Criação de uma ontologia que monte a relação semântica entre os termos utilizados nos Spams e nas mensagens autênticas (tesauros);
- Trabalhar com um dicionário de termos e bag-of-words bilíngues;
- O mesmo código numérico para palavras com o mesmo sentido.

- Criação de um aplicativo que faça o pré-processamento completo dos textos e utilizando DMX classifique os mesmos;
- Reconhecimento automático do idioma da mensagem e fazer a redução das palavras aos seus respectivos radicais (stemming) em função deste.
- Utilizar ontologias/taxonomias na redução dos radicais das palavras, detecção de idiomas, definição de pesos etc.

7. REFERÊNCIAS BIBLIOGRÁFICAS

[1] Estatísticas do governo americano. **Internet Crime Complaint Center**. http://www.ic3.gov - Acessado em 20 de maio de 2010.

[2] Google Inc, **Gmail uses Google's innovative technology to keep spam out of your inbox**. < http://www.google.com/mail/help/fightspam/spamexplained.html> Google® 2010

[2] EDWARDS, JOHN, It Security, **The Real Cost of Spam**, http://www.itsecurity.com/features/real-cost-of-spam-121007/, Acessado em agosto de 2010.

[3] **Anti-spam Comparison Report**, Jan 2009, Test Laboratory Details. <http://fr.trendmicro.com/imperia/md/content/uk/whitepaper/wp06_wclantispamrpt_090317us.pdf>

[4] Comitê Gestor da Internet no Brasil: **Tipos de Spam**. Disponível em: <www.antispam.br/tipos/>. Acesso em: Setembro de 2009.

[5] **Canning spam: Proposed solutions to unwanted email** PFLEEGER, S. L., E BLOOM, G. Canning spam: Proposed solutions to unwanted email. *IEEE Security & Privacy Magazine 3*, 2 março 40–47. (2007)

[6] Microsoft Corporation, **Microsoft Security Intelligence Report**, Volume 8, (Julho 2009) 107 – 128

[7] **Internet 2009 in numbers**, Royal Pingdom, <http://royal.pingdom.com/2010/01/22/internet-2009-in-numbers/>,acessado em maio de 2010.

[8] It Security, **The Real Cost of Spam**, http://www.itsecurity.com/features/real-cost-of-spam-121007/ , Acessado em agosto de 2010.

[9] **Spam** (Monty Python), < http://en.wikipedia.org/wiki/Spam_(Monty_Python)>, acessado em maio de 2010.

[10] **Cartilha de Segurança para a Internet, Centro de Estudos, Resposta e Tratamento de Incidentes de Segurança no Brasil**. http://cartilha.cert.br/spam/sec1.html#subsec1.3, acessado em: maio de 2010.

[11] LOWD, D., MEEK, C. (2005), "**Good word attacks on statistical spam filters**". Em Second Conference on Email and Anti-Spam (CEAS2005) (julho).

[12] CHAWLA, N.V.; Japkowicz, N.; Kotcz, A. (2004) (Editors) Editorial: **Special Issue on Learning from Imbalanced Data Sets**. ACM SIGKDD Explorations. v.6. p.1-6.

[13] KONCHADY, MANU (2006), "**Text Mining Application Programing**", Primeira Edição, Thomson, Boston, Massachusetts

[14] Gnedenko, Boris Vladimirov, (2008), "**A Teoria da Probabilidade**", Ed. Ciência Moderna, primeira edição.

[15] COLAS, F., BRAZDIL, P., **Comparison of SVM and Some Older Classification Algorithms in Text Classification Tasks**. Chile IFIP - WCC 2006

[16] **As lendas, as verdades e as meias-verdades / meias-mentiras**. Disponível em: <www.quatrocantos.com/LENDAS/index_crono.htm>. Acesso em Setembro de 2009.

[17] Comitê Gestor da Internet no Brasil: **História: Origem e curiosidades** <http://www.antispam.br/historia/>. Acesso em maio de 2010.

[18] **What is usenet exactly**? < http://www.usenet.org/usenet.html>, acessado em maio de 2010

[19] Wired, May 1, 1978: **Spam, From Novelty to Nuisance in a Couple of Decades**. < http://www.wired.com/science/discoveries/news/2007/05/dayintech_0501>, Acessado em maio de 2010.

[20] EMERY, T. MIT (2007), **conference takes aim at spam emails**. *Associated Press*.

[21] **Phishing**, Wikipedia, <http://en.wikipedia.org/wiki/Phishing>, acessado em maio 2010

[22] Forouzan , Behrouz A. and Fegan, Sophia Chung, (2009), "**Protocolo TCP/IP**", 3 edição, editora MacGraw Hill

[23] Heuser, Carlos Alberto, (1999), **Projeto de Banco de Dados**, Ed. Sagra&Luzzatto, Porto Alegre,

[24] Bratko, A., Cormack, G., Filipic, B., Lynam, T., and Zupan, B. Spam filtering using statistical data compression models. Journal of Machine Learning Research 7 (Dec. 2006).

[25] HOANCA, B. (2006),**How good are our weapons in the spam wars?** IEEE Technology and Society Magazine 25-1,22-30

Livros Grátis

(http://www.livrosgratis.com.br)

Milhares de Livros para Download:

Baixar livros de Literatura
Baixar livros de Literatura de Cordel
Baixar livros de Literatura Infantil
Baixar livros de Matemática
Baixar livros de Medicina
Baixar livros de Medicina Veterinária
Baixar livros de Meio Ambiente
Baixar livros de Meteorologia
Baixar Monografias e TCC
Baixar livros Multidisciplinar
Baixar livros de Música
Baixar livros de Psicologia
Baixar livros de Química
Baixar livros de Saúde Coletiva
Baixar livros de Serviço Social
Baixar livros de Sociologia
Baixar livros de Teologia
Baixar livros de Trabalho
Baixar livros de Turismo