9ª MOSTRA ACADÊMICA UNIMEP
08 a 10/11 de 2011
Tema: "Ambiente e Sustentabilidade"

19 Congresso de Iniciação Científica

DESENVOLVIMENTO DE UM MECANISMO EFICIENTE DE CAPTURA E ANÁLISE DE COMENTÁRIOS NA WEB

Autor(es)

JEFFERSON DIAS DOS SANTOS

Orientador(es)

PLÍNIO ROBERTO SOUZA VILELA

Apoio Financeiro

FAPIC/UNIMEP

1. Introdução

Muitas redes estão presentes no nosso cotidiano, redes de supermercados, de trens, de esgoto, de celulares. Os sistemas baseados na rede mundial de computadores – a Internet – vem crescendo e tornando-se cada vez mais presente como ferramentas de comunicação e de interação social.

No relacionamento e nas interações entre pessoas, é que as idéias, visões do mundo, opinião e valores vão sendo construídos e moldados. E para tanto atualmente as redes sociais baseadas em Internet, tem papel importante. Uma rede social é uma estrutura dinâmica constituída por um "conjunto de atores e suas conexões" (RECUERO, 2005), onde esses atores trocam informações, experiências e interagem.

Pela ausência de informações que caracterizam um ator pessoal, nas redes sociais baseadas em Internet é necessária a criação de um vínculo da pessoa com o ator virtual. No ambiente virtual, os perfis representam um ator pessoal e são preenchidos com informações como gostos, paixões, vontades e características físicas da pessoa que aquele perfil representa.

Os usuários da Internet deixam um rastro de comportamento, preferências, mudanças de humor, e hábitos através de sua participação em *chats*, fóruns, comunicadores instantâneos, ferramentas de redes sociais e outras formas de comunicação online (RECUERO, 2005). O rastro deixado por esses usuários é uma interessante forma de se analisar e avaliar produtos e serviços.

Pensando nisso Macedo (2010) desenvolveu o IMV – Índice de Avaliação de Opinião Macedo-Vilela que trabalha com os comentários das redes sociais, mais especificamente o *Orkut*, capturando e os classificando em positivo, negativo ou neutro, e através de uma equação gera um índice.

A ferramenta que Macedo desenvolveu não capturava um grande volume de dados. Para isso nesse projeto, foi desenvolvido uma ferramenta chamada de *DownTweets*, a qual acessa a rede de microblogs *Twitter* e realiza a busca e captura de comentários, utilizando mecanismos que a própria rede fornece. A ferramenta foi pensada para capturar a maior quantidade possível de posts dessa rede.

Como a ferramenta obtém um grande volume de comentários, é necessário selecionar os mais importantes e impactantes. Com esse intuito foram selecionadas informações sobre os usuários e sobre os comentários, que foram agrupadas em três grandes categorias: influência, engajamento e relevância. Com os parâmetros dessas categorias é possível obter subsidio para posteriormente realizar a ordenação e seleção dos *posts* mais relevantes.

2. Objetivos

Com o crescente uso das redes sociais e o dinamismo do *Twitter*, uma grande quantidade de comentários é deixada pelos usuários dessa rede. Este trabalho objetivou o estudo da API provida pelo *Twitter* e meios de como utilizá-la para a captura do maior volume de *tweets*; bem como os limites que são impostos por essa rede. A fim de desenvolver um mecanismo que fosse capaz de acessar a rede e com base em palavras chaves trouxesse o máximo possível de comentários em que houve a ocorrência de tais termos. Não só a captura dos comentários é necessária, mas também a sua análise e classificação. Para tanto foi levantando um conjunto de dados que contribuem para a organização do conjunto de comentários capturados.

3. Desenvolvimento

O *Twitter* e sua API

O *Twitter* é uma rede social muito dinâmica e possui um diferencial frente às outras redes, que é o seguir e ser seguido. Seguir é uma forma de acompanhar os *posts*, ou como são chamados os *tweets* de outros usuários.
O *Twitter* fornece mecanismos para facilitar o desenvolvimento de aplicativos que interajam com a rede. O mecanismo, ou API (*Application Programming Interface*, Interface para programação de aplicação), trabalha sobre o protocolo HTTP, o mesmo que os navegadores *web* utilizam. A API possui três partes:

- API *REST*: é usada para acessar dados tais como informações dos usuários, *status* dos usuários e atualizações da timeline.
- API *Search*: Ou de busca, permite ao desenvolvedor interagir com o mecanismo de busca.
- API *Streaming*: permite acesso próximo de tempo real a um grande volume de dados para *tweets*.

De modo a não esgotar os recursos dos servidores, o *Twitter* permite um número limitado de requisições. Tais requisições e acessos aos recursos da API do *Twitter* são contados e debitados de um valor de acessos permitidos por hora, esse valor é chamado de *Rate Limit*.

API de busca

A API de buscas permite realizar buscas, utilizando palavras chaves, sobre os *tweets* dos últimos 8 dias. O resultado da busca é trazido de forma paginada, e para acessar todos os resultados é necessário navegar por todas as páginas retornadas. O limite máximo de posts, que são retornados é de 1500. Ela é muito dinâmica e permite customizações, como especificar o número de resultados por página, a página que se deseja visualizar, período no qual o *tweet* foi postado entre outros.

DownTweets

Para realizar a captura dos comentários foi estudado o mecanismo de busca da rede *Twitter* e seus limites de acesso, que levou a elaboração de uma ferramenta que utiliza ao máximo os recursos fornecidos por esta rede para obter um grande volume de *tweets*. Atualmente a ferramenta é capaz de permanecer em execução por um longo período de tempo capturando *posts*.
A aplicação desenvolvida chamada de *DownTweets* (Figura 1), foi codificada em Java utilizando-se a biblioteca Twitter4j, que possibilita realizar acesso aos recursos do *Twitter* a partir de um programa Java de maneira bem simplificada. A ferramenta tem o objetivo de baixar *tweets* baseados em palavras chave, para isso realiza uma busca, traz um conjunto de *tweets* e armazena as informações do *post* e do usuário. Para armazenar os dados baixados foi usado o gerenciador de banco de dados *MySql*.
Foi realizado um experimento no qual a ferramenta mostrou-se eficiente. Permaneceu em execução por um período de 74 dias, onde capturou a quantidade de 663.047 *tweets* sobre diversos temas.

Classificação dos comentários

Para a classificação dos comentários foram tentadas duas abordagens.

1 Classificação de Macedo

Utilizando alguns conceitos de um processador de conversa, o algoritmo que Macedo (2010) propôs foi implementado. O algoritmo funciona da seguinte maneira: Obtém a lista de *tweets* e para cada um, separa-o em *tokens* os quais são confrontados frente a uma base de adjetivos e palavras (a mesma que Macedo utilizou). Caso seja um adjetivo verifica-se a classificação do mesmo, se for antecedido de uma palavra negativa o valor do mesmo é invertido, ou seja, se for negativo torna-se positivo e se for positivo torna-se negativo.
Em um primeiro experimento foram separados manualmente 100 *posts* negativos, 100 *posts* neutros e 100 *posts* positivos. Para a seleção dos mesmos, foi tomado o cuidado de separar os que não possuíssem erros de grafia e gírias.
O algoritmo (Figura 2) cometeu alguns erros de classificação, dentre estes os mais perigosos são os falso positivo, pois são *posts* que são negativos e foram classificados como positivos. Pensando em um ambiente onde existe um monitoramento e uma intervenção aos *tweets* que falam de uma determinada marca ou produtos é necessária, os positivos seriam ignorados, pois a atenção maior é dada aos que falam mal, negativos.

Do conjunto de *tweets* pré-selecionados e pré-classificados, o algoritmo classificou:

• Dentre os positivos: 41 como positivos, 2 como negativos e 57 como neutros.
• Dentre os negativos: 13 como positivos, 18 como negativos e 69 como neutros.
• Dentre os neutros: 10 como positivos, 0 como negativos e 90 como neutros.

Verifica-se que a grande maioria foi classificada como neutros, notavelmente um erro, pois este fato ocorre não somente nos neutros. Algumas hipóteses sobre os erros do algoritmo podem ser levantadas:
O algoritmo não esta bom, a ponto de conseguir classificar os posts? Deve se tentar uma abordagem onde são analisadas as palavras no seu contexto?
A base de adjetivos e palavras não contém termos o suficiente? Faltam palavras e/ou adjetivos na base?

2 Classificação segundo uma nota
Para a classificação segundo uma nota foi necessário a adaptação da aplicação de modo que capturasse informações adicionais, pois foi verificou-se que era possível obter informações adicionais sobre o *post* e sobre o usuário que o postou.
Essas informações foram agrupadas em classes. Segue a definição de cada classe e os parâmetros que as compõem.

Influência
É a capacidade de uma pessoa influir sobre outra, isto é, a ação ou pensamento que determinada pessoa pratica e que acaba por interferir, mesmo que inconscientemente, de maneira positiva ou não no comportamento cotidiano de outrem. É composta de alguns fatores como, por exemplo, a proximidade: quanto mais próximas forem elas, como amigos ou parentes, maiores são as chances dessa característica se expressar.
Quanto esta posição em contraste com a relevância da informação postada, pode-se dizer que quanto maior for a importância do que está escrito no *post* para quem lê, maior será a sua influência. Se não houver confiança na pessoa que influi, o que ela fala simplesmente é ignorado.
A influência tem forte ligação com a popularidade e status: quanto mais popular e conhecido for, mais influente será tal pessoa em uma rede. No *Twitter* seria a “capacidade de um usuário de gerar comportamentos nos seguidores”. (RECUERO).

Os parâmetros que compõem essa classe são:
Seguidores: quantidade de seguidores que um dado usuário possui.
Alcance: quantidade de usuários que um *post* atingiu, ou, a distância geográfica que um *tweet* alcançou.
Listas: listas que um dado usuário foi adicionado.

Relevância
A relevância é a característica dada a algo ou alguém que é importante e possuí certo valor, ou seja, uma informação é relevante quando condiz com o indispensável num contexto ou para determinada pessoa. Ao repassar uma informação é dado maior valor a ela, ou seja, maior relevância. A importância ou relevância de uma informação é algo que tem relação íntima com o ambiente onde ela se encontra e com os envolvidos com ela.

Os parâmetros que compõem essa classe são:
Retweets: posts que um usuário repassa aos seus seguidores.
Favoritos: post que foi marcado como favorito.
#FF, *Follow Friday*: essa hashtag é usada para indicar usuários interessante de serem seguidos. Faz parte de uma tradição de toda a sexta feira no *Twitter*.

Engajamento
É o ato de ser participante em um lugar ou situação. Antes de se tornar engajado em algo primeiramente é necessário estar presente. Estando presente e tendo um mínimo de contanto é um primeiro passo para ser engajado. Interagir com freqüência e regularidade são as formas que mais claramente demonstram o engajamento, ou seja, sai do papel de apenas um mero observador e passar a ser uma presença ativa.

Os parâmetros que compõem essa classe são:
Replies: respostas a um determinado *tweet*.
Menções: quantidade de *tweets* que possuem um @username de um usuário.
Status: quantidade total de *tweets* postados por um usuário.
Listas: quantidade de listas que um usuário criou.
Favoritos: quantidade de posts que um usuário marcou como favorito.

O cálculo da nota
A fim de tentar medir o impacto de um post e tentar diminuir possíveis problemas de uma informação que fala mal de um produto e/ou serviço no *Twitter*, o levantamento de dados que dão indícios do que um *tweet* pode gerar de movimentação na rede é um

importante aliado na tomada de decisões. E nesse âmbito uma ferramenta que fique monitorando e realizando verificações do possível impacto que um comentário desse tipo pode ter é algo que se faz necessário.
Tomando como base o conjunto de informações que levam em consideração aspectos do *tweet* e do usuário que realizou o *post*, e que estão agrupadas nas classes Influência, Relevância e Engajamento, é possível obter subsídios para o cálculo de uma nota ao *tweet*.

4. Resultado e Discussão

Com o crescente uso da Internet, cresce também a utilização dos meios virtuais para a comunicação. Dentre estes, encontram-se as redes sociais, que atualmente constituem um dos principais meios onde os usuários podem expor preferências, gostos e opiniões através dos *posts*; e estes se devidamente capturados e analisados, fornecem subsidio para se indicar como um produto ou serviço está sendo falado.
Pensando nisso foi desenvolvida a ferramenta *DownTweets* que acessa a rede *Twitter* e captura comentários de um determinado produto ou serviço. Após a captura a análise é outra etapa necessária para que sejam identificados os mais relevantes dentre todos do montante capturado. Para tal análise nesse projeto foi realizado um experimento onde se utilizou a idéia de Macedo (2010) para classificar os comentários como positivo, negativo ou neutro, baseado nos adjetivos. Devido ao grande número de fatores que podem dificultar tal análise, como erros ortográficos, figuras de linguagem e gírias, outra abordagem foi necessária, classificar os comentários segundo uma nota. Para realizar tal tarefa foi necessária a captura de informações adicionais sobre o usuário e o *tweet*.
Como resultado foi compilado uma tabela onde tais informações foram agrupas em três grandes categorias: influência, engajamento e relevância.

5. Considerações Finais

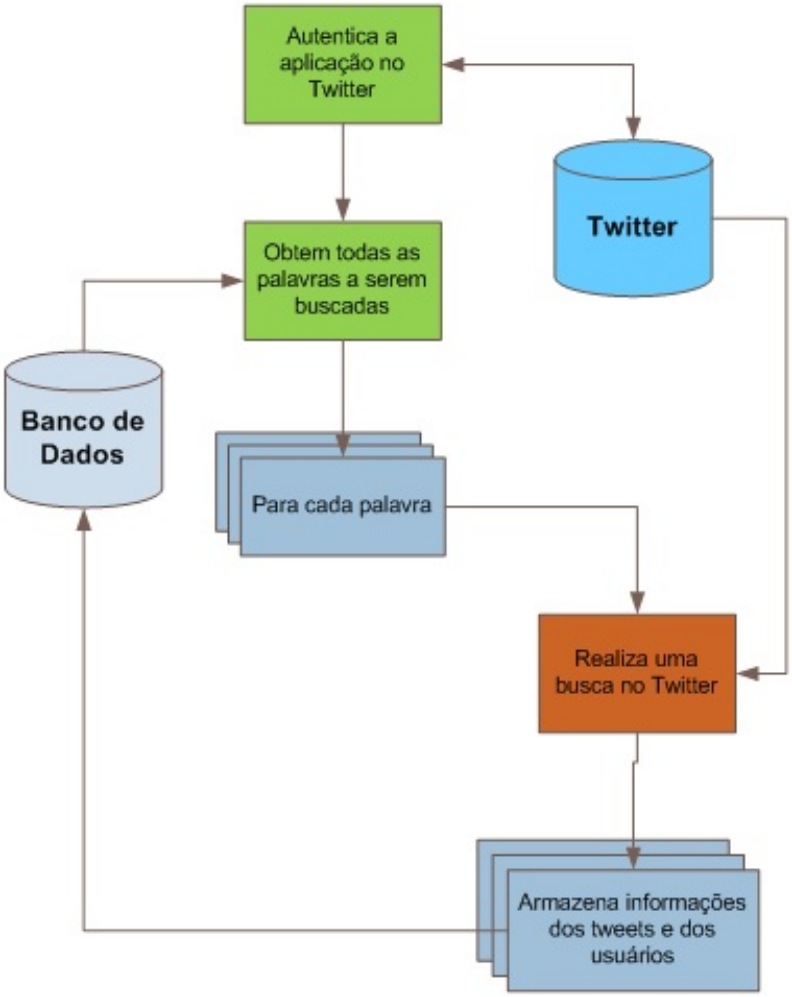
Ao estudar redes sociais, mais em especifico o *Twitter*, foi verificado que existe uma quantidade muito grande de informação e que estas não são aproveitadas adequadamente. Os objetivos desse projeto foram estudar meios de como extrair informações do *Twitter* e desenvolver uma ferramenta que fosse capaz de capturar um grande volume *posts* baseados em certas palavras, mas sempre levando em consideração os limites impostos pela API da rede.
A captura de *posts* e o monitoramento em redes sociais é um trabalho muito oneroso se feito manualmente, e muitas vezes é feito dessa forma; e com o grande volume de informação presente nas redes encontrar meios de organizar e filtrar as informações de maior relevância é uma tarefa de grande importância.
Esperamos que com a compilação dos grupos de parâmetros e a ferramenta *DownTweets*, resultados desse trabalho, ajudar com as atividades de monitoramento de comentários no *Twitter*, e que os resultados dessa pesquisa venham a ajudar em trabalhos futuros.

Referências Bibliográficas

COSTA, L.; JUNQUEIRA, V.; MARTINHO, C.; FECURI, J. “Redes: Uma introdução às dinâmicas de conectividade e da auto-organização”. Brasilia, 2003: WWF-Brasil. 1ª Ed. 91p.
MACEDO, Paulo Cesar. “Mecanismo Automático de Conceituação de Empresas e Produtos em Redes Sociais”. Tese de Mestrado, FACEN-UNIMEP, Piracicaba – SP, Brasil, 2010.
RECUERO, Raquel. “Redes Sociais na Internet”. Porto Alegre; Editora Sulina, 2009. 191p.
RECUERO, Raquel. “Redes Sociais na Internet: Considerações Iniciais”. Ecompos, v.2, 2005.
SUERETH, Russell. “Developing Natural Language Interfaces: Processing Human Conversation”. Editora McGraw-Hill.
Twitter. "API Documentation | dev.twitter.com". Disponível em: . Acesso em: novembro de 2010.
TWitter4j. "TWitter4J - A Java library for the Twitter API". Disponível em: . Acesso em: novembro de 2010.

Anexos

```
ListaTweets ← Obter Tweet da base de dados;
Flag InverterValor ← Falso;
Para cada Tweet
        Separar o Tweet em tokens;
        Para cada token Então
                Se for adjetivo Então
                        Se peso for < 5 Então
                                Se inverterValor = Verdadeiro Então
                                        Classificacao ← Positivo;
                                Senao
                                        Classificacao ← Negativo;
                                FimSe
                        Senao Se peso = 5 Então
                                Classificacao ← Neutro;
                        Senao Se peso > 5 Então
                                Se interverValor = Verdadeiro Então
                                        Classificacao ← Negativo;
                                Senão
                                        Classificacao ← Positivo;
                                FimSe
                        FimSe
                        inverterValor = Falso;
                FimSe
                Se for Palavra Então
                        Se peso = -1 Então
                                inverterValor = Verdadeiro;
                        FimSe
                FimSe
        FimPara
FimPara
Retorna Classificacao;
```


Autentica a aplicação no Twitter
Twitter
Obtem todas as palavras a serem buscadas
Banco de Dados
Para cada palavra
Realiza uma busca no Twitter
Armazena informações dos tweets e dos usuários