UNIVERSIDADE FEDERAL DE OURO PRETO
INSTITUTO DE CIÊNCIAS EXATAS E APLICADAS
DEPARTAMENTO DE COMPUTAÇÃO E SISTEMAS DE INFORMAÇÃO

MATHEUS MOREIRA DA SILVA

# MINERAÇÃO DE DADOS NO TWITTER: UMA FERRAMENTA PRÁTICA PARA EXTRAÇÃO E ANÁLISE DOS RESULTADOS

João Monlevade
2017

MATHEUS MOREIRA DA SILVA

# MINERAÇÃO DE DADOS NO TWITTER: UMA FERRAMENTA PRÁTICA PARA EXTRAÇÃO E ANÁLISE DOS RESULTADOS

Monografia apresentada ao curso Sistemas de Informação do Instituto de Ciências Exatas e Aplicadas, da Universidade Federal de Ouro Preto, como requisito parcial para aprovação na Disciplina “Trabalho de Conclusão de Curso II”.

Orientador: Fernando Bernardes de Oliveira

João Monlevade

2017

Universidade Federal de Ouro Preto
Instituto de Ciências Exatas e Aplicadas
Colegiado do Curso de Sistemas de Informação
Campus João Monlevade

**ATA DE DEFESA**

Aos 29 dias do mês de março de 2017, às 17 horas e 30 minutos, na sala C304 do Instituto de Ciências Exatas e Aplicadas, foi realizada a defesa de Monografia pelo aluno **MATHEUS MOREIRA DA SILVA,** sendo a Comissão Examinadora constituída pelos professoros: Prof. Dr. Fernando Bernardes de Oliveira, Profª. Msc. Helen de Cassia Sousa da Costa Lima e Profª. Msc. Daniela Rodrigues Dias.

O candidato apresentou a monografia intitulada: *"MINERAÇÃO DE DADOS NO TWITTER: UMA FERRAMENTA PRÁTICA PARA EXTRAÇÃO E ANÁLISE DOS RESULTADOS".* A comissão examinadora deliberou, por unanimidade, pela aprovação do candidato, com nota 10,0 (Dez), concedendo-lhe o prazo de 15 dias para incorporação das alterações sugeridas ao texto final.

Na forma regulamentar, foi lavrada a presente ata que é assinada pelos membros da Comissão Examinadora e pelo graduando.

João Monlevade, 29 de março de 2017.

Prof. Dr. Fernando Bernardes de Olivera
Professor Orientador/Presidente

Profª. Msc. Helen de Cassia Sousa da Costa Lima
Professor Convidado

Profª. Msc. Daniela Rodrigues Dias
Professor Convidado

Matheus Moreira da Silva
Graduando

Universidade Federal de Ouro Preto
Instituto de Ciências Exatas e Aplicadas
Colegiado do Curso de Sistemas de Informação
Campus João Monlevade

**Folha de Aprovação**

**Curso de Sistemas de Informação**

**FOLHA DE APROVAÇÃO DA BANCA EXAMINADORA**

**MINERAÇÃO DE DADOS NO TWITTER: UMA FERRAMENTA PRÁTICA PARA EXTRAÇÃO E ANÁLISE DOS RESULTADOS**

**Matheus Moreira da Silva**

**Monografia apresentada ao Instituto de Ciências Exatas e Aplicadas da Universidade Federal de Ouro Preto como requisito parcial da disciplina CSI499 – Trabalho de Conclusão de Curso II do curso de Bacharelado em Sistemas de Informação e aprovada pela Banca Examinadora abaixo assinada:**

Prof. Dr. Fernando Bernardes de Oliveira
DECSI – UFOP

Profª. Msc. Helen de Cassia Sousa da Costa Lima
DECSI – UFOP

Profª. Msc. Daniela Rodrigues Dias
DECSI – UFOP

João Monlevade, 29 de março de 2017.

UNIVERSIDADE FEDERAL DE OURO PRETO
INSTITUTO DE CIÊNCIAS EXATAS E APLICADAS
COLEGIADO DO CURSO DE SISTEMAS DE INFORMAÇÃO

**ANEXO III – Termo de Responsabilidade**

## TERMO DE RESPONSABILIDADE

Eu, Matheus Moreira da Silva, declaro que o texto do trabalho de conclusão de curso intitulado "**MINERAÇÃO DE DADOS NO TWITTER: UMA FERRAMENTA PRÁTICA PARA EXTRAÇÃO E ANÁLISE DOS RESULTADOS**" é de minha inteira responsabilidade e que não há utilização de texto, material fotográfico, código fonte de programa ou qualquer outro material pertencente a terceiros sem as devidas referências ou consentimento dos respectivos autores.

João Monlevade, 05 de Abril de 2017

Matheus Moreira da Silva
Assinatura do aluno

# DEDICATÓRIA

Dedico esse trabalho aos meus pais, Aníbal e Consolação, por todos os sacrifícios que fizeram para que eu conseguisse chegar aqui.

Às minhas irmãs, Gabriela e Juliana, por terem me incentivado sempre que eu estive desanimado e precisando de palavras positivas.

A minha namorada Viviane, que me incentivou e auxiliou em todas as dificuldades e me deu todo apoio desde o meu primeiro dia estudando nesta instituição.

Aos meus amigos e demais familiares, pelo incentivo e pelo apoio constante.

# AGRADECIMENTOS

Agradeço a Deus primeiramente por ter me dado saúde e força para superar as dificuldades.

À Universidade Federal de Ouro Preto, seu corpo docente, direção e administração pela oportunidade de fazer este curso.

Ao meu orientador, Prof Dr. Fernando Bernardes de Oliveira, pelo suporte e pela paciência durante a execução deste trabalho, pelas suas correções e incentivos.

À ArcelorMittal, ao Lucas Vilela meu coordenador de estágio e aos demais colegas de setor, pela oportunidade de estagiar nesta conceituada empresa que me permitiu crescer pessoalmente e amadurecer a minha visão profissional.

A todos os meus familiares, pelas orações, paciência e incentivos para a conclusão deste curso.

Ao professor, Msc. Eduardo da Silva Ribeiro, pela oportunidade de ser monitor da disciplina de Banco de Dados I.

À Flávia Cristina Miguel Reis, pela oportunidade de ser instrutor no projeto de inclusão digital da biblioteca do ICEA.

## RESUMO

As redes sociais são compostas por relações entre usuários que geram um amplo domínio de informações que são modificadas com uma rápida frequência. Um exemplo de rede social é o *Twitter*, o qual oferece uma comunicação interativa rápida entre seus usuários em uma escala global. Observa-se que as postagens oriundas das interações entre os usuários do *Twitter*, quando submetidas a um tratamento de dados, podem recuperar informações importantes sobre o comportamento desses usuários. Assim, este trabalho visa desenvolver uma ferramenta prática e funcional para a mineração e visualização de dados do *Twitter*. Os dados são extraídos por meio da *API* disponibilizada pelo *Twitter* que permite recuperar *Tweets* recentemente publicados. Os dados extraídos são submetidos a uma coleção de algoritmos de mineração de dados visando encontrar associações relevantes entre os dados. Os resultados obtidos são expostos aos usuários por meios de bibliotecas gráficas como *Google Charts* e *D3.js* com o intuito de facilitar a visualização e análise. A aplicação foi testada sobre temas da atualidade e se mostrou funcional uma vez que foi capaz de realizar as tarefas requisitadas da maneira esperada.



**Palavras-chave:** *Twitter, Data Mining, Web Mining.*

## ABSTRACT

*Social Networks are composed by relations between users that generate a broad domain of information, where are modified with a quick frequency. The example is Twitter, which offers a fast and interactive communication with users on a global scale. It's observed that the posts originating from the interactions between Twitter users, when submitted to a processing data, it's possible retrieve important information about users behavior. This work aims to develop a practical and functional framework for the mining and visualization of Twitter data. The data are extracted by API available by Twitter, it allow extracts the recent tweets published. The extracted data are submitted to a collection of data mining algorithms, in order to find relationships between the data. The results are exposed to users by means of graphic libraries such as: Google Charts and D3.js in order to facilitate the visualization and analysis. The application was tested on current topics and proved to be functional once it was able to perform the requested tasks as expected.*



**Keywords:** *Twitter, Data Mining, Web Mining.*

# LISTA DE FIGURAS

# LISTA DE QUADROS

# LISTA DE ABREVIATURAS

| | | |
|---|---|---|
| API | - | *Application Programming Interface* |
| BSON | - | *Binary JavaScript Object Notation* |
| DM | - | *Data Mining* |
| HTTP | - | *Hypertext Transfer Protocol* |
| JSON | - | *JavaScript Object Notation* |
| KDD | - | *Knowledge Discovery in Databases* |
| KDT | - | *Knowledge Discovery in Text* |
| MVC | - | Modelo-Visão-Controlador |
| PNL | - | Processamento de Linguagem Natural |
| REST | - | *Representational State Transfer* |
| SVG | - | *Scalable Vector Graphics* |
| WCA | - | *Web Crawlers Agents* |

# SUMÁRIO

# 1 INTRODUÇÃO

O *Twitter* é uma rede social com aproximadamente 310 milhões de usuários ativos mensalmente, e recebe aproximadamente um bilhão de visitas únicas por mês. O *Twitter* tem como característica ser um *microblog*, no qual os usuários podem escrever mensagens curtas com até 140 caracteres que podem conter elementos alfanuméricos ou multimídia, chamadas *Tweets*. “O *Twitter* tem como missão: capacitar todos os usuários a criar e compartilhar ideias e informações instantaneamente, sem qualquer barreira” (*TWITTER*, 2016a).

Considerando o grande volume de dados criados e modificados por dia nesta rede social, diversos estudos e ferramentas foram criados para a recuperação dos dados gerados e, a *posteriori*, para se encontrar padrões entre os dados. Santos (2010) propõe o desenvolvimento de um protótipo capaz de realizar mineração de opiniões em textos de redes sociais. Essa mineração em textos foi corroborada por Gonçalves *et al*. (2013), que exemplificam em seus estudos questões de análise de sentimentos aplicado ao *Twitter*, com o uso do ambiente de desenvolvimento *RapidMiner*. Existem também outras ferramentas comerciais que realizam estudos de padrões estatísticos sobre o *Twitter*, tais como: *Audiense, The Archivist, Tweepsmap, Socialopinion* e *Twitter Analytics*.

O *Twitter* em si fornece duas distintas *Application Programming Interface (API*) para recuperação dos dados: *a Representational State Transfer (REST) API* e a *Streaming API*. As *API’s* permitem aos desenvolvedores acessarem atualizações, *status*, dados de usuários e uma amostra dos *Tweets* publicados. (*TWITTER*, 2016c).

Amo (2004) aponta que uma vez que os dados tenham sido extraídos, as etapas seguintes consistem em especificar o que se deseja buscar nos dados, tais como: qual tipo de regularidade ou categoria de padrões deseja-se analisar. Uma vez que se tenha especificado o que se deseja buscar, analisam-se quais algoritmos são mais favoráveis a tais tipos de dados. Rezende *et al.* (2003) ressaltam que, dependendo dos tipos de dados escolhidos, pode ser necessário a execução dos algoritmos de extração de padrões diversas vezes ou até mesmo técnicas de combinações de algoritmos.

Diante das inúmeras técnicas disponíveis para a extração desses padrões, a proposta deste trabalho é desenvolver uma ferramenta que combine as técnicas para a extração de dados na rede social *Twitter* e algoritmos de análise de padrões. Para isso, será

necessário o auxílio de combinações de bibliotecas gráficas, que serão adotadas com a finalidade de criarem uma melhor visualização e análise dos resultados obtidos.

## 1.1 Problema

Os usuários de rede sociais costumam manter as suas contas sempre atualizadas. Diante disso Santos (2010) aponta que com essa interação entre as pessoas, uma enorme quantidade de dados é gerada todos os dias. Assim, pelo grande volume de dados, a busca de um tipo específico de informação acaba se perdendo dentre o vasto conteúdo disponível. Em seu cerne, a proposta do *Twitter* é permitir uma troca de informações aberta e em tempo real entre os usuários. Por conseguinte, como elucidado por Santos (2010), uma vasta quantidade de dados também são gerados sobre os comportamentos dos usuários e a amplitude de suas interações (locais e globais).

O problema relacionado à grande quantidade de informações em rede social está centrado na dificuldade em como filtrar essas informações que correm em um fluxo constante, e em como recuperar apenas o conteúdo que se deseja. Ademais, bem como resumir de maneira clara e representativa a imensa quantidade de dados encontrada. Camilo e Silva (2009) apresentam que uma das primeiras atividades na mineração de dados a ser feita é se obter uma visualização dos dados, de modo que se possa ter uma visão geral, para depois decidir-se quais as técnicas são mais indicadas para a análise. Após a identificação do tipo de informação que se pretende obter, as próximas etapas requerem estudos e decisões no que diz a respeito em como as tarefas serão executas. Assim, as decisões consistem das seguintes questões: quais as maneiras de se obter estes dados? (posto que o *Twitter* possui distintas *API's* com diferentes comportamentos), quais recursos deverão ser utilizadas para se realizar a análise? (tendo em vista vez que se tem inúmeras ferramentas, bibliotecas e *frameworks* para gerar a análise de padrões) e, por fim, como representar as informações geradas?

## 1.2 Objetivos

O objetivo geral deste trabalho é o desenvolvimento de uma ferramenta capaz de extrair dados da rede social *Twitter*, permitindo a visualização dos dados de maneira prática e funcional. A ferramenta deverá seguir alguns passos desde a extração até a exibição dos resultados, sendo: extração de dados, armazenamento, pré-processamento, análise de padrões e exibição dos resultados.

### 1.2.1 Objetivos específicos

O trabalho contempla os seguintes objetivos específicos:

1. Realizar o levantamento bibliográfico dos artigos relacionados, métodos de extração, análise e exibição dos resultados;
2. Demonstrar os métodos e técnicas de extração de dados no *Twitter*;
3. Definir uma modelagem da ferramenta para extração e análise dos dados provenientes da *REST API* do *Twitter*;
4. Apresentar os mecanismos de visualização dos dados e avaliar os mesmos, por exemplo: *frameworks*, bibliotecas e *API's*;
5. Analisar a ferramenta desenvolvida e realizar testes da mesma sobre algum tema da atualidade.

## 1.3 Justificativa

A importância das redes sociais pode ser interpretada sob a ótica de diversos pontos de vista. Russell (2014) ressalta a importância da rede social *Twitter* como uma ferramenta de *marketing* devido o número de usuários e a imensidão de informações geradas todos os dias. Förster *et al.* (2014) discutem a influência do *Twitter* como uma ferramenta de comunicação para conectar as pessoas. Desse modo, surge o interesse em se investigar os *Tweets* dos usuários em diferentes cidades do mundo; e, sugere que as características únicas ou compartilhadas nessas cidades poderiam ser analisadas em comparação com outras cidades.

Neste sentido análises de dados em rede sociais podem evidenciar como determinados eventos são recebidos pela população e sua respectiva repercussão. Segundo Oliveira *et al.* (2012) com o uso de técnicas de mineração de dados e descoberta de opiniões em rede sociais, torna-se possível o auxilio no desenvolvimento de campanhas de *marketing* e publicidade para cada grupo de interesse dos usuários. Assim, por meio de técnicas computacionais é possível se extrair informações que derivam destes dados. Porém, a extração não é trivial, embora existam inúmeras ferramentas complexas que fazem este processo.

Camilo e Silva (2009) apresentam a importância da mineração de dados para setores como iniciativa privada, o setor público e o terceiro setor também podem ser beneficiados. A ideia é corroborada por Thomé (2008), que cita áreas nas quais a Mineração de Dados é aplicada de forma satisfatória, tais como: *Telemarketing*, Vendas, Finanças.

Assim, a ferramenta desenvolvida para este trabalho tem o intuito de facilitar a extração, processamento e a análise de padrão sobre *Tweets* de um determinado tema. A mesma foi desenvolvida com a finalidade de ajudar as pessoas que necessitam de uma ferramenta prática, rápida e de uso simples, para a geração de variadas análises e exibição dos resultados, por meio de algoritmos de análise de padrões.

## 1.4 Estrutura do trabalho

O restante deste trabalho é organizado como segue. O Capítulo 2 apresenta os conceitos gerais sobre a descoberta de conhecimento em banco de dados, a descoberta de conhecimento em textos, mineração de dados na *Web* e mineração de dados em rede sociais. O Capítulo 3 apresenta a metodologia e o desenvolvimento da ferramenta para realizar a extração, persistência e análise dos dados extraídos da rede social *Twitter*. Os testes e resultados da ferramenta proposta se encontram no Capítulo 4. Por fim, as conclusões do trabalho e as propostas de continuidade são abordadas no capítulo 5.

# 2 CONCEITOS GERAIS E REVISÃO DA LITERATURA

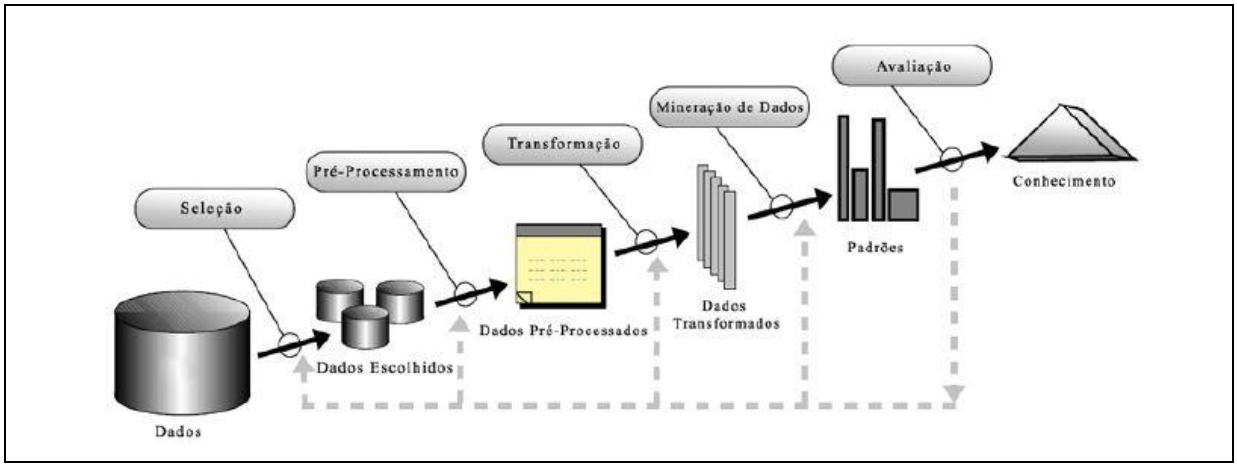
Neste capitulo será abordado a revisão bibliográfica, no qual apresentará definições para os termos *Knowledge Discovery in Databases*, *Knowledge Discovery in Texts, Web Mining*, *Text Mining* e Mineração em Rede Social, explicando de maneira sucinta do que são constituídas essas abordagens.

Na Seção 2.1. serão apresentados alguns conceitos sobre a descoberta de conhecimento em bases de dados, no qual se encontram as etapas fundamentais que seguem como base para o desenvolvimento da ferramenta proposta. Conceitos relacionados à descoberta de conhecimento em textos serão apresentados na Seção 2.2. Na Seção 2.3. é apresentada a base teórica sobre mineração de dados e mineração na *Web*. Na Seção 2.4. é apresentada a significância da mineração em rede social e os conceitos sobre extração de dados no *Twitter*.

## 2.1 *Knowledge discovery in databases*

O grande volume de dados gerados ultimamente acarretou na criação de modelos para transformação destes dados em informação, com a finalidade de extrair um conteúdo relevante dentre um conjunto de dados brutos. O *Knowledge Discovery in Databases* (*KDD*) ou Descoberta de Conhecimento em Bases de Dados é uma tentativa de solucionar o problema causado pela chamada "era da informação", à sobrecarga de dados (FAYYAD *ET AL.*, 1996). Ainda segundo Fayyad *et al.*(1996) e também de acordo com Soares (2008), o *KDD* engloba etapas que produzem conhecimentos a partir de dados relacionados e sua principal característica é a extração não trivial de informações e conhecimentos implicitamente contidos em uma base de dados. Fayyad *et al.*(1996) apresentam o *KDD* sendo composto por um conjunto de etapas que requererem do usuário a capacidade de análise e de tomada de decisão, e que as fases dos *KDD* muitas das vezes passam por constantes ciclos até que um resultado útil seja alcançado. As principais fases do processo segundo aquele autor podem ser visualizadas na Figura 1.

Figura 1: O processo de *KDD*.




Fonte: Fayyad *et al.*(1996).

De acordo com a Figura 1, as etapas podem ser descritas como:

- **Seleção**: são selecionados os dados a serem utilizados na busca por padrões e na geração de conhecimento.
- **Pré-processamento**: ocorre o tratamento e a preparação dos dados para uso. Também se identifica e retira os valores inválidos, inconsistentes ou redundantes.
- **Transformação**: geralmente são aplicadas técnicas de redução de dimensionalidade e de projeção dos dados.
- **Mineração**: consiste em encontrar padrões através de algum algoritmo e técnicas computacionais.
- **Avaliação**: análise dos resultados da mineração e na geração de conhecimento pela interpretação e utilização dos resultados em benefício do negócio.

## 2.2 *Knowledge discovery in text*

O *KDD* possui alguns processos similares, por exemplo, o *Knowledge Discovery in Text* (*KDT*) ou Descoberta de Conhecimento em Texto. O *KDT* é composto por uma combinação de diversas técnicas e métodos para extrair conhecimento sobre base de textos. Segundo Barion (2015), o *KDT* pode ser visto como uma extensão da área de *KDD*, focada na análise de textos. O que difere os processos são os tipos de dados no *KDT*: a mineração de textos pretende extrair conhecimentos úteis de dados não estruturados, e o

*KDD* que procura descobrir padrões emergentes em banco de dados estruturados. Como exemplificado por Barion (2015), dados estruturados podem se encontrar organizados em linhas e colunas, como geralmente são encontrados em banco de dados relacionais. Já os dados não estruturados referem-se a dados que não podem ser organizados em linhas e colunas, como vídeos, comentários em redes sociais e *e-mails*, entre outros.

Silva (2013) e Barion (2015) corroboram onde expõem que o *KDT* combina técnicas e conhecimentos de diversos segmentos, tais como; Informática, Estatística, Linguística, Matemática e outras, tendo a capacidade de extrair conhecimento a partir de grandes coleções de textos.

## 2.3 *Data mining e Web mining*

Os conceitos citados nas Seções 2.1 e 2.2 têm uma etapa crucial em comum, o *Data Mining*, que consiste em um conjunto de técnicas que procuram padrões e relacionamentos nos dados. Ele é uma das principais etapas de um processo da descoberta de conhecimento. O processo de *Data Mining* baseia-se em modelos computacionais com o intuito de descobrir automaticamente novos fatos e relações entre dados. O processo de mineração deve ser iterativo, interativo e divido em fases, seguindo um conjunto de etapas durante todo o processo. O *Data Mining* tal como o *KDT* são processos multidisciplinares e alguns autores abordam definições distintas sobre conceitos sobre o mesmo. Thomé (2008) explana que o *Data Mining* consiste de um conjunto de algoritmos e técnicas específicas que são capazes de produzir como resultado um modelo e a enumeração de padrões que se correlacionem com determinados fatos ou fenômenos. Fayyad *et al.* (1996) já definem a Mineração de Dados como um passo no processo de *KDD* que consiste na realização da análise dos dados e na aplicação de algoritmos de descoberta que, sob certas limitações computacionais, produzem um conjunto de padrões de certos dados. Hand *et al.* (2001) expõem a Mineração de Dados como a análise de grandes conjuntos de dados de modo a encontrar relacionamentos inesperados e de resumir os dados de modo que os mesmos sejam tanto úteis quanto compreensíveis ao seus donos.

O processo de *Data Mining* foi inicialmente criado para trabalhar com grandes volumes de dados em: bancos de dados, *Data Werehouse*, arquivos e outros meios estruturados dos dados. Porém, atualmente os dados estão em inúmeros formatos sejam eles estruturados ou não.

Quando se trata de informações dispostas em dados não estruturados, é válido citar como exemplo as vastas informações que se encontram na *Web*. As informações encontradas *online* podem apresentar uma quantidade imensurável de páginas, contendo vários tipos de informações e em vários idiomas. Para o estudo e análise deste conteúdo foi desenvolvido o conceito de *Web Mining*, derivado da expressão *Data Mining*. Segundo Camilo e Silva (2009), a *Web Mining* tem sido alvo de recentes pesquisas, pois ela reúne em seu ambiente, quase a totalidade dos tipos de estruturas complexas e simples que existem. Além disso, possui um volume de dados gigantesco que atendem às diversas necessidades e possui os mais diversos conteúdos. A *Web Mining* consiste em minerar as estruturas de ligação, o conteúdo, os padrões de acesso, classificação de documentos, entre outras.

Quando múltiplos conceitos como *KDT* e *Web Mining* são relacionados, o cenário para a descoberta de conhecimento em texto torna-se uma ferramenta crucial. A crucialidade se dá pela praticidade na extração de informações a partir de bases textuais sem a necessidade de leitura, tais como: opiniões em rede sociais, preferências das pessoas e até mesmo dados pessoais e geográficos dos mesmos a partir de suas iterações com rede sociais.

A extração de dados na *Web* não é um processo trivial. Esse processo pode ser realizado por distintos métodos, seja por *Web Crawlers Agents* (*WCA*) ou também por *API's*. De acordo com Magalhães (2008) *WCA* são programas que automaticamente vasculham páginas *Web* para colherem informações que podem ser analisadas e mineradas em um local *on-line* ou *off-line*. Por outro lado, a extração de dados na *Web* também pode ser realizada com o auxílio de *API's,* que são definidas como conjuntos de rotinas e padrões de programação para acesso a um aplicativo de *software* ou plataforma baseado na *Web*.

## 2.4 Mineração em rede social

As redes sociais são compostas por relações entre usuários que geram um amplo domínio de informações que são modificadas com uma rápida frequência. Análises em rede sociais podem mensurar de alguma maneira o que se passa em certa localidade ao longo dos tempos e ainda podem servir como complemento de pesquisas sociológicas (SILVA, 2013).

O *Twitter*, por exemplo, oferece uma comunicação interativa e rápida entre seus usuários em uma escala global. Observa-se que as postagens oriundas das interações entre os usuários do *Twitter*, quando submetidas a um tratamento de dados podem recuperar

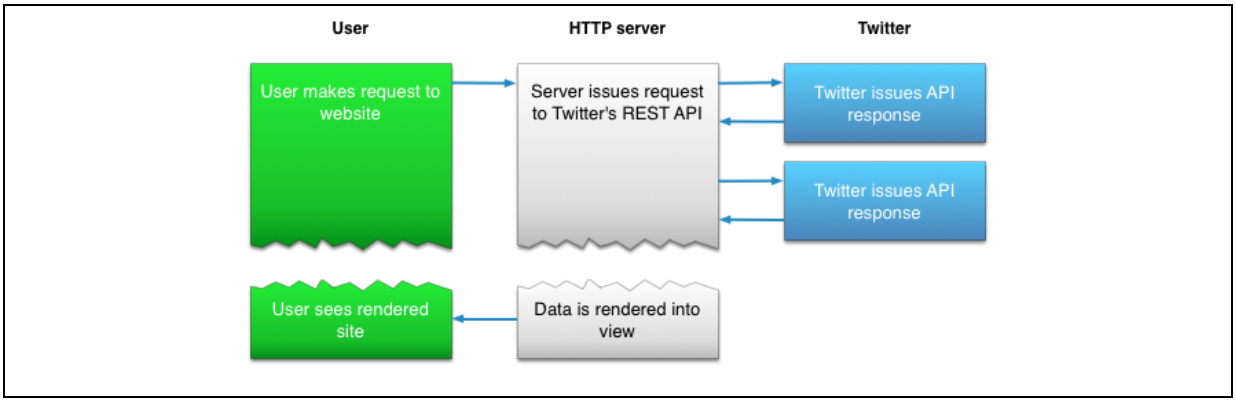
informações importantes sobre o comportamento desses usuários. Cabral *et al.* (2015) afirmam que em plataformas como o *Twitter*, os usuários tendem a se expressarem livremente com o uso das #*hashtags* (marcadores que agrupam *Tweets*), o que cria um meio ideal de se capturar as opiniões comuns sobre diversos tópicos.

De acordo com o estudo de Gonçalves *et al.*(2013), muitos usuários de redes sociais, como os do *Twitter*, tendem a expressarem sentimentos em suas mensagens com a utilização de *emoticons*. Por conseguinte, mensagens contendo *emoticons* quando analisadas em grande escala podem ser utilizadas para medir variações de humor do público coletivo. Essa análise tem como objetivo compreender como os sentimentos variam de acordo com eventos sociais, econômicos, políticos, etc.

A extração de dados na rede social *Twitter* pode ser realizada com o auxílio das *API's* disponibilizadas pelo mesmo: a *REST API* e a *Streaming API*. A *REST API* permite recuperar *Tweets* recentes publicados nos últimos dias, e permite uma busca por: #*hashtags*, assunto, localidades, usuário, dentre outros (*TWITTER*, 2016b). A *API* utiliza do *Search API* para que a recuperação seja realizada. O *Twitter Search API* faz parte do *REST API* do *Twitter*. Ele permite consultas em relação aos índices dos *Tweets* recentes ou populares publicados nos últimos sete dias. O uso do *Streaming API* do *Twitter* se difere, porque requer uma conexão ativa e persistente entre o cliente e o servidor do *Twitter*. Os dados neste tipo de *API* são extraídos em tempo real sem que se tenha o atraso da fila da *REST API* (*TWITTER*, 2016c).

O que distingue ambos é o tipo de conexão estabelecida. Quando se utiliza a *REST API*, cada requisição de mensagem abre e fecha uma requisição de conexão *Hypertext Transfer Protocol* (*HTTP*). Por exemplo, considere um aplicativo da *Web* que aceite solicitações de usuários, faça uma ou mais solicitações à *API* do *Twitter* e, em seguida, formate e imprima o resultado para o usuário, como uma resposta à solicitação inicial do usuário, como ilustra a Figura 2.

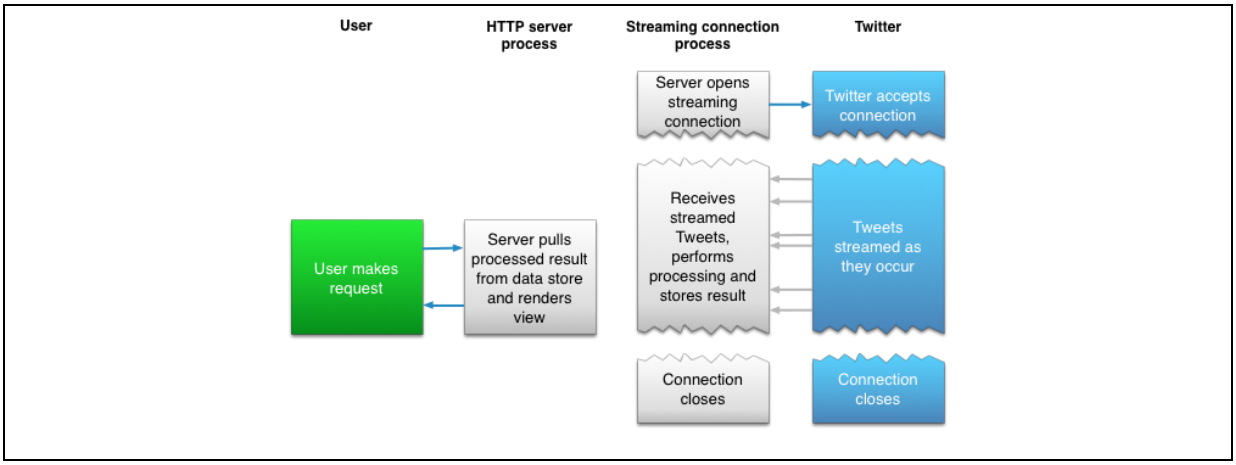
Figura 2: Exemplo da *API REST*.




Fonte: *Twitter* (2016).

O exemplo acima ilustra o processo de conexão pela *REST API*. Quando comparado os processos de conexão se diferem. A Figura 3 ilustra o processo de conexão pela *Streaming API*. Ela não é capaz de abrir e fechar uma requisição a cada solicitação, como mostrado na *REST API*. Em vez disso, o *Streaming API* mantém a conexão aberta por um processo paralelo e os *Tweets* são atualizados por notificações de um processo a parte.

Figura 3: Exemplo do *Streaming API.*

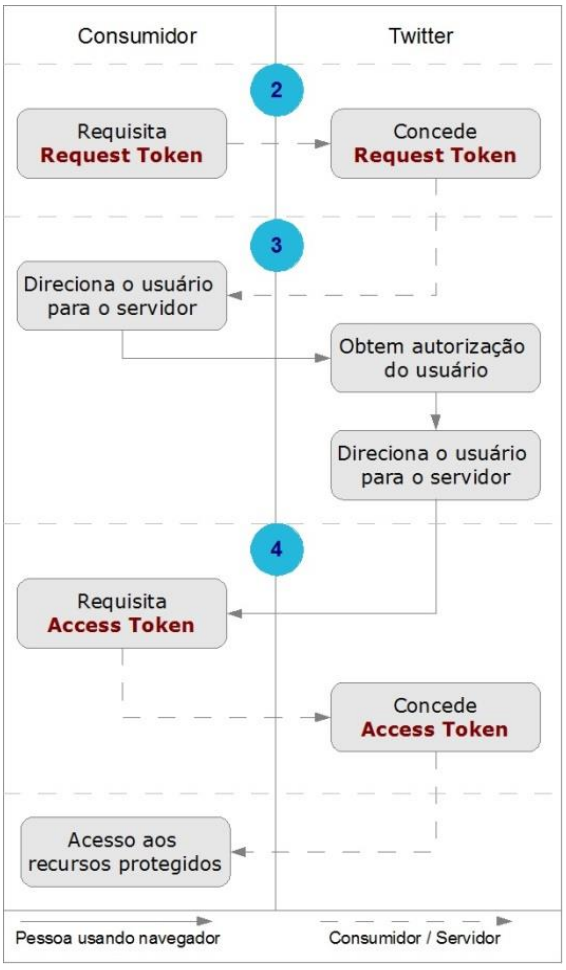



Fonte: *Twitter* (2016).

O acesso aos *Tweets* por meio da *REST API* tem uma desvantagem em relação ao *Streaming API*. A *REST API* tem um limite de, no máximo, de 450 requisições em um intervalo de tempo de 15 minutos. Apesar dessa desvantagem, a *REST API*, recebe todos os resultados buscados em uma única requisição, o que acaba sendo vantajoso.

Para se ter acesso as *API's* do *Twitter* é necessária uma autenticação precedida de autorização por parte do usuário. Para obter acesso as chaves de autenticação, é preciso registrar um aplicativo no site do *Twitter*, o que fornece um conjunto de chaves e *tokens*, que são usados no *script* da aplicação. Esta autenticação e autorização são realizadas pelo protocolo *OAuth*. O *OAuth* é um padrão aberto que permite aos usuários de um *site* garantirem acesso a uma aplicação externa aos seus recursos privados sem a necessidade de compartilhar suas senhas e *logins*. Para isso, o protocolo *OAuth* estipula um fluxo de comunicação entre a aplicação solicitante e a aplicação no caso o *Twitter* (*TWITTER*, 2016d). A Figura 4 detalha o fluxo de comunicação *OAuth* para obtenção do *token.* No fluxo de comunicação, o consumidor requisita o *token* aos servidores do *Twitter* que certifica a integridade do usuário, antes de conceder o *token,* e o acesso aos recursos protegidos.

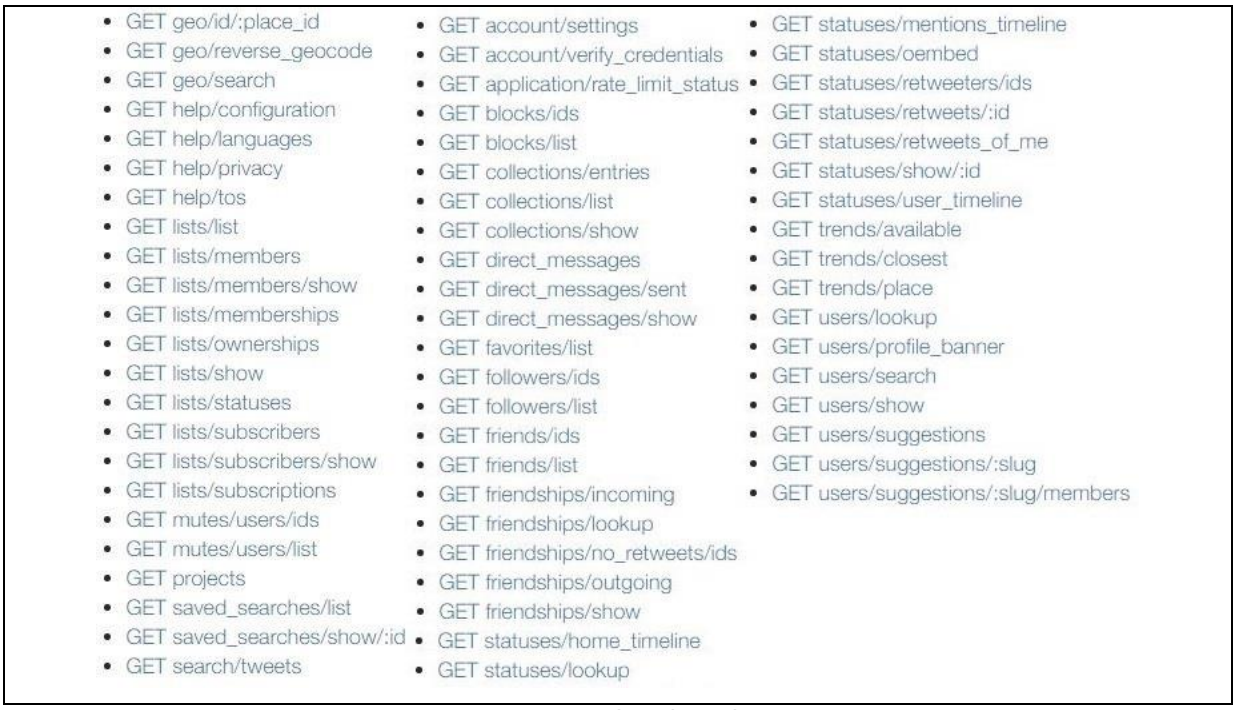
Figura 4: Fluxo *OAuth* para obtenção do *token* de acesso à *API* do *Twitter*.




Fonte: *Twitter* (2016).

Uma vez que é realizado o processo de autenticação mostrado na Figura 4, o acesso as *API's* do *Twitter* é liberado e os dados poderão ser intercambiados entre o *Twitter* e a aplicação. A *API* do *Twitter* permite diferentes tipos de solicitações para intercambiar os dados entre a aplicação consumidora e o *Twitter*, são os métodos: *GET* e *POST* (MAKICE, 2009). Os métodos *GET* são utilizados para recuperar diferentes tipos de dados do servidor, como descrito pela Figura 5.

Figura 5: Métodos *GET* da *API* do *Twitter*.




Fonte: *Twitter* (2016).

Os métodos *POST* são necessários para fazem alterações nos servidores do *Twitter* em vez de apenas recuperar os dados (MAKICE, 2009). Os métodos *POST* para a realização de tais alterações podem ser visualizados na Figura 6.

Figura 6: Métodos *POST* da API do *Twitter*.


Fonte: *Twitter* (2016).

Após a autenticação *OAuth,* os métodos *GET* e *POST* permitem consultar e manipular diversos conteúdo nos servidores do *Twitter*. As respostas das solicitações são trocadas no formato *JavaScript Object Notation* (*JSON*) que é um formato para intercâmbio de dados considerado simples tanto para a leitura e escrita humana, quanto para a realização das mesmas coisas de maneira computacional. O formato *JSON* é definido em formato texto, e está disponível para uso em várias linguagens de programação (JSON, 2017).

## 2.5 Visualizações dos dados

A visualização de dados é a representação dos mesmos em um formato gráfico. O objetivo da visualização é promover a compreensão dos dados, comunicar as ideias importantes utilizando da simplificação dos valores dos mesmos.

A visualização dá a capacidade de usar os dados de forma intuitiva, sem conhecimentos técnicos aprofundados. Mesmo usuários iniciantes podem criar visualizações de dados que sejam significativas, como gráficos de pizza, gráficos de linha, gráficos de bolhas e mapas de calor.

Segundo Schneider (2016), a apresentação das informações deve possuir uma maior dinamicidade, pois ela torna possível que a mesma informação seja mostrada com maior ou menor grau de granularidade sem a necessidade de carregamento de outros

gráficos. A dinamicidade dos gráficos permite a apresentação de gráficos de uma maneira mais clara e com maior número de informações. Esta dinamicidade tem por característica permitir que determinados dados sejam mostrados na tela, apenas se o usuário posicionar o ponteiro do mouse em determinadas áreas do gráfico.

Segundo Zuk (2006), com o uso da heurística de informação visual é são possíveis encontrar a maioria dos problemas relacionados à usabilidade com relação à visualização de dados. Esta heurística leva em consideração os seguintes fatores: percepção, cognição e usabilidade.

Quando se trata da representação dos dados, Santos *et. al.* (2009) apresenta uma abordagem onde explica que cada dado em uma base pode ser facilmente visualizado se tivermos duas ou três dimensões. Segundo Santos *et. al.* (2009), dados semelhantes devem aparecer geometricamente próximos no espaço de atributos, e a distância calculada neste espaço entre dois pontos é usada por várias técnicas de mineração de dados para representar semelhança e diferença entre os dados correspondentes.

## 2.6 Considerações finais

Este capítulo abordou os conceitos da descoberta de conhecimento com base em dados, descoberta de conhecimento com base em texto e conceitos ligados à mineração de dados e a mineração *Web* seja ela em rede social ou não.

No próximo capítulo será apresentada a metodologia, bem como todos os processos e recursos utilizados no desenvolvimento da ferramenta.

# 3 METODOLOGIA E DESENVOLVIMENTO DA FERRAMENTA

Neste capitulo será abordado à metodologia de pesquisa e a escolha dos mecanismos para o desenvolvimento da ferramenta, fornecendo uma contextualização detalhada sobre cada tópico abordado.

Na Seção 3.1 será apresentado o método de pesquisa e referencial teórico deste trabalho. Conceitos relacionados ao desenvolvimento da ferramenta serão apresentados na Seção 3.2.

## 3.1 A pesquisa

O presente trabalho é caracterizado como uma pesquisa aplicada, pois o mesmo utiliza-se de mecanismos e conhecimentos existentes para resolver problemas atuais. No trabalho são utilizadas técnicas de engenharia de *software*, *data mining*, *web mining* para desenvolver a aplicação com a finalidade de realizar a extração, a mineração de dados e a representação dos dados oriundos do *Twitter*.

O referencial teórico deste trabalho se baseia em: monografias, livros, artigos científicos e alguns sites de conteúdos técnicos, tais como: as documentações do *Twitter*, *D3-Data-Driven Documents* e *Google Charts*.

## 3.2 Desenvolvimento da ferramenta

Esta seção apresentará uma visão detalhada sobre as linguagens, bibliotecas, ferramentas gráficas utilizadas para o desenvolvimento da ferramenta e de suas funções como citadas nos objetivos deste trabalho.

### 3.2.1 Linguagem de programação, bibliotecas e *frameworks*

O primeiro passo para a construção da ferramenta se deu pela decisão sobre qual a linguagem de programação a ser utilizada, e, para tal escolha é relevante se considerar as linguagens suportadas pela *Twitter Libraries*. O *Twitter* fornece suporte a múltiplas linguagens, como: *C++, DotNet, Java, Objetive-C, Perl, PHP, Python, Ruby,* dentre outras. Embora as mesmas não sejam necessariamente construídas ou testadas pelo *Twitter*, elas fornecem suporte a atual *API* do *Twitter* (*TWITTER*, 2017).

Dentre as opções a ferramenta foi desenvolvida na linguagem *Ruby*, uma vez que a mesma possui a vantagem de ser uma linguagem de programação interpretada multiparadigma, de tipagem dinâmica e forte (*RUBY*, 2017). Outro ponto positivo na escolha desta linguagem é o fato da mesma possuir um uma grande diversidade de bibliotecas e aplicativos na linguagem *Ruby,* que podem facilmente se integrarem ao sistema, uma vez que eles são usados para distribuir funcionalidades reutilizáveis.

A linguagem *Ruby* permite estender ou modificar a funcionalidade de aplicações com o uso das *Ruby Gems* (*GEMS*, 2017). O trabalho utiliza algumas destas *Gems* para implementar algumas das funções propostas no objetivo específicos deste trabalho, tais como:

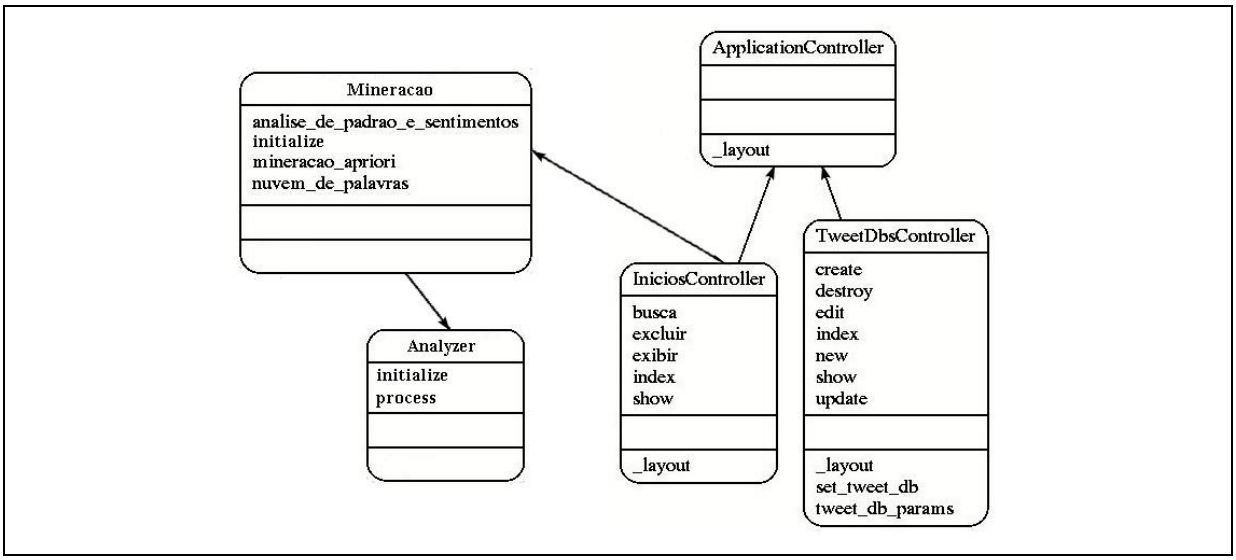
- Mongoid v5.1: é a *Gem* que realiza a comunicação da aplicação com o banco de dados *MongoDB*, que foi o banco de dados adotado neste trabalho.
- *Twitter* v5.16: estabelece a comunicação com as *API's* do *Twitter*.
- *Therubyracer* v0.12: realiza chamada de códigos *JavaScript* e manipula objetos *JavaScript* no *Ruby*.
- *Apriori-ruby* v0.0.9: é a *Gem* que conta com a implementação do algoritmo de mineração de dados *Apriori* na linguagem *Ruby*.
- *Data_mining* v0.0.6: é uma pequena coleção de vários algoritmos de mineração de dados. Os algoritmos compostos nessa *Gem* são: *Density Based Clustering*, *Apriori*, *PageRank*, *k-Nearest Neighbor Classifier.*
- *Sentimentalizer* v0.3: o algoritmo gera a análise de sentimento em inglês dos *Tweets* na linguagem *Ruby*, os textos são classificados como positivo, negativo ou neutro.

Além das *Gems* citadas acima, o trabalho conta com o auxílio de bibliotecas gráficas. O *Geo-Charts* é um componente do repositório de bibliotecas gráficas do *Google Charts*. O *D3.js* é uma biblioteca *JavaScript* para manipulação de documentos com base em dados e na criação de modelos gráficos. Outro *framework* utilizado no desenvolvimento do trabalho foi o *Ruby on Rails* v4.2, que é um *framework* que proporciona praticidade no desenvolvimento de aplicativos para *Web.* O motivo pelo qual o *Rails* foi escolhido é o fato do mesmo ser prático para reuso de código, além de trazer o benefício de possuir uma sintaxe simples, e de adotar o padrão arquitetural Modelo-Visão-Controlador (MVC) que tem por finalidade dividir a aplicação em componentes (*RAILS*, 2017).

### 3.2.2 Padrão arquitetural e diagramas

O trabalho seguiu o padrão arquitetural MVC, o qual se divide em três camadas: *Model* que é responsável pela leitura, escrita e validações dos dados; a *View,* que é responsável pela interação com o usuário; e por fim a *Controller* que interpreta os eventos que acontecem na *View* e manipula os dados que estão no *Model.* Nas Figuras 7 e 8, respectivamente, observar-se o diagrama de classe da camada *Controller* e a representação da persistência dos dados do pacote *Model* deste trabalho.

Figura 7: Diagrama de classe do pacote *Controller.*




Fonte: Elaborado pelo autor.

A Figura 7 ilustra as classes que compõem o pacote *Controller* com seus respectivos métodos. As explicações sobre as classes e os métodos presentes no pacote *Controller* encontram-se detalhada nas próximas seções.

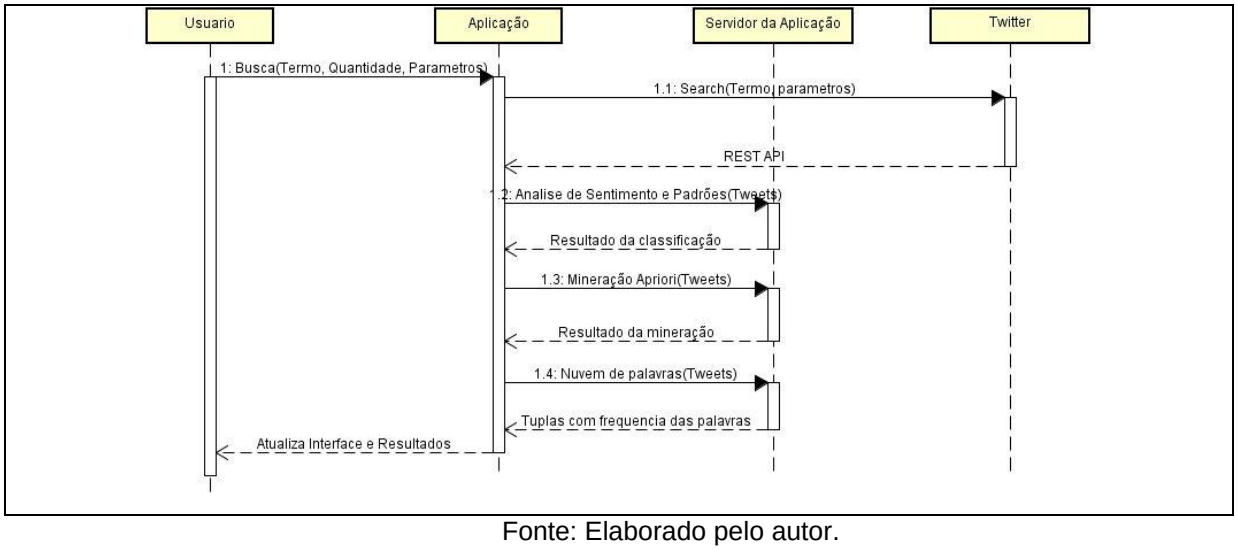
Figura 8: Representação da persistência dos dados do pacote *Model.*


Fonte: Elaborado pelo autor.

Os dados estão persistidos como ilustrado na Figura 8. Neste trabalho, os dados manipulados pelo *Model* encontram-se salvos por meio de um banco de dados não relacional, visto que o mesmo pode se adequar ao formato *JSON*, que por sinal é o formato das reposta de solicitação do *Twitter.*

A aplicação foi desenvolvida com base no diagrama de sequência apresentado na Figura 9.

Figura 9: Diagrama de sequência.




Fonte: Elaborado pelo autor.

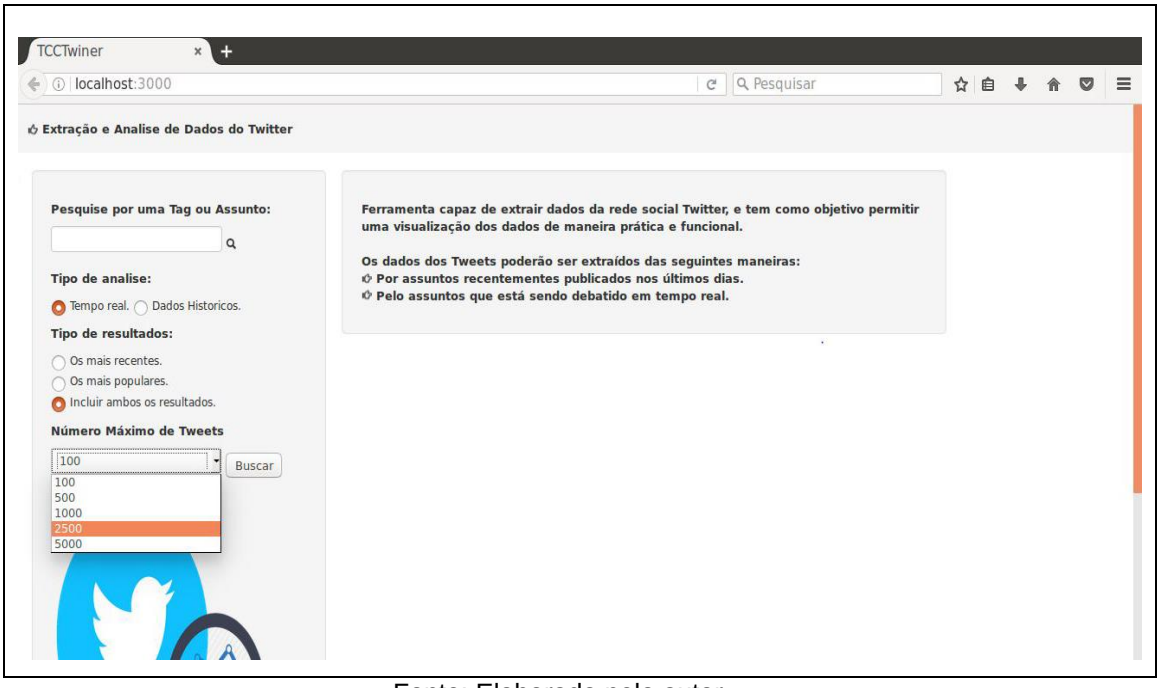
De acordo com a Figura 9, o processamento se inicia a partir do momento que o usuário insere os parâmetros para a busca. Os parâmetros de busca contêm: o termo a ser buscado, a quantidade de *Tweets* e o tipo de busca que será realizada, considerando os

seguintes aspectos: os *Tweets* mais populares, *Tweets* mais recentes e a mistura de ambos. A próxima ação é o processo de extração de dados no servidor do *Twitter*. Este processo utiliza o método *Search* com os parâmetros inseridos pelo usuário. O servidor do *Twitter* responde a solicitação na *REST API*, com documentos *JSON* contendo os *Tweets* sobre o termo pesquisado. Uma vez que a aplicação recebe a resposta do servidor do *Twitter*, ela armazena os dados e o direciona para os métodos de mineração para posteriormente se obter os resultados das técnicas aplicadas. Conforme os resultados forem retornados, os mesmos serão exibidos aos usuários.

### 3.2.3 Interface da aplicação

A página inicial permite que o usuário insira o termo de busca em um campo de texto e que o mesmo configure os demais parâmetros da busca, como ilustrado na Figura 10.

Figura 10: Página inicial da aplicação.




Fonte: Elaborado pelo autor.

A interface apresentada na Figura 10 ilustra os parâmetros de busca utilizados na aplicação, tal como:

- O termo de busca na aplicação permite a pesquisa sobre um termo específico ou alguma *#hashtag* associada a algum assunto.

- A opção tipo de análise permite ao usuário escolher sobre qual análise será aplicada aos dados: se será feita a extração e análise sobre os dados dos últimos sete dias ou dos dados que se encontram no banco de dados da aplicação.
- A opção tipo de resultado utiliza do parâmetro *result_type,* que é um parâmetro adicional da plataforma do *Twitter*. Esse parâmetro seleciona se o conjunto de resultados será representado por *Tweets* recentes ou populares, ou até mesmo, por uma combinação de ambos (*TWITTER*, 2016b).
- A opção o número máximo de *Tweets* determina um conjunto máximo de *Tweets* a serem extraídos. Existe este limite para que não seja gerada uma sobrecarga na aplicação, e diminua o tempo necessário para realizar a análise sobre os dados.

### 3.2.4 Processo de extração dos dados na aplicação

O processo de extração de dados no *Twitter* pode ser realizado com o auxílio das *API's REST* e *Stremming* como apresentado na Seção 2.4.1. No decorrer deste trabalho, a *API* escolhida para o desenvolvimento foi a *REST*, uma vez que a extração de dados pela mesma é realizada sobre *Tweets* já publicados e que se encontram nos bancos de dados do *Twitter.* A *REST API* também foi escolhida, porque ela não necessita de uma conexão ativa e persistente entre a aplicação cliente e o servidor do *Twitter*.

A extração de dados realizada pela ferramenta utiliza o método *Search* do *Twitter*. Este método faz parte da *REST API,* o que permite realizar consultas em relação aos *Tweets* recentes ou populares, e comporta-se de maneira semelhante às pesquisas realizadas no cliente *Web* do *Twitter* (*TWITTER*, 2016b). O Quadro 1 exemplifica os parâmetros no código utilizado para realizar a autenticação e a extração dos dados.

Quadro 1: Código de Autenticação e Extração.

```
client = Twitter::REST::Client.new do |config|
        config.consumer_key    = "API Key"
        config.consumer_secret = "API Secret"
        config.access_token        = "Access Token"
        config.access_token_secret = "Secret Token"
end
client.search(@termo , result_type: @tipo_de_resultado).take(@qnt_Tweets).each do |a|
        //Trecho de código para manipulação dos dados.
End
```

Fonte: Elaborado pelo autor.

O trecho de código acima realiza a autenticação da aplicação com o servidor do *Twitter* e posteriormente realiza a extração dos dados a partir dos parâmetros configurados pelos usuários. Os dados são recebidos como resposta no formato *JSON*, e possuem praticidade para a manipulação, sendo facilmente persistido para posterior análise.

### 3.2.5 Persistência dos dados na aplicação

Após a etapa de extração, é necessário o armazenamento dos dados para futuros processamentos e análises sobre os mesmos. Para a realização da persistência dos dados, a aplicação utilizará de um modelo de banco de dados não relacional, tendo em vista que desta maneira não será requerida a criação de tabelas, e os dados poderão ser salvos orientados a documentos, como exemplificado na Seção 3.2.2.

O banco de dados adotado na aplicação é o *MongoDB*, pois nele a representação dos documentos é em objetos similares as respostas do *Tweets.* O *MongoDB* persiste seus dados em objetos *Binary JavaScript Object Notation* (*BSON*), que são muito semelhantes aos objetos *JSON* (*MONGODB*, 2017).

### 3.2.6 Pré-processamento dos dados na aplicação

O pré-processamento se torna necessário devido uma grande quantidade de informações contidas nos dados que não são particularmente úteis para a análise de padrões. De acordo com Carvalho Filho (2014), na etapa de pré-processamento de dados textuais pode-se utilizar técnicas de Processamento de Linguagem Natural (PNL) que servem como remoção de segmentação de palavras, símbolos de pontuação, espaços em branco, números e palavras que são muito frequentes, como as *Stopword*. *Stopwords* são palavras muito comuns que aparecem no texto e carregam pouco significado servindo apenas com uma função sintática e não indicando nenhuma relevância ao assunto (EL-KHAIR, 2006). No Quadros 2 é possível observar os exemplos de *Stopwords*.

Quadro 2: Exemplos de *Stopwords.*

| | |
|---|---|
| An | aquela |
| and | cada |
| any | com |
| are | como |
| as | meus |
| at | mesmos |
| away | nestes |
| back | outras |
| be | qual |
| because | quando |
| do | quanto |
| does | que |
| each | quem |
| else | são |
| even | se |

Fonte: Elaborado pelo autor.

As palavras citadas no Quadro 2 são apenas uma fração do banco de dados de *Stopword* utilizadas neste trabalho. O banco de *Stopwords* foi construído com base na publicação de Almeida (2017), que cita quais são as palavras mais comuns das seguintes línguas: português, inglês, francês e espanhol. O uso de *Stopword* neste trabalho tem por finalidade remover palavras que não expressam algum sentimento ou polaridade, como artigos, preposições, dentre outros elementos gramaticais.

Vale ressaltar que neste trabalho, o pré-processamento é utilizado para padronizar os dados. Todas as palavras são transformadas para minúsculo, e caracteres especiais e números são removidos do corpo da mensagem, tendo em vista que os mesmos não carregam nenhum tipo de significado que seja útil no contexto. Isso é exemplificado no Quadro 3.

Quadro 3: Antes e depois da etapa de pré-processamento.

| | |
|---|---|
| @SergioMaLheiros Força aos capixabas e a todos os moradores do Espírito Santo! #ES #PrayForES | sergiomalheiros forca capixabas moradores espirito santo es prayfores |

Fonte: Elaborado pelo autor.

O Quadro 3 exemplifica o pré-processamento antes e depois sobre o corpo da mensagem.

### 3.2.7 Análise de padrões dos dados pela aplicação

Para a tarefa de análise de padrões, alguns *softwares* e bibliotecas foram estudados. Dentre elas, o *Weka* que é uma ferramenta desenvolvida em *Java*, e fornece implementações para os principais métodos de mineração de dados, tais como: regressão, classificação, *clustering* (agrupamento), mineração de regras de associação e seleção de atributos (WITTEN ET AL, 2011). A ferramenta fornece uma *API Java* que é flexível e permite a sua integração a qualquer tipo de sistema *Java* ou *Ruby*. Além da ferramenta citada acima, é possível encontrar bibliotecas de mineração no repositório de *RubyGems*. Uma delas é a *DataMining*, a qual implementa uma pequena coleção de algoritmos de mineração, como citado na Seção 3.2.1.

No contexto deste trabalho, a biblioteca que melhor se adequou ao desenvolvimento foi a *DataMining*, visto que a mesma é nativa na linguagem *Ruby*, e assim consegue processar os dados de maneira mais rápida e eficiente. Dentre a coleção de algoritmos do *DataMining,* o algoritmo de *Apriori* foi utilizado na aplicação para realizar as tarefas de associação. Este algoritmo é utilizado para encontrar associações relevantes entre os atributos dos *Tweets*. A associação leva em consideração a relação entre os atributos que constam: as localizações e os dispositivos, e também os dias das postagens e os idiomas. Segundo De Amo (2004), o algoritmo de *Apriori* utiliza da regra de associação para o problema denominado análise de cesta de mercado, que visa encontrar regularidades no

comportamento de compra dos clientes. O *Apriori* tenta encontrar conjuntos de produtos que são frequentemente comprados em conjunto, de modo que a partir da presença de certos produtos em um carrinho de compras pode-se inferir (com uma alta probabilidade) que certos outros produtos estão presentes. O Quadro 4 ilustra o algoritmo de *Apriori* utilizado no desenvolvimento desta ferramenta.

Quadro 4: Exemplo algoritmo de *Apriori.*

```
transações = [["cafe","pao","leite"], ["cafe","biscoito","leite"], ["biscoito", "pao","manteiga"], ["biscoito", "manteiga", "leite"]]
item_set = Apriori::ItemSet.new(transações)
suporte = 50
confiança = 60
item_set.mine(suporte, confiança)
```

```
#Resultado
=> {"cafe=>leite" = 100.0%, "leite=>cafe" = 66.6%, "leite=>biscoito" = 66.66%,
"biscoito=>leite" = 66.66%, "biscoito=>manteiga" = 66.66%, "manteiga=>biscoito" =
100.0%}
```

Fonte: *Apriori* Documentação (APRIORI, 2017).

O Quadro 4 exemplifica o do algoritmo *Apriori* e seu funcionamento. Segundo De Amo (2004), o algoritmo de *Apriori* pode ser interpretado como uma expressão da forma A → B, onde A e B são conjuntos de itens. Por exemplo, {pão, leite} → {café} é uma regra de associação. A ideia por trás desta regra é que pessoas que compram pão e leite têm a tendência de também comprar café. Isto é, se alguém compra pão e leite então também compra café. Repare que esta regra é diferente da regra {café} → {pão, leite}. O algoritmo possui três parâmetros configuráveis: a transação que contém os conjuntos de itens; o suporte que é a taxa de frequência dos itens dentro dos conjuntos; e a confiança. Este último é o que representa a porcentagem das transações que suportam um item em ‘B’ dentre todas as transações que também suportam um item em ‘A’.

Além do algoritmo *Apriori*, a aplicação adota também como base o uso de algoritmos de *hash* para realizar a classificação dos dados com base na sua frequência. Camilo e Silva (2009) alega que a etapa de classificação visa identificar a qual classe um determinado

registro pertence. Nesta tarefa, o modelo analisa o conjunto de registros fornecidos e verifica a existência da classe do registro. Em caso afirmativo, o algoritmo o classifica; caso contrário, cria-se uma nova classe para o mesmo. Assim, cada registro é indicado à qual classe pertence. O Quadro 5 apresenta como a classificação está sendo realiza na aplicação.

Quadro 5: Exemplo do algoritmo de *hash* de classificação.

```
localizacao = {"Pais" => "Quantidade de Tweets"}

country = informacao.pais

if(country.to_s.size > 0)
        if localizacao.has_key?(country)
                localizacao[country] = localizacao[country] + 1
        else
                localizacao[country] = 1
        end
end
```

Fonte: Elaborado pelo autor.

O exemplo de código ilustrado no Quadro 5 trata-se da maneira de como os dados vêm sendo classificados para posterior representação gráfica. O modelo de código acima foi utilizado em diversas outras situações na aplicação. O mesmo tem o intuito de realizar a classificação dos atributos dos *Tweets.* Desse modo, consegue-se identificar valores que são representativos para a criação dos gráficos, como as questões geográficas dos *Tweets*, questões idiomáticas e as *WordClouds*. *WordClouds* são interpretações visuais de coleções que são usadas para representar grandes conjuntos de informações. Já as *tags* geradas na *WordCloud*, são associações entre as palavras que são descritas de acordo com suas frequências (KUO *ET AL.*, 2007).

O último processo de análise de padrões adotado pela ferramenta é o processo da análise de sentimentos. Segundo Gonçalves *et al.* (2013) a análise de sentimentos pode ser utilizada para medir variação de humor do público em escala global. A aplicação da análise de sentimento tem o intuito de ajudar a compreender o comportamento dos usuários sobre determinado assunto, produto ou serviço. Santos (2010) explica que a análise de sentimento

é um tipo de classificação de textos que objetiva rotulá-los de acordo com o sentimento ou opinião contido nele. O algoritmo adotado pela ferramenta é o *Sentimentalizer*. O mesmo classifica os *Tweets* gerados na língua inglesa em três escalas: positiva, negativa e neutra. O Quadro 6 exemplifica o modo como o algoritmo classifica os *Tweets.*

Quadro 6: Exemplo da classificação do *Sentimentalizer.*

**#Sentimentalizer.analyze('message or tweet or status')**

analyzer = Analyzer.new

analyzer.process(**'I love ruby'**)

**=> {'text' => ' 'I love ruby', 'sentiment' => ' :) ' } #positivo**

analyzer.process(**'I like ruby'**)

**=> {'text' => 'I like ruby', 'sentiment' => ' :| ' } #neutro**

analyzer.process(**'I really like ruby'**)

**=> {'text' => 'I really like ruby', 'sentiment' => ' :) ' } #positivo**

analyzer.process(**'I hate ruby'**)

**=> {'text' => 'I hate ruby', 'sentiment' => ' :( ' } #negativo**

Fonte: Documentação *Sentimentalizer* (*SENTIMENTALIZER*, 2017).

O exemplo do Quadro 6 ilustra como o funcionamento do *Sentimentalizer*. Este algoritmo funciona da seguinte maneira: as sentenças são divididas em *tokens* e os mesmos recebem a atribuição de uma pontuação numérica para o seu sentimento médio. A pontuação total é então usada para determinar o sentimento geral. Por exemplo, o limite padrão é “0”, e se uma frase tiver uma pontuação de “0” a mesma é considerada "neutra". Porém, se for mais alto que o limite a mesma é "positiva”. Caso contrário, ela é "negativa".

### 3.2.8 Representação das informações

Para representação da informação, a aplicação utiliza-se de duas distintas bibliotecas gráficas: a *D3.js* e a *Google Charts* . A representação da *WordCloud* é realizada com o auxílio da biblioteca *D3.js,* que é uma biblioteca *JavaScript* para manipular documentos com base em dados. A *D3.js* é uma biblioteca *JavaScript* que permite criar modelos gráficos usando uma abordagem mais simples orientada por dados, alavancando padrões existentes da *Web* (ZHU, 2013).

O uso da *Google Charts* na aplicação se deu para a representação dos demais gráficos, que são expostos como classes de *JavaScript*. Estes gráficos podem ser personalizados para se adequarem à aparência necessária para representação dos dados. Os gráficos são altamente interativos e expõem eventos que permitem conectá-los para criar painéis complexos ou outras experiências integradas a *Web*. Os gráficos são renderizados usando a tecnologia *HTML5* e *Scalable Vector Graphics* (*SVG)* para fornecer compatibilidade entre navegadores e a portabilidade de plataformas móvel (CHARTS, 2017).

## 3.3 Considerações finais

Neste capítulo foram apresentados os passos e mecanismos utilizados durante o desenvolvimento da ferramenta proposta. Desta maneira uma vez que implementada a ferramenta a mesma necessita passar pelo processo de testes para comprovar a sua funcionalidade.  O capítulo 4 abordará os processos de testes e avaliações da ferramenta.

A ferramenta desenvolvida neste trabalho se encontra disponível sobre licença gratuita no *GitHub*, e pode ser consultada partir do seguinte endereço: <https://github.com/moreiramsilva/TCC_2_Ferramenta_Extracao_Dados_Twitter/>.

# 4 TESTE E RESULTADOS

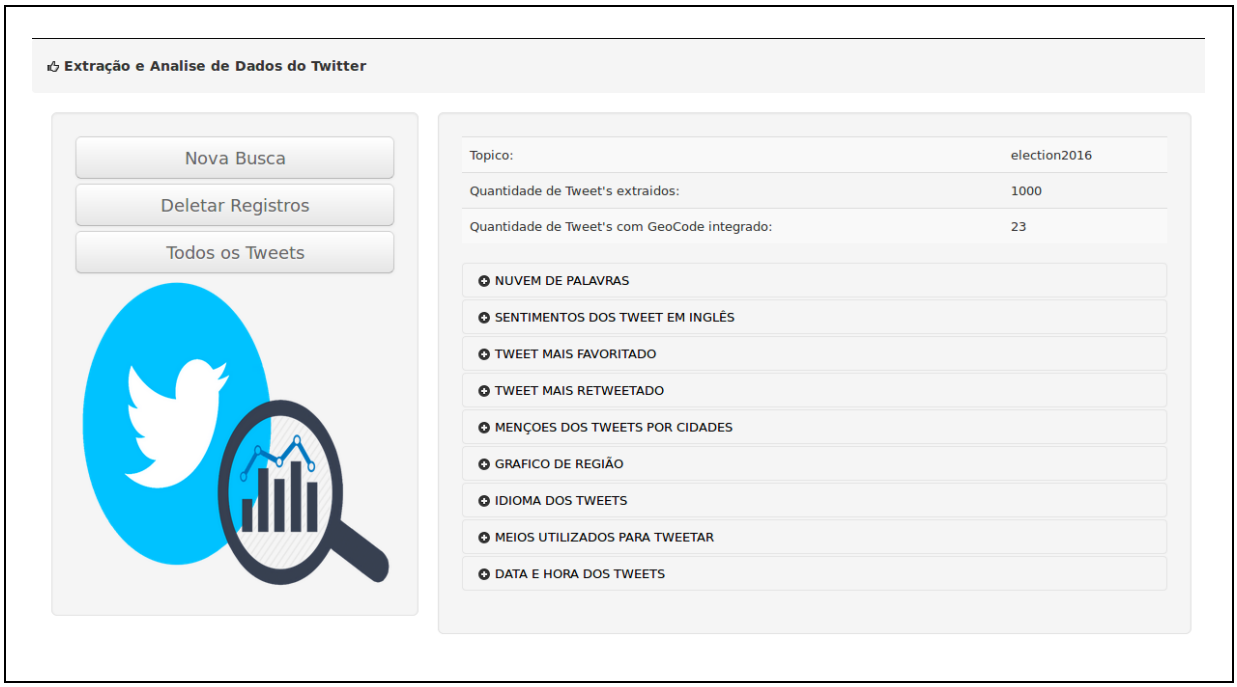
Neste capítulo são apresentados os testes sobre a ferramenta desenvolvida e a análise dos resultados provenientes dos dados da rede social *Twitter*. Os testes realizados para exemplificar a funcionalidade da ferramenta foram feitos sobre dois distintos temas da atualidade: Eleição Presidencial Americana e a Greve dos Policiais Militares no Estado do Espírito Santo, a fim de garantir uma maior abrangência nos resultados. A máquina virtual utilizada para a realização dos testes possui a seguinte configuração: Sistema Operacional *Linux Ubuntu* 16.04, 4GB de memória *RAM* e processador 5ª geração *Intel Core* i5.

É importante ressaltar que os resultados podem ser limitados por recursos computacionais, e que sempre devem ser observados apenas como um indicador e não como verdade absoluta sobre os temas.

## 4.1 *#election2016*

O primeiro tema pesquisado foi a Eleição Presidencial Americana que ocorreu na terça-feira, 8 de novembro de 2016. Cerca de 120 milhões de norte-americanos foram às urnas para decidir quem seria o 45º presidente dos Estados Unidos (Neves, 2017). A pesquisa sobre o tema analisou os mais recentes 1000 *Tweets* sobre a *tag #election2016* com a finalidade de descobrir a existência de algum padrão entre as informações contidas nos mesmos. A Figura 11 apresenta a interface dos resultados na ferramenta.

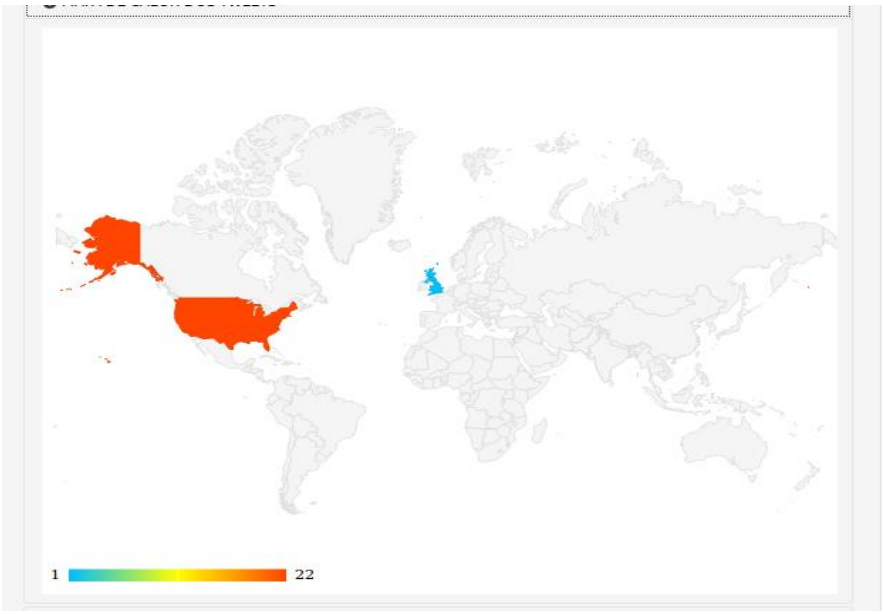
Figura 11: Interface dos resultados.




Fonte: Elaborado pelo autor.

A Figura 11 ilustra as informações pertinentes à busca, além de mostrar as demais análises realizadas pela ferramenta. A busca pela *tag #election2016,* resultou em apenas 23 *Tweets* com as coordenadas geográficas integradas, permitindo assim, se saber a real localização das postagens. Uma vez que as coordenas geográficas dos *Tweets* for conhecida*,* é possível gerar uma representação gráfica que identifique os países e as cidades de onde os *Tweets* foram postados, como ilustrado nas Figuras 12 e 13 respectivamente.

Figura 12: Gráfico de incidência por região sobre as postagens dos *Tweets* a respeito da Eleição Presidencial Americana.




Fonte: Elaborado pelo autor.

A Figura 12 mostram as localidades dos *Tweets* postados em nível de nação. A mesma ilustra a incidência dos *Tweets* com cores quentes e frias, variando a coloração devido à intensidade: sendo azul (baixa intensidade) e vermelho (alta intensidade). No exemplo acima ocorre à incidência de 22 *Tweets* provenientes dos Estados Unidos da América, e apenas um da Inglaterra.

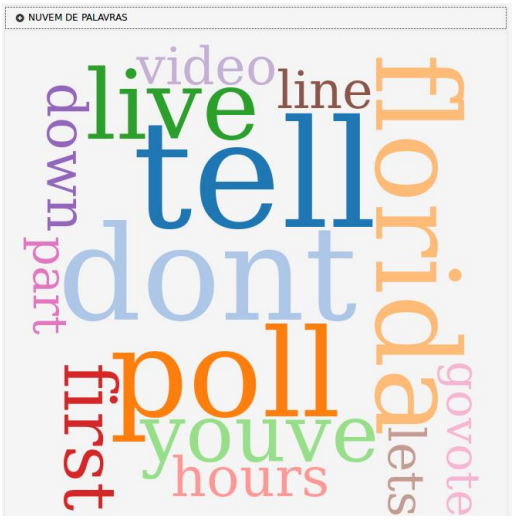
Figura 13: Menções dos *Tweets* por cidades.


Fonte: Elaborado pelo autor.

A Figura 13 apresenta as localidades dos *Tweets* de maneira acurada, é possível observar a incidência dos mesmos em nível das cidades.

A aplicação também trás outras análises como a *WordCloud*. Como descrito na Seção 3.2.7, a *WordCloud* é elaborada com base na frequência das palavras no conjunto de *Tweets*. A Figura 14 ilustra a *WordCloud* gerada sobre os *Tweets* ligados a *tag* *#election2016*.

Figura 14: *WordCloud* eleições.

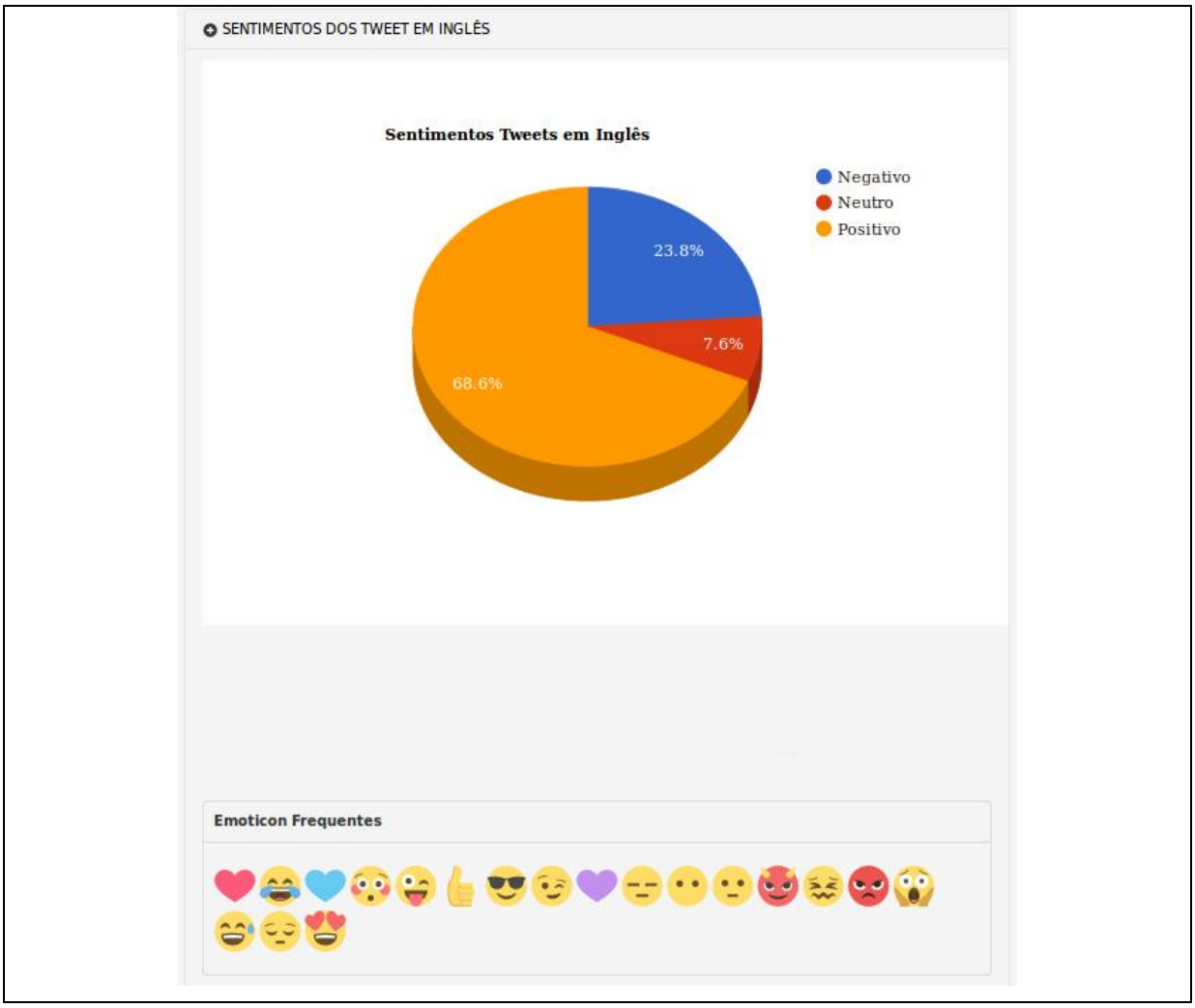



Fonte: Elaborado pelo autor.

Quando foram analisados os termos mais citados dentre os *Tweets*, percebem-se quais são as palavras que eventualmente incentivam os eleitores a irem votar, tais como: *you've*, *govote, let's.* Dentre os outros termos frequentes, se tem o estado da *Florida*, a palavra *poll* que se refere às eleições e termos de negação como, por exemplo, *don't.* O uso dessa técnica de *Wordcloud* permite maior conhecimento dos termos que se refletem nos vocabulários dos eleitores ao publicar *Tweets* sobre o assunto.

Além da *WordCloud,* a aplicação apresenta a análise de sentimentos de *Tweets* na língua inglesa, como ilustrado na Figura 15. Esta análise tem por finalidade categorizar os *Tweets* da língua inglesa em três distintas categorias: positivos, negativos e os neutros. Esta técnica busca identificar o sentimento do usuário pela combinação das palavras usadas. Ademais, a ferramenta expõe os *emoticons* mais frequentes na análise dos textos.

Figura 15: Análise de sentimentos eleições.




Fonte: Elaborado pelo autor.

O gráfico da Figura 15 ilustra a porcentagem de *Tweets* em cada uma das categorias de sentimentos. Observa-se que, pela análise da ferramenta, 68,6% dos *Tweets* foram classificados como positivo, 23,8% foram classificados como negativo, e os 7,6% restantes foram definidos como neutros. Ao observar a frequência dos *emoticons,* é possivel observar também uma maior quantidade de *emoticons* com uma conotação positiva. Porém, nem todos os *Tweets* expressam opiniões positivas ou negativas. Alguns deles apresentam apenas fatos, comentários ambíguos, elementos multimídia que não cabem em nenhuma das duas opções e são removidos na etapa de pré-processamento dos dados.

A ferramenta também permite ter uma visualização dos *Tweets* com o maior número de favoritos e de *Retweets*. Carvalho Filho (2014) apresenta que, uma vez que os usuários compartilham de ideias similares, os mesmos tendem a “retuitar” e “favoritar” os *Tweets* de alguém, dando assim, a ênfase de que o usuário está concordando com a opinião de quem postou o *Tweet,* apresentando a influência do usuário sobre os demais. A Figura 16 e 17 respectivamente ilustra os *Tweets* mais “Retuitado” e o mais “Favoritado” dentre os analisados.

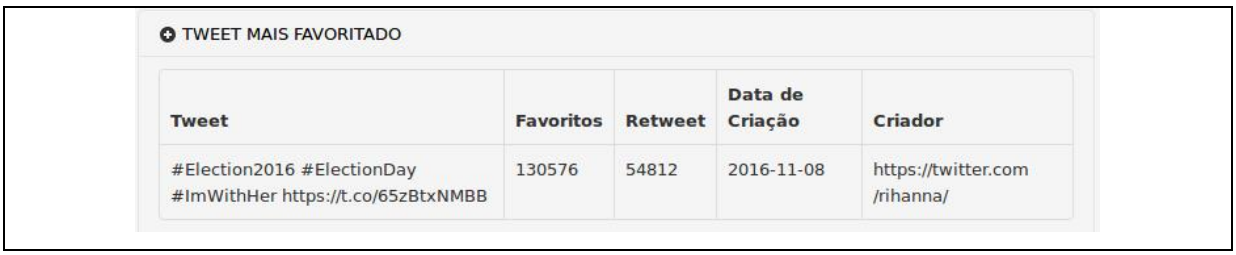
Figura 16: Maior número de *Retweets.*


TWEET MAIS RETWEETADO

| Tweet | Favoritos | Retweet | Data de Criação | Criador |
|---|---|---|---|---|
| #Election2016 #ElectionDay #ImWithHer https://t.co/65zBtxNMBB | 130576 | 54812 | 2016-11-08 | https://twitter.com /rihanna/ |

Fonte: Elaborado pelo autor.

Figura 17: Maior número de Favoritos.


TWEET MAIS FAVORITADO

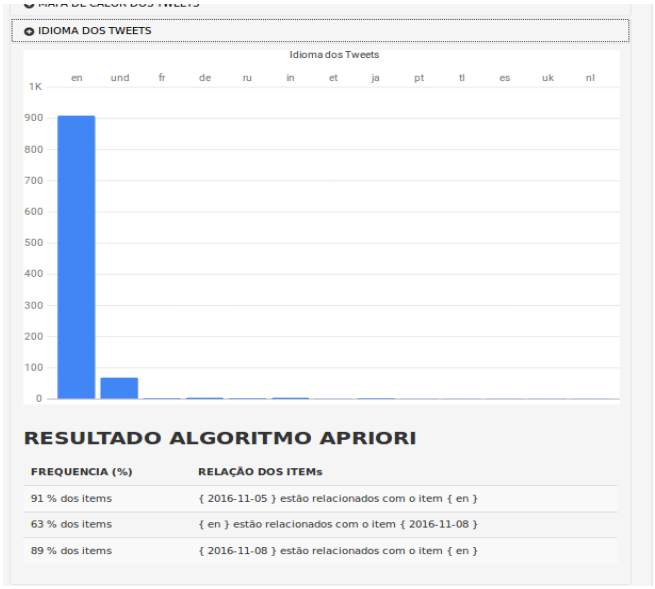
| Tweet | Favoritos | Retweet | Data de Criação | Criador |
|---|---|---|---|---|
| #Election2016 #ElectionDay #ImWithHer https://t.co/65zBtxNMBB | 130576 | 54812 | 2016-11-08 | https://twitter.com /rihanna/ |

Fonte: Elaborado pelo autor.

Vale ressaltar que na análise em questão os mesmos *Tweest* foram apresentados. Entretanto, uma abordagem sobre outro tema pode gerar *Tweets* diferenciados. Os *Tweets* exemplificados nas Figuras 16 e 17, respectivamente, ilustram a influência da cantora Rihanna sobre seus seguidores que compartilharam do mesmo sentimento ao apoiar a candidata Hilary, como descrito na *tag* #*I'mWithHer*.

As demais análises realizadas pela aplicação têm por finalidade agrupar os atributos dos *Tweets* e gerar gráficos quantitativos sobre cada atributo distinto. O gráfico criado tem por finalidade facilitar a compreensão dos dados e trazer uma informação sobre a frequência de cada atributo. A Figura 18 ilustra a quantidade de *Tweets* sobre cada idioma e apresenta o resultado da análise pelo algoritmo *Apriori* sobre os parâmetros correspondentes às datas de postagens e os idiomas dos *Tweets.*

Figura 18: Quantidade de *Tweets* por idioma.

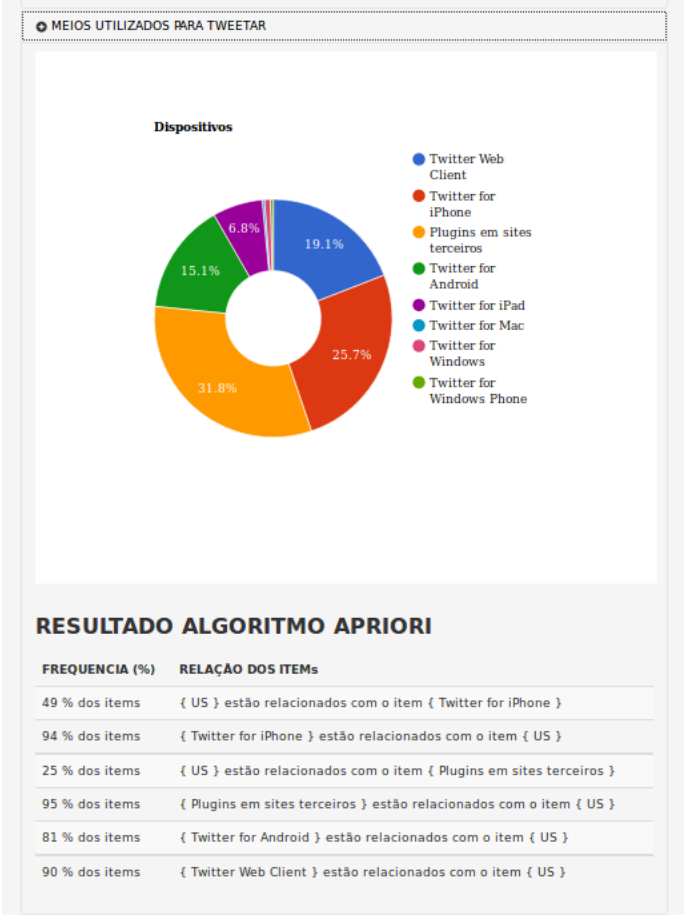



Fonte: Elaborado pelo autor.

Observa-se na Figura 18 que dos *Tweets* extraídos, pouco mais de 900 foram da língua inglesa e os demais estão divididos nos idiomas: francês, russo, japonês, português, espanhol, dentre outros. O resultado do algoritmo *Apriori* apresenta que 63% dos *Tweets* da língua inglesa sobre o tema abordado foram postados no dia 08 de novembro de 2016.

A aplicação também leva em consideração os meios utilizados para a criação dos *Tweets* sobre as eleições Americanas. A Figura 19 ilustra os principais meios encontrados para se interagir com esta rede social e suas respectivas frequências de uso.

Figura 19: Dispositivos e frequência de uso dos mesmos.

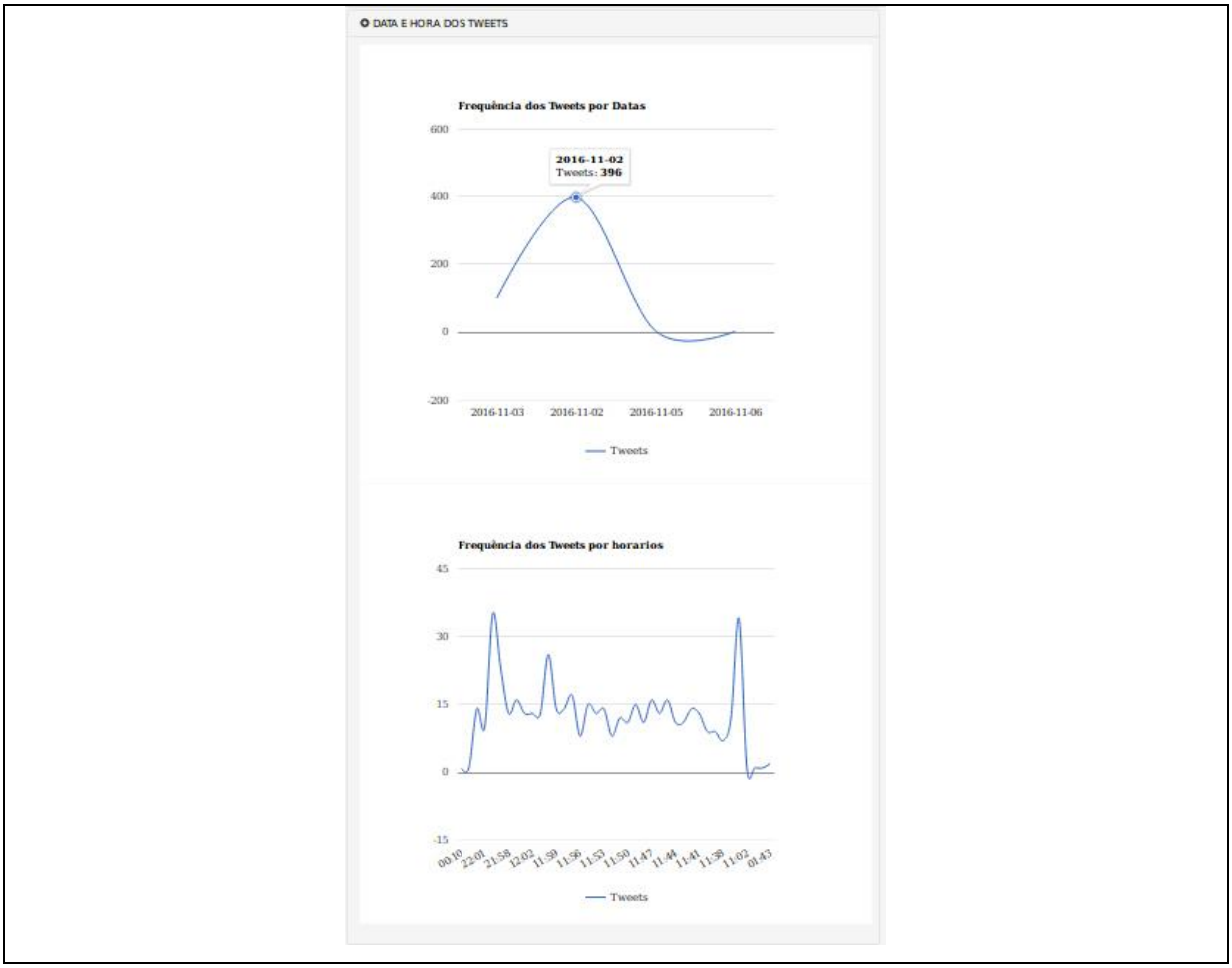



Fonte: Elaborado pelo autor.

A Figura 19 ilustra que o meio mais utilizado para a criação dos *Tweets,* foi por aplicações de terceiros, que apresentou uma taxa de 31,8% de todos os *Tweets*; seguido pelo *iPhone* com 25,7%, o cliente *Web* do *Twitter* com uma taxa de 19,1%. O *Android* tem uma taxa de 15,1%, seguido por porcentagens menores nos demais dispositivos. O resultado do algoritmo *Apriori* apresentou a relação do *itemset* “US” referente aos *Tweets* criados nos Estados Unidos. Verifica-se que 49% dos *Tweets* foram provenientes dos Estados Unidos da América e realizados via *iPhone* enquanto 25% dos *Tweets* Americanos foram realizados por aplicações de terceiros.

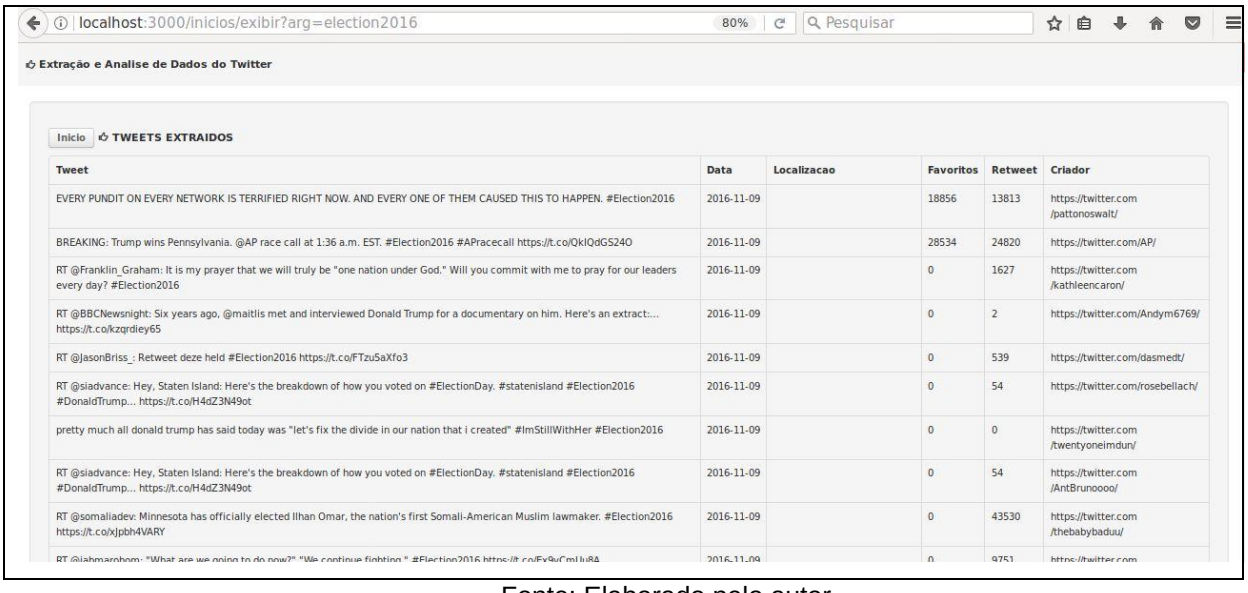
Por fim a aplicação apresenta a frequência de *Tweets* por dias e horários, como podem ser visto na Figura 20, e também permite com que o usuário tenha acesso a todos os *Tweets* que foram analisados como exemplificado na Figura 21.

Figura 20: Frequência de *Tweets* por tempo.




Fonte: Elaborado pelo autor.

Figura 21: Amostra de *Tweets* que foram analisados.

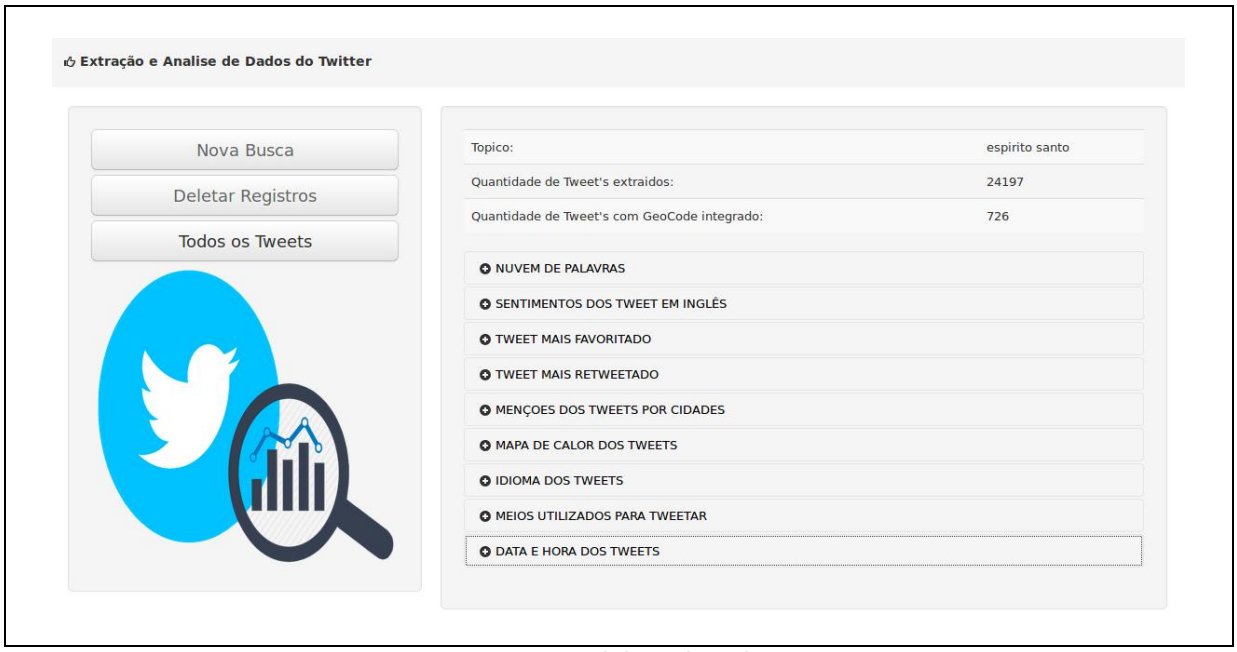



Fonte: Elaborado pelo autor.

## 4.2 #espiritosanto

O segundo tema abordado nos testes foi relativo à greve de policiais militares no estado do Espírito Santo. Como explanado por Do Carmo (2017), devido ao índice de violência durante a greve dos policiais militares, os moradores relataram o medo de andar nas ruas das cidades capixabas, o que fez com que os comércios ficassem fechados com receio de serem saqueados. O assunto ganhou repercussão nas redes sociais, e a análise sobre o mesmo teve o intuito de buscar conhecimentos sobre o impacto do assunto. A Figura 22 ilustra o resultado da extração dos dados sobre o tópico abordado.

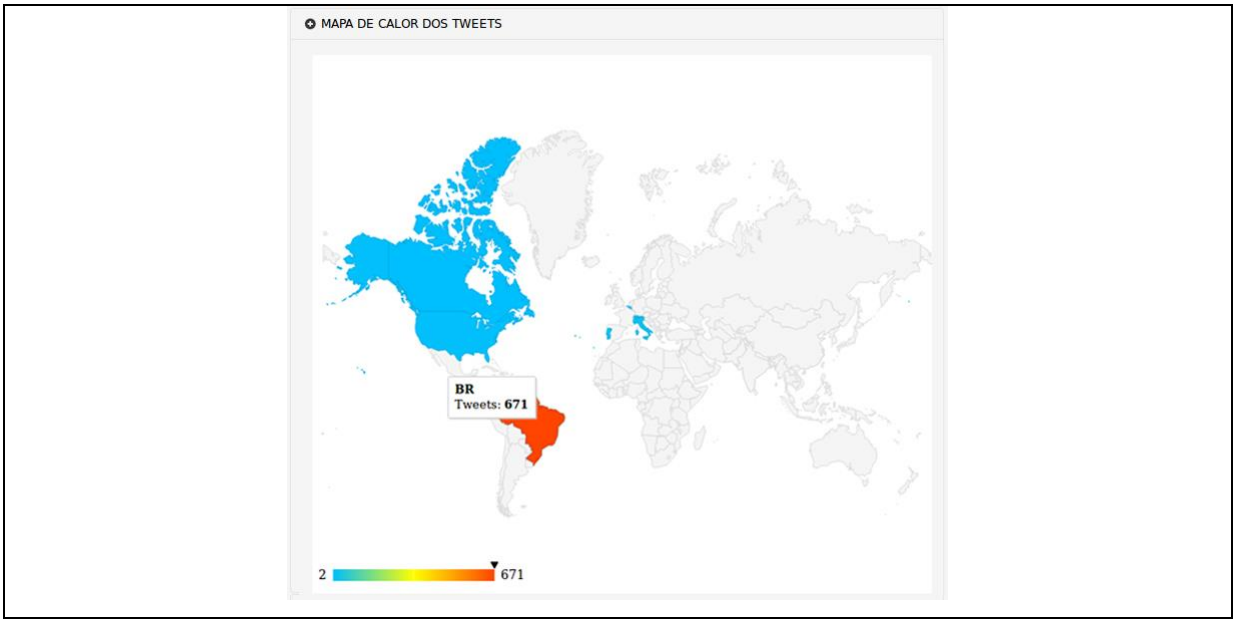
Figura 22: Resultado da extração #EspiritoSanto.




Fonte: Elaborado pelo autor.

Para a análise, foram extraídos os mais populares *Tweets*, onde se obteve o total de 24.197 *Tweets* sobre a *tag* #EspiritoSanto. Do total de *Tweets* extraídos, 726 possuíam coordenadas geográficas integradas, sendo assim ilustrando em quais regiões o assunto estava sendo debatido pela população. A Figura 23 e 24 ilustram os locais onde o assunto foi debatido.

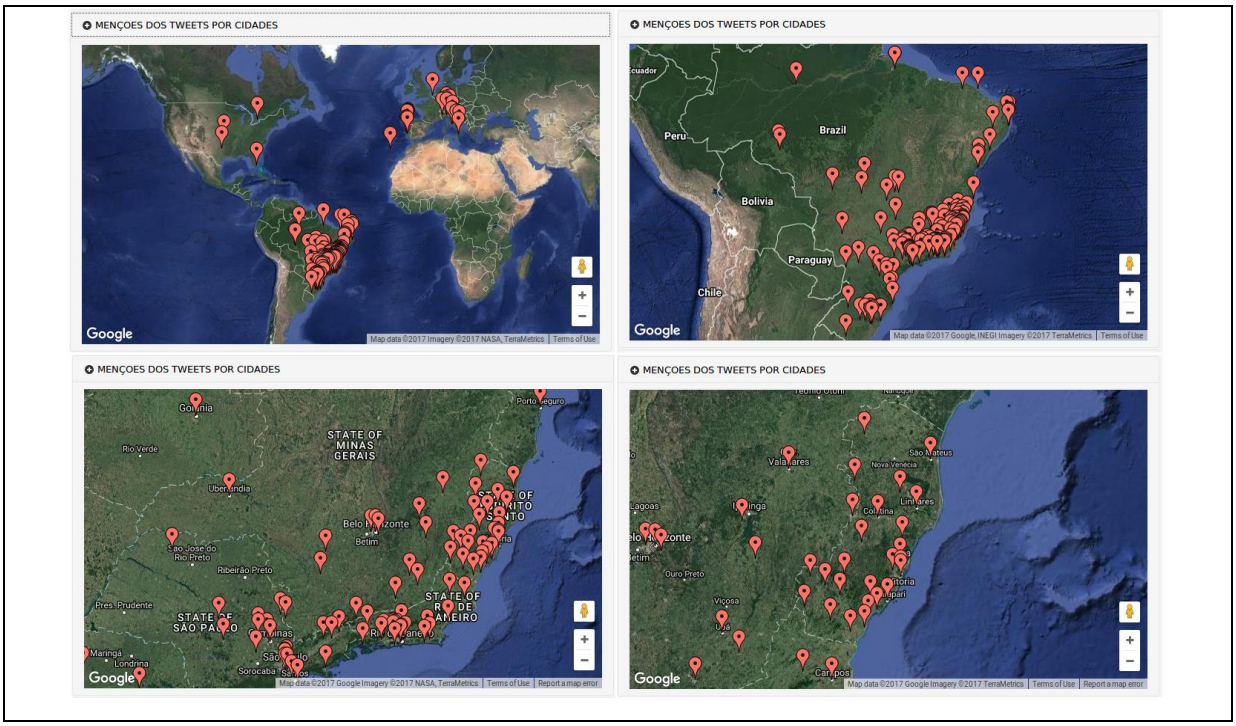
Figura 23: Mapa de Região sobre tópico Espírito Santo.




Fonte: Elaborado pelo autor.

Como ilustrado na Figura 23, percebe-se que o tópico teve abordagem internacional, sendo debatido em países como Itália, Portugal, Estados Unidos e Canadá. Como observado, o maior índice foi obtido no Brasil com a frequência de 671 *Tweets*.

Figura 24: Menções dos *Tweets* por cidades sobre tópico Espírito Santo.




Fonte: Elaborado pelo autor.

A Figura 24 representa as localidades de um modo mais ampliado até os níveis mais detalhados (em níveis de cidades). No que tange a análise de texto sobre o tópico abordado, a Figura 25 ilustra a *WordCloud* gerada.

Figura 25: *WordCloud* sobre tópico Espírito Santo.

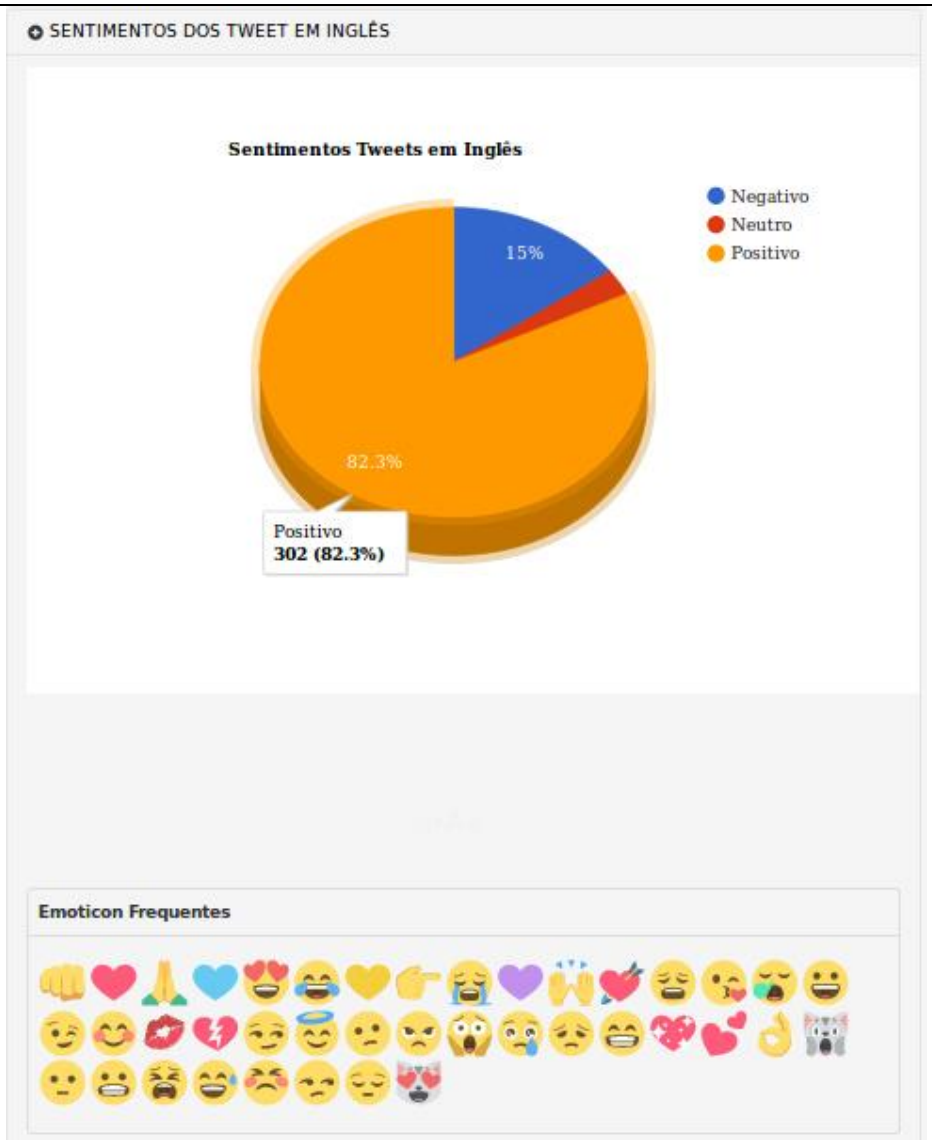

Fonte: Elaborado pelo autor.

Quando analisado os termos citados, os mesmos sugerem conotações de solidariedade à população, como: Amor, Deus, Orei, Orem, Abençoado, Coração, Abençoado. Porém, também nota-se outras palavras que tem significados cruciais para o contexto, como: Segurança, Invade, Crise. As palavras obtidas, de certo maneira, sugerem a opinião das pessoas sobre o ocorrido no Espírito Santo. Vale ressaltar que, a linguagem utilizada na *Internet* possui abreviações e gírias, e possuem muitos erros de digitação que dificultam a identificação dos termos.

Além da opinião dos usuários do *Twitter* sobre a crise no Espírito Santo, é possível realizar a análise dos sentimentos sobre o *Tweets* gerados na língua inglesa. Deste modo a

Figura 26 apresenta os 367 *Tweets* publicados na língua inglesa com a polaridade dos mesmos.

Figura 26: Análise de sentimento sobre tópico Espírito Santo.




Fonte: Elaborado pelo autor.

Observa-se pela análise da ferramenta que 82,3% dos *Tweets* foram classificados como positivo, 15% foram classificados como negativo, e os 2,7% restantes foram definidos como neutros. Embora o assunto se trate de uma crise, a maioria dos *Tweets* tem caráter positivo, isso pode sugerir que os usuários estavam demonstrando solidariedade e compaixão para com os capixabas. Uma vez que, se comparado com os termos da *WordCloud* palavras com conotações de solidariedade tiveram uma alta frequência.

## 4.3 Avaliações da ferramenta

É válido discorrer que pelos exemplos evidenciados nas Seções 4.1 e 4.2 a ferramenta criada pode ser dita funcional. Isso se justifica pelo fato de que: a mesma extrai os dados do servidor do *Twitter*, persistem os mesmos no banco de dados, e por meio de análises robustas gerar a informação com base na requisição solicitada pelo usuário.

A ferramenta possui algumas limitações, visto que apresentou-se instável quando solicitado um grande limite de *Tweets* para extração e análise. Em cenários que se possuem um alto número de *Tweets* a serem submetidos à análise de sentimentos, essa limitação é observada; o que por sua vez, pode ser justificado pela capacidade computacional da máquina que realizou os testes. Isso ocorre, porque à análise é custosa e acarreta em um alto consumo de memória, requerendo assim um grande poder de processamento e ocasionando em um grande tempo de espera por parte dos usuários.

Outro possível cenário de limitação da ferramenta é o fato de que ela não garante a extração da quantidade de *Tweets* pedida pelo usuário, uma vez que a conexão *REST API* tem suas limitações de requisições diárias. E, por fim, é importante observar que para o funcionamento correto da ferramenta deve-se avaliar a disponibilidade dos servidores do *Twitter* e também dos servidores do *Google Charts*, para o uso das bibliotecas gráficas *online.*

# 5 CONCLUSÕES

A mineração de dados em rede sociais e análise de sentimentos pode ser um fator crucial para que as pessoas e empresas possam medir os indicadores de influência sobre determinado termo ou assunto. Os conhecimentos sobre esses indicativos de influências podem ser utilizados para os mais variados fins, como os exemplificado na Seção 1.3.

As informações provenientes do *Twitter* têm escala global. A praticidade para a troca de informação faz com que os usuários tendem a se expressarem livremente por meio de #*hashtags*, criando grupos sobre opiniões em comuns Deste modo, fatos e acontecimentos locais podem repercutir entre os mais variados lugares e idiomas, tomando assim, proporções gigantescas.

Por mais que o próprio *Twitter* disponibilize formas para a extração dos seus dados, esse processo não é trivial. Isso ocorre porque o processo requer alguns fatores, bem como um extenso estudo sobre as *API's,* os tipos de dados que se são possíveis de extração, as linguagens suportadas pelo *Twitter library* e os limites de requisições de cada *API,* dentre outros.

Visto que o processo de extração não é trivial, deve-se considerar a complexidade dos processos de pré-processamento e análise. O pré-processamento sobre o texto não é realizado de uma maneira simples. Na internet há a liberdade de escrita e isso faz com que os usuários representem a mesma opinião de distintas maneiras, utilizado de gírias, abreviações, ou com erros gramaticais, tornando assim o processo de análise de texto mais complexo do que o esperado.

O processo de representação da informação também tem suas complexidades, pois cada tipo de gráfico gerado possui uma forma diferente de entrada de parâmetros. Por conseguinte, para a criação dos mesmos, as saídas dos algoritmos de análise exige uma remodelação de modo que se adeque as entradas de dados de cada tipo de gráfico em específico.

O desenvolvimento da ferramenta não é simples, por mais que exista uma extensa documentação sobre as *API's*. Durante o desenvolvimento inúmeros problemas foram encontrados, como, por exemplo: incompatibilidade de versões das bibliotecas *Rails* e do *MongoDb*, limitações de requisições sobre a *API REST,* e também o problema de

autenticação para uso da biblioteca gráfica do *Google Charts.* Mas, esses problemas foram contornados a partir de fóruns do *Twitter* e as documentações técnicas.

Com os exemplos analisados neste estudo, percebeu-se como um tema específico de uma região pode ser repercutido em níveis globais. Deste modo, foi possível identificar quais os sentimentos da população referente às suas postagens quando os *Tweets* se encontravam na língua inglesa, e qual a influência de personalidades famosas sobre seus seguidores. Diante destes fenômenos de repercussão e influência, a ferramenta desenvolvida tem grande potencial para estratégias de *marketing*, sendo que, a partir dos gráficos de incidência e análise de sentimentos, é possível desenvolver estratégias personalizadas para locais geográficos que de fato possam propiciar o aumento das vendas de determinado produto ou serviço. Outro benefício encontrado é que por meio destas análises é possível identificar a aceitação popular sobre determinado produto ou serviço, permitindo que, de certa forma, as empresas consigam um *feedback* mais rápido e em tempo real sobre a opinião de seus clientes e ou futuros clientes  em relação a empresa, e até mesmo, ter conhecimento sobre a opinião de dado local sobre algum assunto.

Embora existam outras ferramentas similares ofertadas no mercado, como as que foram citadas no Capitulo 1. A ferramenta criada, uma vez que seja disponibilizada gratuitamente online, permitirá que o próprio usuário faça as consultas e obtenha os resultados sem nenhum custo. Desta forma, tornando simplificado o processo de extração e análise de dados do *Twitter*.

No que tange aos ganhos pessoais, foi possível aprimorar os conhecimentos da linguagem *Ruby* e do *Framework Rail e* aprimorar os conhecimentos sobre bancos de dados não relacionais. O desenvolvimento do trabalho também permitiu aprofundar os conhecimentos sobre *Data mining* e as formas de recuperação e representação de dados e principalmente, permitiu um maior contato e conhecimentos sobre a rede social *Twitter* e suas *API's.*

Para futuros trabalhos, sugere-se que seja implementada a análise de sentimentos dos *Tweets* para outras línguas, além da inglesa. Outro ponto a ser aprimorado seria o desenvolvimento de técnicas para que a ferramenta possa também operar com a *Streaming API* e a *REST API* em paralelo*.* Outra melhoria poderia se dar por meio da disponibilidade de um servidor para hospedar a ferramenta visto que a mesma foi desenvolvida sobre um *framework* para aplicações *Web,* bem como a integração da ferramenta com outras redes sociais, por exemplo: *Facebook, Youtube,* dentre outros, o que tornará os resultados mais

abrangentes. E por fim, sugere-se a paralelização dos processamentos e análises dos dados, uma vez que o processo é custoso para uma máquina simples.

# REFERÊNCIAS


ALMEIDA, Rubens Queiroz. **As palavras mais comuns do inglês, português, espanhol, alemão e francês.** Disponível em: <http://www.dicas-l.com.br/arquivo/as_palavras_mais_comuns_do_ingles_portugues_espanhol_alemao_e_frances.php> Acesso em 26 fev. 2016.

*APRIORI*. **RubyDoc**. Disponível em: <http://www.rubydoc.info/gems/apriori-ruby/0.0.9>. Acesso em 19 fev. 2017.

BARION, Eliana Cristina Nogueira; LAGO, Decio. Mineração de textos. **Revista de Ciências Exatas e Tecnologia**, v. 3, n. 3, p. 123-140, 2015.

CABRAL, Mayara Kaynne Fragoso; BRUNO, João Carlos; CORADO, Vanessa Aires. **MINERAÇÃO DE DADOS DO *TWITTER* ATRAVÉS DO R**. 2015. Disponível em: <http://propi.ifto.edu.br/ocs/index.php/jice/6jice/paper/viewFile/6985/3359>. Acesso em: 14 dez. 2016.

CAMILO, Cássio Oliveira; SILVA, João Carlos da. Mineração de dados: Conceitos, tarefas, métodos e ferramentas. **Universidade Federal de Goiás (UFC)**, p. 1-29, 2009.

CARVALHO FILHO, José Adail. **Mineração de Textos:** Análise de Sentimento Utilizando *Tweets* Referentes À Copa Do Mundo 2014. 2014. Disponível em: < http://www.repositoriobib.ufc.br/000017/0000179f.pdf> Acesso em: 14 dez. 2016.

CHARTS. Google. Disponível em: <https://developers.google.com/chart/interactive/docs/>. Acesso em: 22 fev. 2017.

DE AMO, Sandra. Técnicas de mineração de dados. **Jornada de Atualização em Informática**, 2004.

DE MAGALHÃES, Lúcia Helena. **Uma análise de ferramentas para mineração de conteúdo de páginas *Web***. 2008. Tese de Doutorado. UNIVERSIDADE FEDERAL DO RIO DE JANEIRO.

DO CARMO, Sidney Gonçalves. Com PM em greve, ES tem aumento de violência e pede ajuda do Exército. Disponível em: <http://www1.folha.uol.com.br/cotidiano/2017/02/1856179-com-pm-em-greve-es-tem-aumento-de-violencia-e-pede-ajuda-do-exercito.shtml>. Acesso em: 22 fev. 2017

EL-KHAIR, Ibrahim Abu. *Effects of stop words elimination for Arabic information retrieval: a comparative study*. ***International Journal of Computing & Information Sciences***, v. 4, n. 3, p. 119-133, 2006.

FAYYAD, Usama; PIATETSKY-SHAPIRO, Gregory; SMYTH, Padhraic. *From Data Mining to knowledge discovery in databases.* **AI magazine**, v. 17, n. 3, p. 37, 1996.

FÖRSTER, Thorsten *et al. The Tweet and the city: Comparing Twitter activities in informational world cities*. In*:* ***Proceedings of the 2014 Conference: Informationsqualität und Wissensgenerierung, Frankfurt am Main, Germany***. 2014. p. 8-9.

GEMS. Ruby Gems Guide. Disponível em: < http://guides.rubygems.org/ >. Acesso em 19 jan. 2017.

GONCALVES, Pollyanna; BENEVENUTO, Fabrício; ALMEIDA, Virgílio. O que *Tweets* contendo emoticons podem revelar sobre sentimentos coletivos. In*:* ***II Brazilian Workshop on Social Network Analysis and Mining (BraSNAM*****)**. 2013.

HAND, David J.; MANNILA, Heikki; SMYTH, Padhraic. ***Principles of Data Mining***. MIT press, 2001.

JSON. **Introducing JSON**. Disponível em: <http://www.json.org/>. Acesso em: 19 jan. 2017.

KUO, Byron YL *et al. Tag clouds for summarizing web search results. In:* ***Proceedings of the 16th international conference on World Wide Web***. ACM, 2007. p. 1203-1204.

MAKICE, Kevin. ***Twitter API:*** *Up and running: Learn how to build applications with the Twitter API*. O'Reilly Media, Inc., 2009.

*MONGODB*. **Mongodb**. Disponível em: <https://docs.mongodb.com/ruby-driver/master/>. Acesso em 29 jan. 2017.

NEVES, Lidia. **Eleições nos Estados Unidos ocorrem por meio de delegados.** Disponível em: <http://agenciabrasil.ebc.com.br/internacional/noticia/2016-11/eleicoes-nos-estados-unidos-ocorrem-por-meio-de-delegados-entenda>. Acesso em 29 jan. 2017.

*RUBY.* **Sobre o Ruby**. Disponível em: <https://www.ruby-lang.org/pt/about/>. Acesso em: 19 jan. 2017.

*RAILS*. ***Welcome to Rails***. Disponível em: <http://api.rubyonrails.org/>. Acesso em: 19 jan. 2017.

RUSSELL, Matthew A. ***Mining the Social Web***. 2nd Edition. Sebastopol, CA: O'Reilly Media, 2014.

SANTOS, Leandro M. **Protótipo para mineração de opinião em redes sociais:** estudo de casos selecionados usando o *Twitter*. 2010. Disponível em: < http://repositorio.ufla.br/bitstream/1/5190/1/MONOGRAFIA_Prototipo_para_mineracao_de_opiniao_em_redes_sociais_estudo_de_casos_selecionados_usando_o_twitter.pdf> Acesso em: 19 jan. 2017.

SANTOS, Rafael et al. Conceitos de Mineração de Dados na Web. **XV Simpósio Brasileiro de Sistemas Multimıdia e *Web*, VI Simpósio Brasileiro de Sistemas Colaborativos– Anais**, p. 81-124, 2009.

SCHNEIDER, Vania et al. Critérios De Usabilidade Em Processo De Migração De Biblioteca Gráfica Em Um Sistema De Informações Ambientais. ***Scientia cum Industria***, v. 4, n. 2, p. 103-107, 2016.

*SENTIMENTALIZER*. **Rubydoc**. Disponível em: <http://www.rubydoc.info/gems/sentimentalizer/0.3.0>. Acesso em: 24 fev. 2017.

SILVA, Gabriel Luiz Andriotti Da. ***Text Mining*****, um estudo a partir da rede social *Twitter***. 2013.

SOARES JUNIOR, J. S. *et al.* Uma análise estatística dos indicadores de criminalidade de Salvador. **Conjuntura & Planejamento**, v. 161, p. 40-49, 2008.

THOMÉ, Antonio Carlos Gay. **Redes neurais: uma ferramenta para *KDD* e *Data Mining***. 2008. Disponível em: < http://equipe.nce.ufrj.br/thome/grad/nn/mat_didatico/apostila_kdd_mbi.pdf> Acesso em: 15 nov. 2016.

*TWITTER*. ***About the Twitter***. Disponível em: <https://about.twitter.com/pt/company>. Acesso em: 15 nov. 2016a.

*TWITTER*. ***Libraries***. Disponível em < https://dev.twitter.com/resources/twitter-libraries>. Acesso em: 16 dez. 2017.

*TWITTER*. ***Rest APIs***. Disponível em: <https://dev.twitter.com/rest/public/search>. Acesso em: 15 nov. 2016b.

*TWITTER*. ***Streaming APIs***. Disponível em: <https://dev.twitter.com/streaming/overview>. Acesso em: 15 nov. 2016c.

*TWITTER*. ***OAuth***. Disponível em <https://dev.twitter.com/oauth/application-only>. Acesso em: 16 dez. 2016d.

WITTEN, Ian H. *et al.* ***Data Mining:*** *Practical machine learning tools and techniques*. Morgan Kaufmann, 2011.

ZHU, Nick Qi. ***Data visualization with D3. js cookbook***. Packt Publishing Ltd, 2013.

ZUK, Torre et al. *Heuristics for information visualization evaluation. In:* ***Proceedings of the 2006 AVI workshop on BEyond time and errors: novel evaluation methods for information visualization***. ACM, 2006. p. 1-6.