Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

TwitSisbra: Detecção de Terremotos através da Rede Social Twitter

Matheus Lima de Andrade
Thiago Mitio Kobayashi

Monografia apresentada como requisito parcial
para conclusão do Bacharelado em Ciência da Computação

Orientador
Prof. Dr. Maristela Terto De Holanda

Brasília
2016

Universidade de Brasília — UnB
Instituto de Ciências Exatas
Departamento de Ciência da Computação
Bacharelado em Ciência da Computação

Coordenador: Prof. Dr. Rodrigo Bonifácio de Almeida

Banca examinadora composta por:

Prof. Dr. Maristela Terto De Holanda (Orientador) — CIC/UnB
Prof. Dr. George Sand França — OBSIS/UnB
Prof. Mr. Henrique Pereira de Freitas Filho — DAA/IFG

CIP — Catalogação Internacional na Publicação

Andrade, Matheus Lima de.

TwitSisbra: Detecção de Terremotos através da Rede Social Twitter / Matheus Lima de Andrade, Thiago Mitio Kobayashi. Brasília : UnB, 2016.
109 p. : il. ; 29,5 cm.

Monografia (Graduação) — Universidade de Brasília, Brasília, 2016.

1. Twitter, 2. Redes Sociais, 3. Eventos Sísmicos, 4. Mineração de Dados

CDU 004.4

Endereço: Universidade de Brasília
Campus Universitário Darcy Ribeiro — Asa Norte
CEP 70910-900
Brasília–DF — Brasil

Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

TwitSisbra: Detecção de Terremotos através da Rede Social Twitter

Matheus Lima de Andrade
Thiago Mitio Kobayashi

Monografia apresentada como requisito parcial
para conclusão do Bacharelado em Ciência da Computação

Prof. Dr. Maristela Terto De Holanda (Orientador)
CIC/UnB

Prof. Dr. George Sand França
OBSIS/UnB

Prof. Mr. Henrique Pereira de Freitas Filho
DAA/IFG

Prof. Dr. Rodrigo Bonifácio de Almeida
Coordenador do Bacharelado em Ciência da Computação

Brasília, 18 de Agosto de 2016

Dedicatória

Eu, Matheus, dedico este trabalho à todos aqueles que participaram de algum momento da minha vida acadêmica, permitindo das mais diversas maneiras que eu pudesse concluir o curso que eu tanto sonhei numa Universidade incrível. Dedico especialmente para meus pais, meus irmãos, minha namorada e sua família, e aos meus amigos e colegas que conheci neste percurso.

Matheus Lima de Andrade

Dedico este trabalho à minha família, noiva e amigos.

Thiago Mitio Kobayashi

Agradecimentos

Agradeço a Deus, nosso pai e amigo, que me sustentou e me guiou por TODOS os momentos até hoje na minha vida, e que me deu muito mais daquilo que pedi ou pensei.

Agradeço ao meu pai, Ivan, e a minha mãe, Helenilze, que deram tudo e muito mais daquilo que podiam para que seus três filhos pudessem estudar numa Universidade Federal, ajudando não somente na questão financeira, como também na educação que vêm do lar e principalmente nos instruindo nos caminhos do SENHOR.

Agradeço por meus irmãos, Paulo e Lucas, que me incentivaram na caminhada acadêmica.

Agradeço ao meu tio Celso e também Valdívia e sua família, por abrirem as portas de sua casa nos momentos iniciais aqui em Brasília.

Agradeço ao Walder, a Eunice e suas filhas, por abrirem as portas de sua casa para mim.

Agradeço pela Primeira Igreja Batista no Cruzeiro Novo, por me receber de braços abertos e ser como uma família para mim.

Agradeço por ter conhecido a mulher que Deus preparou pra mim aqui nessa cidade, Dórean, que me incentivou, apoiou, e esteve ao meu lado em todos os momentos. Agradeço também à sua família que me trata como um filho.

Agradeço a cada um dos meus colegas e amigos que fiz durante esses anos aqui na UnB, que me ajudaram, e me incentivaram a prosseguir.

Agradeço aos professores que me trouxeram o conhecimento das mais diversas maneiras possíveis.

Agradeço especialmente a professora Maristela, que se dispôs de maneira muito solicita a me orientar na execução deste trabalho. Agradeço também ao professor George Sand, pela disposição em ajudar de maneira sempre amigável e pronto a tirar dúvidas.

Por fim agradeço a todos que de alguma maneira, seja pelas orações, ajudando nos trabalhos, me motivando, tornaram este sonho realidade.

MUITO OBRIGADO,
Matheus Lima de Andrade

Agradeço a Deus por sempre estar comigo em momentos de dificuldade ou de alegria. Pela minha família por me fornecerem todo apoio que precisei. E à minha noiva, que sempre está ao meu lado me ajudando ser melhor.

Agradeço também a professora Maristela e o professor George Sand por terem nos orientado por toda a realização deste trabalho.

Thiago Mitio Kobayashi

Resumo

O contínuo aumento das redes sociais através da internet nos últimos anos permitiu aos usuários se conectarem e compartilharem informações em tempo real, espalhando essas informações a milhares de outros usuários em um curto espaço de tempo. Os usuários dessas redes costumam postar suas opiniões sobre diversas situações que ocorrem ao seu redor como grandes eventos, epidemias, catástrofes, entre outros acontecimentos. No caso de desastres, sejam eles naturais ou não, é possível que a informação possa chegar mais rápido pela comunicação entre os usuários dessas redes do que dos centros de pesquisa. Entretanto, essa quantidade de dados continua crescendo, e a análise desses dados de forma não automatizada pode ser um problema. Para que se possa armazenar e processar esses dados relacionando-os com a geografia local, existem os Bancos de Dados Geográficos.Estes possibilitam a armazenagem de atributos geográficos, tornando as informações retiradas desses meios ainda mais completa. Dessa forma, este trabalho mostra como o processo de mineração de dados foi usado para coletar, estruturar e analisar o texto extraído do *Twitter* e como permitir que as postagens dos usuários, conhecidas como *tweets*, que tratam de eventos sismológicos, podem contribuir para tomada de decisão rápida quando desastres acontecem.

Palavras-chave: Twitter, Redes Sociais, Eventos Sísmicos, Mineração de Dados

Abstract

The continous rise of social networks through the internet in recent years has allowed users to connect and share information in real time, spreading information to thousands of other users in a short time. Users of these networks often post their opinions about several situations that occur around them as big events, epidemics, disasters, and other situations. In case of disasters, whether natural or not, it is possible that the information can arrive faster through communication between the users of these networks than the research centers. However, this amount of data continues to grow, and the not automated analysis of these data can be a problem. To store and process the data, conecting them with local geographic information, there are Geographic Data Bases, which allow the storage of geographic atributes, making the information retrieved from those sources even more reliable. Thus, this work shows how the data mining process was used to collect, structure and analyze the extracted text of Twitter and how to allow the posts of users, known as tweets, that address seismological events can contribute to quickly decisions when disasters happen.

Keywords: Twitter, Social Networks, Seismic Events, Data Mining

Sumário

Lista de Figuras

Capítulo 1

Introdução

Com o passar dos anos, novas áreas foram povoadas e as cidades se estabeleceram, formando grandes aglomerados urbanos. Dessa forma, os danos causados pelos desastres naturais passam a atingir proporções catastróficas. Em 1755, ocorreu o terremoto de Lisboa, Portugal, que atingiu 8,6 graus na escala Richter, vitimando mais de 30.000 pessoas, por decorrência dos tremores de terra, do tsunami e dos incêndios que devastaram a cidade [43]. Na Jamaica, em 1962, um terremoto destruiu a cidade de Porto Royal, matando milhares de pessoas.

No Brasil também foram registrados alguns desses eventos. Os dados registrados pela rede sismográfica provisória de Montes Claros mostraram que a partir de 1995 houve um número expressivo de tremores na cidade situada no norte de Minas Gerais, cerca de cento e dezessete tremores foram registrados [33].

Conhecidos como tremores, terremotos ou sismos, esses eventos geológicos da natureza, se diferem somente na intensidade, porém todos se referem ao mesmo fenômeno. Tais fenômenos quando ocorrem possuem uma capacidade destrutiva inimaginável, podendo dizimar vidas, casas e até cidades inteiras, gerando assim, além de um valor inestimável pelas vidas perdidas, uma infinidade de perdas materiais.

Esses fenômenos geológicos acontecem devido ao sistema de acúmulo de pressões, no qual as rochas, ao atingirem certo limite de resistência, não suportam. Essas pressões são geradas pela dinâmica da Terra, que geram assim a movimentação das placas tectônicas. Estas placas podem se movimentar de diversas formas, como o afastamento, colisão e o deslizamento sobre outra placa. Com o acúmulo da energia gerada por essa atividade, a matéria entra em ruptura e sofre libertação brusca de toda energia acumulada, na forma de ondas sísmicas.

Pode-se observar que os principais terremotos acontecem nas regiões limítrofes dessas placas e muitos dos sismos, que tem menor intensidade, também acontecem nessas regiões, embora possa haver ocorrências desses eventos também em regiões que não façam limite com outras placas.

O Observatório Sismológico da Universidade de Brasília(SIS-UnB), criado em 1980, tem o objetivo de analisar e estudar os movimentos ocorridos no interior da Terra [53]. Sua principal atividade é o monitoramento sismográfico brasileiro. O SIS capta e monitora, através dos sismógrafos, as ondas sísmicas que são geradas pelos terremotos e explosões que se propagam pelo interior da Terra. Os sismógrafos são equipamentos sensíveis às

passagens de ondas sísmicas geradas à longa distância e convertem essas ondas em sinais elétricos.

A quantidade de informação recebida diariamente pelo SIS é muito grande que para ser analisada e estudada, precisa passar por um processamento, a fim de que possa ser utilizados no seu propósito.

O propósito do SIS é organizar essas informações para se tornarem um banco de dados concreto sobre a realidade sísmica do Brasil. Dessa forma é possível obter informações precisas e consistentes acerca de abalos sísmicos em um estado, município ou região.

Para gerenciar os registros sísmicos está sendo criado o Sistema Nacional de Registro sísmicos(WebSisbra) [32], um projeto desenvolvido em conjunto pelo Departamento de Computação da UnB e pelo SIS, que tem como objetivo ser um sistema capaz de otimizar os recursos da pesquisa e trabalhos desenvolvidos pelo SIS por meio de acesso à um banco de dados geográficos, além de providenciar uma interface simples e amigável, de forma que qualquer usuário possa acessar e entender as informações exibidas.

Com a chegada da Web 2.0, a computação ubíqua, e os avanços tecnológicos correspondentes, as mídias sociais alteraram drasticamente os conceitos de contribuição, disseminação e troca de informação. O rápido crescimento do uso da Internet e a popularização das redes sociais promoveram uma mudança na forma de interação das pessoas entre si. Usuários publicam suas opiniões acerca de organizações, eventos, catástrofes, entre outros, em seus perfis de redes sociais, que rapidamente se propagam entre os diversos usuários destas redes. Essa popularização da Internet, por sua vez, amplia as possibilidades e permite o estudo de como os indivíduos inseridos nas redes sociais comunicam-se, informam, criam conteúdos e interagem entre si de forma geral.

A Mineração de textos, que é uma parcela da Mineração de dados, também conhecida como Descoberta de conhecimento em textos, fornece um conjunto de técnicas que podem automatizar o processo de coleta e estruturação de informações. É possível, então, analisar os objetos coletados e classificá-los.

Com essa facilidade advinda das comunicações por essas redes e pela rapidez da comunicação nesse meio, é possível visualizar se eventos catastróficos, como terremotos e abalos sísmicos são disseminados pelos usuários dessas redes, possibilitando assim a criação de um alerta efetivo contra o poder devastador desses fenômenos.

1.1 Objetivo

O objetivo deste trabalho é promover a coleta de dados com origem na rede social *Twitter*, e categorizar os *tweets* de acordo com as informações sobre possíveis desastres sísmicos. Além disso, promover uma maneira de alertar o Observatório Sismológico de Brasília, e assim as autoridades, sobre possíveis desastres causados por tais fenômenos em território brasileiro.

1.2 Justificativa

Atualmente o SIS possui o Websisbra, um eficiente sistema que gerencia as informações coletadas, pelos sismógrafos espalhados pelo Brasil, resultantes de eventos sísmicos dos últimos anos no país. Entretanto, a informação recebida pela troca de informação nas

redes sociais, pode trazer um resultado mais rápido aos Observatórios Sismológicos do que os sismógrafos, devido à intensidade e rapidez cujas informações são veiculadas na Internet, fazendo assim com que as autoridades possam tomar decisões mais eficientes, no sentido de salvar vidas.

1.3 Estrutura

Este trabalho está dividido nos seguintes capítulos:

- Capítulo 2: Este capítulo aborda o referencial teórico do conteúdo proposto no trabalho, e se divide em três partes. A primeira parte apresenta os conceitos básicos sobre Sistemas de informação geográfica(SIG), no que se baseia modelagem cartográfica, arquiteturas de um SIG, Tecnologia dos SIG, como é feita a obtenção e conversão de dados geográficos e os tipos de mapas existentes na Internet. A segunda parte aborda a rede social *Twitter* e suas características. Por último, a terceira parte aborda os conceitos de Descoberta de conhecimento, seja através processamento de Bancos de dados usuais ou Bancos de dados textuais, e as fases para se atingir os objetivos de cada um dos processos, além de tratar sobre a Mineração na web.
- Capítulo 3: Este capítulo apresenta como aconteceu todo o projeto de criação do sistema TwitSisbra, começando com a elaboração da arquitetura do sistema, até a interface onde serão exibidos os resultados, através do uso de diversas tecnologias.
- Capítulo 4: Apresenta as conclusões obtidas ao final do desenvolvimento do trabalho.

Capítulo 2

Referencial Teórico

Este capítulo aborda os conceitos fundamentais utilizados neste trabalho. Sua estrutura é composta por três seções: Sistemas de informação geográfica, redes sociais e descoberta de conhecimento. A seção 2.1 aborda os conceitos básicos de Sistemas de informação geográfica através das subseções: Sistemas de Informação Geográficas, Modelagem Cartográfica, Arquitetura de um SIG, Tecnologia dos Sistemas de Informação Geográfica, Obtenção e Conversão de Dados Geográficos, Tipos de Mapas da Internet. A seção 2.2 parte compreende uma visão geral das redes sociais e mais especificamente da rede social *Twitter*. A seção 2.3 relaciona o que é e como são empregadas as técnicas de Descoberta de conhecimento e Mineração de dados, e sua estrutura é composta por: Descoberta de conhecimento, Descoberta de Conhecimento em Bases de Dados, Fases do Processo de Descoberta de Conhecimento em Bases de Dados - KDD, Descoberta de Conhecimento em Bases Textuais, Fases do Processo de Descoberta de Conhecimento em Bases de Textos - KDT e Mineração na Web.

2.1 Sistemas de Informação Geográficas

Um sistema é um conjunto de elementos inter-relacionados com um objetivo comum. Além disso possui características e leis que o regem. Ou seja, sistema pode ser definido como um conjunto de elementos interdependentes que interagem com objetivos comuns formando um todo, e onde cada um dos elementos componentes comporta-se, por sua vez, como um sistema cujo resultado é maior do que o resultado que as unidades poderiam ter se funcionassem de forma independente [62].

Já sistema de informação é descrito como sendo o sistema que recolhe, processa, armazena e distribui informação numa organização tendo em vista que a informação esteja acessível a quem dela necessita. Um sistema de informação é assim um sistema de atividade humana que poderá ser suportado por computadores [57].

Um Sistema de Informação Geográfica(SIG) é um sistema de informações projetado para a coleta, armazenamento e análise de objetos e fenômenos, sustentados pela localização geográfica. O termo SIG é aplicado para sistemas que realizam o tratamento computacional de dados geográficos e se diferenciam dos sistemas de informações usuais quanto a capacidade de armazenar tanto os atributos descritivos como as geometrias dos diferentes tipos de dados geográficos [27].

Figura 2.1: Como o SIG é utilizado em diversas áreas [12].

Os dados geométricos e alfanuméricos, dessa forma interligados, suprem sistemas computacionais, o que possibilita a análise de problemas predeterminados. Um SIG permite a visualização espacial dos dados através de interfaces gráficas dos sistemas e/ou através da confecção de mapas impressos, nos quais são ilustradas as soluções de problemas. Os dados espaciais são observações documentadas ou resultados da medição. A disponibilidade dos dados oferece oportunidades para a obtenção de informações. Os dados podem ser obtidos pela percepção, através dos sentidos (por exemplo, observação), ou pela execução de um processo de medição. Uma base de dados geográfica é um depósito de fatos ou conceitos do mundo real que possuem atributos convencionais e atributos espaciais que descrevem sua forma e indicam sua localização na Terra.

Nos últimos anos o uso do geoprocessamento como ferramenta de suporte à tomada de decisão tem se consolidado, tendo saído do meio acadêmico para alcançar o mercado com uma velocidade incrível. Órgãos do Governo e grandes empresas começaram a investir no uso de aplicativos disponíveis no mercado como o ArcGIS da ESRI, AutoCAD MAP da Autodesk, dentre outros. As aplicações *desktop* que agregam diversas funções no mesmo sistema (modelagem 3D, análise espacial, processamento digital de imagens, dentre outros) se encontram cada vez mais consolidadas [27].

Com o uso da web já consolidada o SIG continua em busca de mais popularização (por demandas do próprio mercado), evolui e passa a fazer uso também do ambiente web. Os aplicativos podem ser simples, com funcionalidades básicas de consulta à mapas e a bases alfanuméricas, até softwares complexos de análise de informações geográficas. Os usuários não precisam mais ser especialistas em computação, facilitando o acesso de pessoas e instituições não ligadas à essa área. O momento atual é caracterizado por um salto no número de usuários, o surgimento de sites especializados, revistas, dentre outros

[27].

2.1.1 Modelagem Cartográfica

Um modelo cartográfico é uma representação gráfica dos dados e dos procedimentos analíticos usados em um estudo, seu propósito é ajudar o analista a organizar e estruturar os procedimentos que serão usados e a identificar os dados necessários. Dessa maneira, um modelo é um conjunto de regras e procedimentos para representar um fenômeno e prever um resultado, consistindo numa sequência de processos ligados uns aos outros [51].

Os modelos retratam certos aspectos da realidade. Quanto mais fatores são levados em conta no modelo, mais complexo ele se torna, seu manejo se torna caro e mais pesado em termos computacionais. Nem sempre um modelo mais complexo é o melhor, tudo depende do problema a ser solucionado.

A modelagem é uma aproximação metodológica que integra um conjunto de técnicas com o propósito de lidar com a resolução de problemas espaciais. Um modelo serve para testar horizontes alternativos, modificando variáveis e por vezes pode ser útil inclusive para fazer previsões.

Uma das vantagens da modelagem cartográfica está na transparência de relação entre os dados e o seu processamento. A desvantagem é que este método mostra somente uma visão associada aos dados que se escolheu colocar como variáveis no processo, fazendo com que outros dados, relações e processos possam ter ficado de fora. Um modelo é muito mais útil e conveniente quanto mais variáveis se conseguirem usar. Seus erros se baseiam no fato de que não é possível identificar todas as variáveis presentes no processo.

O procedimento de representar esquematicamente um modelo é através da álgebra de mapas, que se baseia numa maneira de agrupar os métodos através dos quais as variáveis e as operações vão desenvolver o modelo [51]. Os conceitos da álgebra de mapas são semelhantes aos da matemática, seguindo uma lógica e uma terminologia através da qual se compõem esquemas ou equações, sendo que estas equações podem ser simples ou complexas. Assim como na matemática são utilizados símbolos para representar valores reais ou variáveis, em álgebra de mapas são utilizados símbolos para representar as variáveis geográficas. Conjuntamente aos operadores matemáticos (soma, subtração, multiplicação e divisão), também se usam operadores lógicos para representar as relações entre variáveis ou as combinações entre mapas.

O modelo de dados adota uma estratégia de especificação que identifica quatro níveis(ou universos) de abstração [28]:

- **Nível do mundo real:** contém os elementos da realidade geográfica a serem modelados como, por exemplo, rios, temperatura, redes telefônicas.
- **Nível conceitual:** comporta as ferramentas para modelar formalmente campos e objetos geográficos em um nível alto de abstração. Este nível determina as classes orientadas a objetos básicas que deverão ser criadas no banco de dados. Neste nível devem ser também definidas as operações e a linguagem de manipulação de dados disponíveis para o usuário.
- **Nível de representação:** associa as classes de campos e objetos geográficos identificadas no nível conceitual a classes de representações, que podem variar conforme

a escala, a projeção cartográfica escolhida, a época de aquisição do dado, ou mesmo conforme a visão do usuário ou aplicação.

- **Nível de implementação (físico ou interno):** define padrões, formas de armazenamento e estruturas de dados para implementar as diferentes representações. As decisões de implementação abordadas admitem um número muito grande de variações, em função das aplicações às quais o sistema é destinado, a disponibilidade de algoritmos e o desempenho do hardware.

Camadas

As camadas são um conceito antigo da cartografia tradicional que se baseava em envolver informações de impressão em folhas transparentes e sobrepondo essas folhas para visualizar um todo.

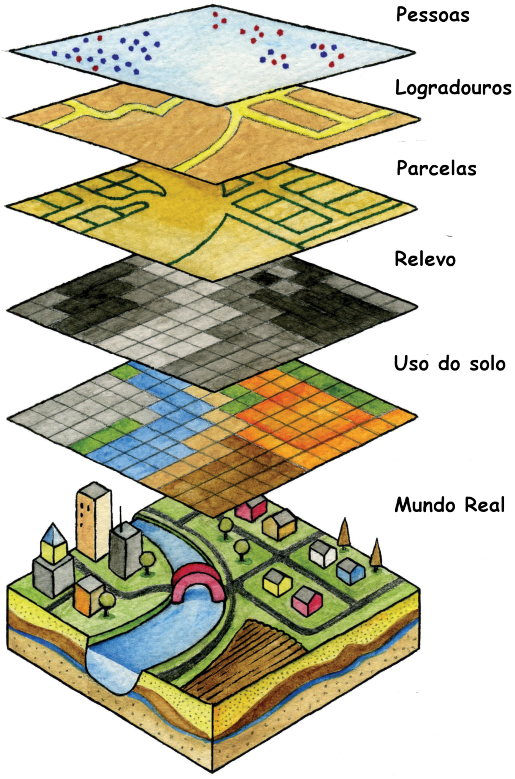

Figura 2.2: Ilustração das diversas camadas de um SIG [2].

As camadas são a essência de um SIG e possibilitam a visualização de altas quantidades de dados sob muitas características e condições diferentes concomitantemente [53]. Tais camadas são representadas através de um formato de armazenamento digital, que contém características das camadas acima ou abaixo da superfície da terra. De acordo o tipo de características que representam, as camadas podem ser de dois tipos de dados espaciais:

- **Vetorial:** trata-se das características representadas por pontos, linhas e polígonos;
- **Matricial(*Raster*):** retrata o mundo real como uma matriz retangular dividida por quadrados.

Representação Vetorial

"As representações vetoriais, têm em comum o fato de que os domínios espaciais são representados por conjuntos de traços, deslocamentos ou vetores, adequadamente referenciados"[50].

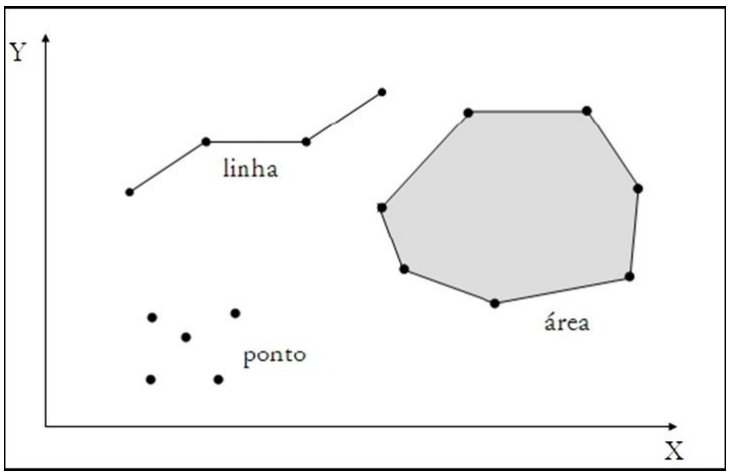

Figura 2.3: Representação vetorial em um SIG [11].

- **Ponto**:
  - Geralmente utilizado na representação de objetos de pequenas dimensões espaciais.
  - Usa um par de coordenadas simples para representar a localização de uma entidade.
  - O tamanho ou a dimensão da entidade pode não ser uma informação importante, somente sua localização pontual.

  Ex: Lotes podem ser representados na base espacial por um ponto, e ter armazenados como atributos, área, proprietário, tipo de uso, etc.

- **Linha**:
  - Definida como um conjunto ordenado de pontos interligados por segmentos de reta (polígono aberto).
  - O ponto inicial e o final são denominados nós e os pontos intermediários são chamados de vértices.
  - É utilizada na representação de entes cuja largura não convém ser expressada graficamente.

  Ex: estradas, cursos de água, redes de saneamento, redes de linhas de transmissão de energia elétrica, entre outros.

- **Área(Polígono)**:

– É usada para representar áreas e é definida como um conjunto ordenado de pontos interligados, onde o primeiro e último ponto coincidem
– Atributos podem ser associados aos polígonos como área, perímetro, uso e ocupação do solo, nome, etc.

Ex: Lotes, quadras, unidades territoriais, propriedades rurais.

Representação Matricial

No modelo matricial (ou *raster*) o terreno é representado por uma matriz M(i, j), composta por i colunas e j linhas, que definem células, denominadas como *pixels* (*picture element*).

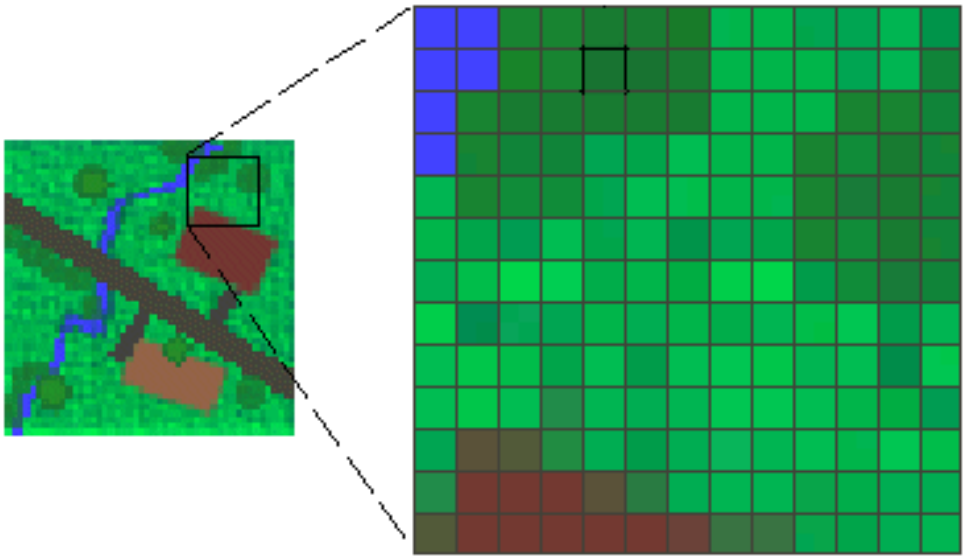

Figura 2.4: Representação matricial em um SIG [10].

Cada *pixel* apresenta um valor referente ao atributo, além dos valores que definem o número da coluna e o número da linha, correspondendo, quando o arquivo está georreferenciado, a um par de coordenadas x e y que se encontre dentro da área abrangida por aquele pixel.

- Formato compatível com dados originados de scanners e sensores remotos.
- Forma mais adequada para representar formas ou fenômenos contínuos no espaço, como: elevação, precipitação, declividade ou dados geoquímicos.

2.1.2 Arquitetura de um SIG

Numa visão abrangente, pode-se indicar que um SIG tem os seguintes componentes:

- Interface com usuário;
- Entrada e integração de dados;

- Funções de consulta e análise espacial;
- Visualização e plotagem;
- Armazenamento e recuperação de dados (organizados sob a forma de um banco de dados geográficos) [28].

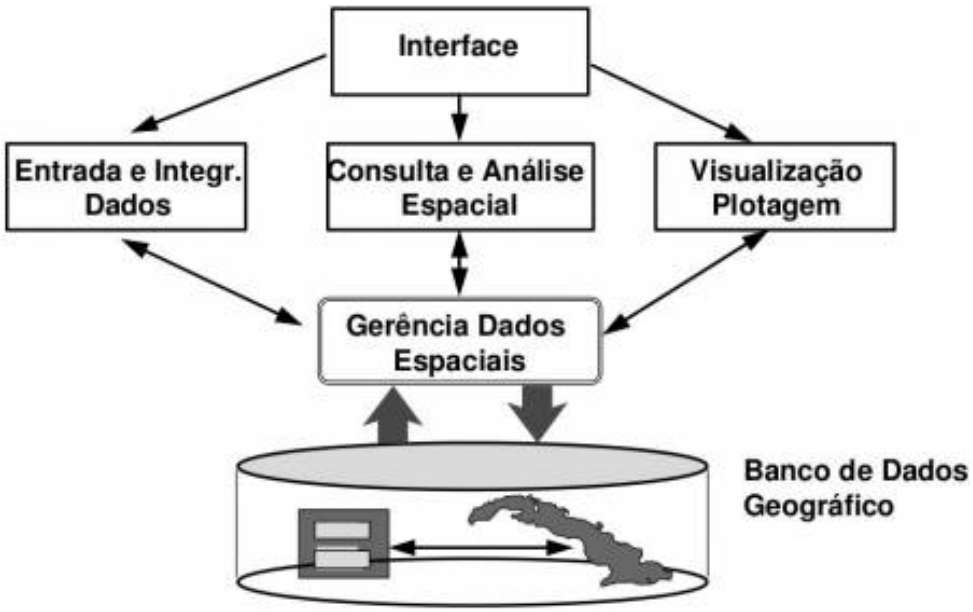

Figura 2.5: Diversas componentes de um SIG [1].

Os componentes do SIG se relacionam seguindo uma hierarquia. O grau mais interno do sistema, possui um sistema de gerência de bancos de dados geográficos que oferece armazenamento e recuperação dos dados espaciais e seus atributos. Na camada intermediária, um SIG deve ter características que permitam o processamento de dados espaciais (entrada, edição, análise, visualização e saída). No nível mais alto que relaciona a interação humano-computador, a interface define como o sistema é operado e controlado.

De maneira geral, as funções de processamento de um SIG operam sobre dados em uma área de trabalho em memória principal. A ligação entre os dados geográficos e as funções de processamento do SIG é feita por mecanismos de seleção e consulta que definem restrições sobre o conjunto de dados [28].

Na atualidade, a grande diferença entre os SIGs é a maneira na qual os dados geográficos são geridos. Há basicamente três diferentes arquiteturas de SIGs que utilizam os recursos de um Sistema Gerenciador de Banco de Dados(SGBD): dual, integrada baseada em SGBDs relacionais e integrada baseada em extensões espaciais sobre SGBDs objeto-relacionais [28].

Arquitetura Dual

Um SIG implementado com a estratégia *Dual* usa um SGBD relacional para manter os atributos convencionais dos objetos geográficos (organizados em tabelas) e arquivos para armazenar as representações geométricas destes objetos. No modelo relacional, os

dados são classificados na forma de uma tabela onde as linhas representam os dados e as colunas representam os atributos. A inserção dos atributos não-espaciais é feita por meio de um SGBD relacional e para cada entidade gráfica inserida no sistema é adicionado um identificador único ou rótulo, pelo qual é feita uma ligação lógica com seus devidos atributos não-espaciais armazenados em tabelas de dados no SGBD.

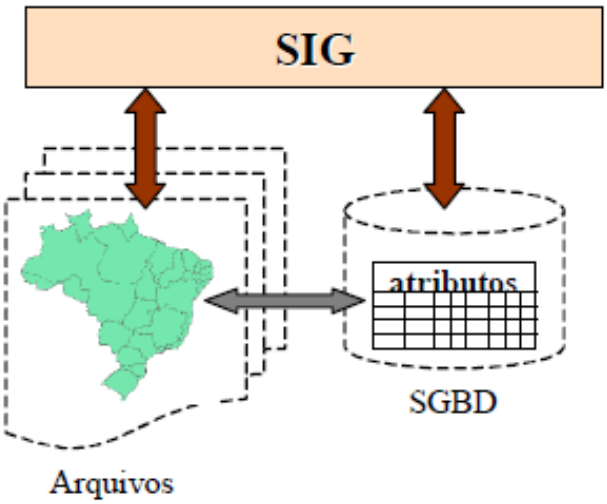

Figura 2.6: Representação da Arquitetura Dual [1].

A grande vantagem desta técnica está em poder utilizar os SGBDs relacionais de mercado. Entretanto, como as representações geométricas dos objetos espaciais estão fora do controle do SGBD, esta estrutura dificulta a comparação das questões de otimização de consultas, gerência de transações e controle de integridade e de concorrência [28].

As desvantagens desta arquitetura são:

- Dificuldades no controle e manejo dos dados espaciais;
- Dificuldade em manter a integridade entre as componentes geográficas e convencionais;
- Consultas mais lentas, devido ao seu processamento ser feito em separado. A parte convencional da consulta é processada pelo SGBD separado da parte espacial, que é processada pelo aplicativo utilizando os arquivos proprietários;
- Dados comportam-se de forma independente. Cada sistema produz seu próprio arquivo proprietário sem seguir um formato padrão, dificultando a integração destes dados [28].

Arquitetura Integrada

A Arquitetura integrada, consiste em armazenar todo o dado espacial em um SGBD, tanto sua componente espacial como a parte não-espacial. Sua grande vantagem está em utilizar os recursos de um SGBD para controle e manipulação de dados espaciais, como

gerência de transações e controle de integridade e concorrência. Portanto, a manutenção de integridade entre a componente espacial e não-espacial é deixada à cargo do SGBD.

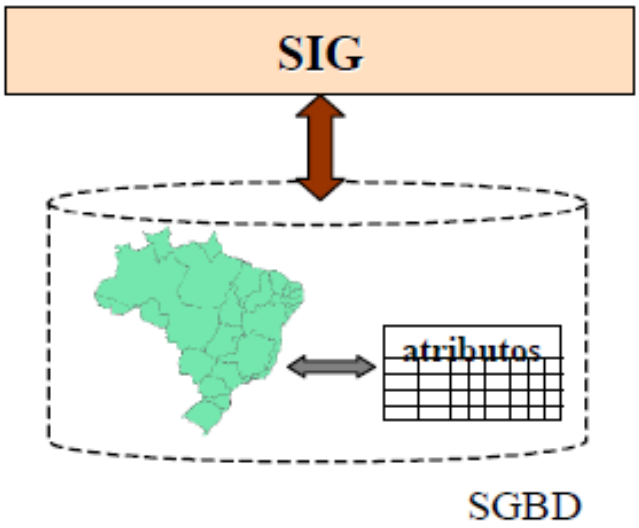

Figura 2.7: Representação da Arquitetura Integrada [1].

Há duas alternativas para a arquitetura integrada:

- (a) baseada em SGBDs relacionais;
- (b) baseada em extensões espaciais sobre SGBDs objeto-relacionais.

A arquitetura integrada baseada em um SGBD relacional utiliza campos longos, chamados de *BLOB*, para armazenar a parte espacial dos dados. Possui como desvantagens:

- Não é capaz de capturar a semântica dos dados espaciais: como o SGBD trata o campo longo como uma cadeia binária, não é possível conhecer a semântica do seu conteúdo;
- Métodos de acesso espacial e otimizador de consultas devem ser implementados pelo SIG: o SGBD não possui mecanismos satisfatórios para o seu tratamento;

O outro tipo de arquitetura integrada consiste em utilizar extensões geográficas desenvolvidas sobre SGBDs objeto-relacionais (SGBDOR). Estas extensões contêm funcões e técnicas que permitem armazenar, acessar e analisar dados geográficos de formato vetorial. Os SGBDs objeto-relacionais, também conhecidos por SGBDs extensíveis, oferecem aptidões para a definição de novos tipos de dados e métodos(operadores) para manipular esses tipos, estendendo assim seu modelo de dados e sua linguagem de consulta [28].

Como pontos fracos dessa arquitetura podem ser elicitados a ausência de mecanismos de controle de integridade sobre os dados espaciais e a ausência de padronização das extensões da linguagem SQL.

Tendo todas arquiteturas em vista, um SGBDOR é mais adequado para tratar dados complexos, como dados geográficos, do que um SGBD relacional, que não oferece esses recursos.

2.1.3 Tecnologia dos Sistemas de Informação Geográfica

O conceito de Sistemas de informação geográfica está relacionado a diversas variantes. Há uma grande diversificação de oferta de SIG, com pelo menos quatro grandes tecnologias que se destacam:

- Os SIG *desktop*, com interfaces amigáveis e funcionalidades crescentes.
- Os Gerenciadores de Dados Geográficos, que armazenam os dados geográficos em ambiente multi-usuário.
- Os Componentes SIG, ambientes de programação que fornecem recursos para que o usuário crie seu próprio software geográfico.
- Os Servidores web de dados geográficos, utilizados para publicação e acesso a dados geográficos através da Internet [27].

Os SIG *Desktop* são sistemas oriundos da cartografia tradicional, com suporte limitado a bancos de dados e cujo paradigma típico de trabalho é o mapa (ou "plano de informação"). Esta classe de sistemas é utilizada principalmente em projetos isolados, onde a preocupação de gerar arquivos digitais de dados é mínima.

A segunda geração de SIGs (bancos de dados geográfico) chegou ao mercado no início da década de 90 e sua característica se dá por ser idealizada para uso em ambientes cliente-servidor, ligado a gerenciadores de bancos de dados relacionais e com ferramentas adicionais para lidar com imagens.

A terceira geração de SIGs (Bibliotecas geográficas digitais), se caracteriza por gerenciar grandes bases de dados geográficos, com acesso através de redes locais e remotas, através da Internet. Com o crescimento dos bancos de dados espaciais e a necessidade de seu compartilhamento com outras instituições, o recurso a tecnologias como bancos de dados distribuídos se torna evidente. Estes sistemas deverão seguir os requisitos de operação entre si, a fim de permitir o acesso de informações espaciais por diferentes SIGs.

Um aspecto fundamental das variadas tecnologias apresentadas é que elas não são independentes: os SIGs *desktop* podem utilizar gerenciadores de dados geográficos, estes, que por sua vez, podem estar ligados a servidores web, e assim os usuários destes dados podem ter interfaces personalizadas, construídas a partir de componentes SIG [27].

2.1.4 Obtenção e Conversão de Dados Geográficos

Os dados são obtidos por meio de digitalização manual, de digitalização automática por meio de leitor ótico de dispositivos de varredura (*scanners*), digitação via teclado, de leituras de dados de fontes secundárias, como fitas magnéticas e discos óticos e também da Internet. Como resultado dessa etapa é possível ter uma grande quantidade de dados em forma bruta.

Um sistema de informação geográfica pode trabalhar com dados de outros projetos, mas para haver relacionamentos entre os dados mantidos em cada banco de dados de cada SIG, esses dados precisam ser convertidos.

A conversão mais simples que existe é conhecida como conversão sintática direta de formatos que realiza a interpretação e tradução dos arquivos de informações geográfica em diferentes formatos.

Após a obtenção e a devida conversão dos dados, caso necessário, eles são transferidos para o banco de dados do SIG.

2.1.5 Tipos de Mapas da Internet

Existem diversas formas de mapas na Internet. Há desde os que são estáticos e servem somente para fins estatísticos até os mais avançados que dão suporte à interação dinâmica com o usuário. Estes últimos necessitam de mais habilidade, devido ao seu desenvolvimento mais complexo. Alguns dos mapas online existentes são[53]:

- **Mapas analíticos:** São usados para fins de análises. Os dados geográficos utilizados podem ser fornecidos ou podem ser carregados pelo usuário no mapa. As análises executadas são, por diversas vezes, realizadas por um servidor SIG e o cliente apresenta a análise resultante.
- **Mapas interativos:** Mostram mudanças no mapa ao longo do tempo. Normalmente estes mapas utilizam alguma das diversas tecnologias existentes que permitem o uso de tipos de dados animados. Alguns deles são: Adobe Flash, Java, QuickTime, Unity. Esse tipo de mapa se revela interessante ao ser representado em alguns dos contextos específicos como no campo meteorológico, informações sobre fenômenos naturais e outros tipos.
- **Mapas colaborativos:** São ambientes onde várias pessoas colaboram para criar e melhorar os mapas na web, potencialmente desenvolvido no estilo do projeto Wikipédia [18]. Pode apresentar conteúdo fiel à realidade, por ter informações inseridas por indivíduos da sociedade. Entretanto, possui algumas dificuldades. É necessário que haja uma análise mínima de qualidade sobre os dados adicionados, antes que eles se tornem acessíveis publicamente. Outro aspecto diz respeito à como esses mapas se comportam em momento em que dois ou mais usuários tentam editar uma mesma área ao mesmo tempo. Nesse caso, a área editada deve ficar bloqueada e somente um usuário pode editar em um dado instante. Alguns projetos de colaboração de mapa existentes são: *Google Map Maker*, *OpenStreetMap*, *WikiMapia*.
- **Mapas dinâmicos:** São mapas criados a partir de origens dinâmicas, como bancos de dados, toda vez que o usuário recarrega as páginas da web. O servidor gera o mapa usando um servidor de mapas web ou um software de escrita automática.
- **Atlas online:** São os mapas antigos que eram feitos em forma impressa, de uma forma mais barata, acessível ao público em geral e com mais facilidade de serem atualizados devido ao meio em que está inserido, a Internet.
- **Mapas em tempo real:** Este tipo de mapa ilustra a ocorrência de um evento ou fenômeno em tempo real(ou o mais próximo possível disso). Os dados desses mapas podem ser coletados por sensores e os mapas são criados ou alimentados em períodos de tempo regulares ou à medida em que estes eventos ocorrem. Se encaixam bem nesse contexto eventos meteorológicos, desastres naturais, monitoramento de trânsito, etc.
- **Mapas estáticos:** Embora parecidos com os a Atlas Online, pois são produzidos com o propósito de não oferecerem nenhuma interatividade com usuário, se diferem

no ponto que estes mapas são criados apenas uma vez e serão raramente atualizados. Normalmente, os mapas estáticos são aqueles mapas de papel digitalizados, mas que não tinham sido projetados com a utilização de um SIG [53].

2.2 Redes sociais: Twitter

Com mais de 300 milhões de usuários ativos mensalmente, o *Twitter* é uma rede social que permite a seus usuários postarem mensagens de texto curtas, com limite de 140 caracteres conhecidas como *tweets*. Também é considerado uma mídia conhecida por ser mais informacional, com escrita informal e grande número de abreviações e termos peculiares. Através da sua estrutura dinâmica, qualquer usuário pode ter acesso às informações que são frequentemente postadas, permitindo que mesmo os usuários que não possuem alguma relação de conexão entre si, possam visualizar as publicações uns dos outros.

Os usuários do *Twitter* constroem ligações unilaterais uns com os outros, que podem ou não ser mútuas, conhecidas por seguir(*follow*). Assim um usuário é seguido por um grupo de usuários, como também aquele mesmo usuário é seguidor de outros. Além disso, essa rede possui características próprias que a difere de outras redes sociais. Algumas singularidades facilitam a difusão do conteúdo, que são elas:

- ***Retweet*:** permite a reprodução de um *tweet* existente, postado por outra pessoa, caso o usuário tenha achado importante. Dessa forma, o usuário compartilha com os seus seguidores o *tweet*. Esta é uma das principais formas de difundir conteúdo nesta rede.
- ***Hashtag*:** admite o usuário citar em seu *tweet* utilizando o símbolo # e em seguida um termo, geralmente uma palavra que resume o conteúdo a ser publicado, transformando em um *hiperlink*.
- **Assuntos do momento (*Trending Topics*):** lista em tempo real das frases mais publicadas no *Twitter* ao redor do mundo com a possibilidade de filtrar por país.

O *Twitter* disponibiliza duas *Application Programming Interface*(*API*) para acessos aos seus recursos por meio de requisições. São elas: a *Twitter REST API* e a *Twitter Streaming API*.

Outros mecanismos de apoio à supervisão de redes sociais podem ser encontrados, como o *Google Insights*, para comparação da intensidade de buscas por local, ou *Topsy*, para medir a influência de um usuário nas redes sociais baseado na quantidade de menções e compartilhamentos. Porém, ambientes como *Facebook* e *Myspace* não aceitam *APIs* para a extração de dados [54] de forma simplificada, fazendo-se necessário o desenvolvimento de uma aplicação específica obedecendo as normas definidas pelos serviços em questão e autorização de acesso aos dados de cada usuário [52].

A facilidade de recuperar dados, inclusive por meio de *APIs* cedidas pela própria empresa, é o grande diferencial do *Twitter* em relação a outras redes sociais populares, como o *LinkedIn* e o *Facebook*, tornando-o uma rede mais interessante de se explorar [52].

Utilizando uma das *APIs* do *Twitter* é possível, com a utilização de comandos específicos, desenvolver algoritmos para realizar requisições de informações e receber as respostas para essas requisições em um formato padronizado.

2.2.1 Twitter REST API

A *Twitter REST API*, fornece ferramentas para recuperação de postagens recentes de usuários a partir de requisições HTTP acrescido do parâmetro de busca. Com esta *API* é possível realizar buscas nos *tweets* que já foram publicados na plataforma. A principal limitação desta *API* é que ela restringe a quantidade de buscas pela complexidade e a frequência das mesmas. Além disso, esta *API* faz buscas no passado, com um período máximo aproximado de uma semana, e os retornos são filtrados para as mensagens de maior relevância, ou seja, não é possível buscar os *tweets* em sua totalidade por meio dessa interface [16].

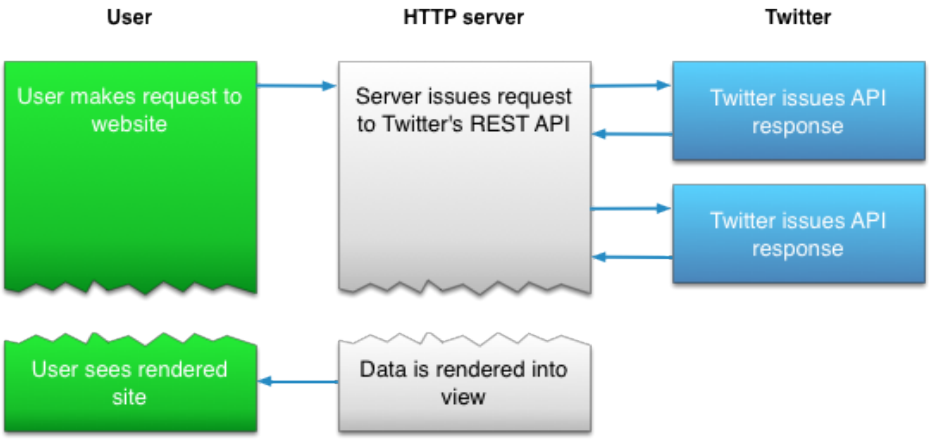

Figura 2.8: Funcionamento da Twitter REST API [16].

Anteriormente, como as solicitações eram anônimas, o limite da taxa desta API era medido pelo IP do cliente solicitante que, ao atingir este limite, passava a receber um erro informando que a taxa limite de buscas foi atingida. Atualmente é necessária a autenticação de usuário, ou seja, é necessário ser membro do serviço para utilizar esta API. Quando o limite de requisições é atingido, novas requisições são bloqueadas por um período de tempo.

A API REST acessa os mesmos recursos disponíveis de um usuário do site como: postar novos *tweets*, acessar *tweets* postados, pesquisar *tweets*, entre outros.

2.2.2 Twitter Streaming API

A *Twitter Streaming API* fornece ferramentas para buscas atualizadas em tempo real e o acesso ao maior número de mensagens possíveis, porém, é também necessária a autenticação como um usuário no *Twitter* para ter acesso aos seus métodos de busca. Assim como na primeira *API*, a busca na *API* de *Streaming* possui limitações, as principais são a proibição de mais de 6 conexões simultâneas por um mesmo usuário e o bloqueio do usuário e do IP caso o *Twitter* considere excessivo o número de tentativas de conexão realizadas em um espaço de tempo determinado [16].

Sua grande vantagem é que muitos dos *tweets* são coletados. Assim, a utilização da *API* de *Streaming* é vantajosa sobre a *API REST* por permitir um número mais alto de *tweets* por requisição. A *API* de *Streaming* não admite o retorno de *tweets* que foram

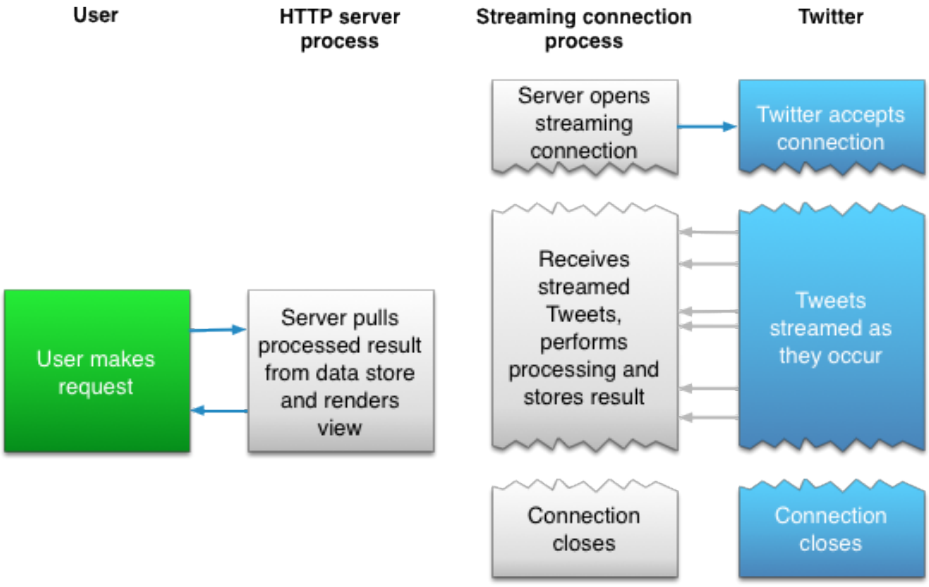

Figura 2.9: Funcionamento da Twitter Streaming API [16].

criados em um momento anterior, mas sim permite acesso aos *tweets* criados durante a conexão [16].

Como o trabalho baseia-se em respostas rápidas, sobre eventos sismológicos acontecidos escolhemos a *API* de *Streaming*, capaz de proporcionar uma resposta rápida e além disso permitir que uma quantidade maior de *tweets* seja observada.

A *API* de *Streaming* do *Twitter* é encontrada em três variantes, que monitoram funções diferentes dentro do serviço [16]:

- ***Public Streams*:** permite fazer o monitoramento de assuntos específicos, através de palavras-chave, utilizando o banco de dados global de mensagens públicas. É possível facilmente obter todos os *tweets* de usuários que utilizaram uma palavra-chave específica em qualquer região do planeta.
- ***User Streams*:** disponibiliza o monitoramento de atividade de um usuário específico desta rede, retornando todas suas atualizações.
- ***Site Streams*:** permite o monitoramento em massa, de vários usuários simultaneamente.

2.3 Descoberta de conhecimento

Com a grande quantidade de informação sendo gerada de forma exponencial, esta informação pode ser uma grande aliada no sentido de tomada de decisão entre as pessoas e a comunidade onde vivem, se tal informação for tratada. Quando os dados, que podem ser considerados como simples fatos, são estruturados eles tornam-se informação. A informação torna-se conhecimento quando é interpretada seja quando é acrescentado um significado à ela ou quando ela é inserida num contexto. É possível, a partir disso, afirmar

que os dados são uma condição necessária para a informação, e a informação é obrigatória para a geração de conhecimento.

A informação pode ser encontrada em três estados:

- **Estruturada:** onde cada campo possui a identificação da informação (Planilha de textos, Bancos de Dados);
- **Semi-estruturada:** possui *tags* que possibilitam a marcação das informações(XML);
- **Não estruturada:** são textos em linguagem natural. As informações não estruturadas podem ser encontradas em artigos, atas, sites, e-mails, ou seja, qualquer documento escrito em linguagem natural.

O aumento da quantidade de informação gerada está cada vez mais em foco, estimulando diversos estudiosos que buscam analisar este acontecimento e também encontrar meios de retirar conhecimento de toda esta informação. "A velocidade e a amplitude com que o conhecimento gerado passou a ser compartilhado provocaram o surgimento de uma dinâmica de reaproveitamento e produção de novos conhecimentos, bem como o aparecimento de novas necessidades de tratar a informação" [49].

A necessidade de uma análise mais precisa da informação produzida fez com que surgisse o conceito de Descoberta de conhecimento e os processos que possam conduzir a isso. Tais processos evidenciam informações que provavelmente não seriam observadas sem a utilização dos mesmos.

A descoberta de conhecimento se dá após a análise de dados com o objetivo primordial de que as pessoas e organizações adquiram novos conhecimentos após lidar com um alto fluxo de dados [46].

De acordo Wives[64], o processo de Descoberta de conhecimento se caracteriza em identificar, receber informações e representá-las de maneira a torná-las significativas ao usuário, com o objetivo de agregar conhecimento e transformar seu padrão de ciência.

Os processos de Descoberta de conhecimento são compostos por várias etapas, onde cada etapa possui várias funções a serem executadas. Uma função é solucionada por meio da escolha de uma estratégia de resolução. As estratégias de resolução utilizam algoritmos, podendo existir mais de um algoritmo possível de ser aplicado a uma determinada estratégia.

Desta maneira, a importância do uso de softwares e computadores com o intuito de tornar o processo automatizado, é evidente para que assim haja colaboração na busca pelo conhecimento de forma mais prática e eficiente.

A descoberta de conhecimento pode ser dividida em duas vertentes: *Knowledge Discovery in Databases*(KDD), Descoberta de conhecimento em bases de dados, e *Knowledge Discovery in Textual Databases*(KDT), Descoberta de conhecimento em texto [46]. Esta separação tem como base o conteúdo que será analisado. Se o conteúdo foi previamente organizado e estruturado, então o processo de descoberta a ser usado será o KDD. Caso o conteúdo encontre-se difundido em documentos textuais, o processo utilizado será o KDT [49].

2.3.1 Descoberta de Conhecimento em Bases de Dados

No final da década de 1980, surge uma nova linha de pesquisa que busca extrair conhecimento de bancos de dados, de forma computacional com do uso de ferramentas de KDD.

KDD é definido como uma maneira não usual de identificar padrões válidos que são anônimos inicialmente, embora com grande potencial de serem proveitosos a partir de uma base de dados [31]. Este conceito é realçado ao dizer que KDD foi projetado para extrair informações importantes aos gestores de informação que não podem ser reveladas de maneira eficaz a partir de relatórios e consultas [36].

"Um padrão que é interessante (de acordo uma medida de interesse do usuário) e determinante o suficiente (mais uma vez de acordo com critérios do usuário) é chamado de conhecimento. A saída de um programa que monitora o conjunto de fatos em um banco de dados e produz padrões neste sentido, é a descoberta de conhecimento" [31].

Segundo alguns autores, o elemento que potencializa o desempenho das operações de Descoberta de conhecimento em grandes conjuntos de dados persistentes é o componente de bancos de dados do KDD. Além disso, a descoberta de conhecimento é simplesmente a aprendizagem de máquina a partir desses conjuntos. Porém, mesmo com a importância do componente de dados KDD e a melhora do desempenho deste, ele individualmente não é suficiente para estimular uma mudança qualitativa na capacidade de um sistema [39].

Com a grande movimentação de dados que tem sobrecarregado as pessoas e comunidades, a exploração de ferramentas de KDD se torna uma área de pesquisa promissora, pois a quantidade de dados é praticamente imensurável e a análise destes dados com o propósito de extrair informações significativas em tempo suficiente é um ponto que não pode ser solucionado sem apoio computacional. Porém, sem a ajuda de humanos para interpretar corretamente os resultados, o KDD se torna desnecessário, pois o grande desafio encontrado na descoberta de conhecimento em base de dados é processar uma quantidade abundante de dados brutos e apresentá-los ao usuário em forma de conhecimento [44].

Em consequência do rápido aumento do volume de dados, nos últimos anos surgiram muitos trabalhos em torno de aplicações de KDD, aumentando o interesse na área e podendo impulsionar ainda mais uma nova geração de instrumentos e teorias computacionais que dêem suporte na extração de informação a partir de dados brutos [31].

2.3.2 Fases do Processo de Descoberta de Conhecimento em Bases de Dados - KDD

O processo de KDD é dividido em cinco etapas [39]. São elas: seleção, pré-processamento, transformação, mineração de dados e interpretação/avaliação do conhecimento extraído. A interação humana nas etapas do processo é necessária devido à relevância da orientação na execução do processo por um gestor que possua ciência acerca da esfera do problema tratado [25].

O KDD é um processo iterativo onde todas as fases são fundamentais para alcançar a finalidade [59]. Usualmente as fases do processo de KDD são:

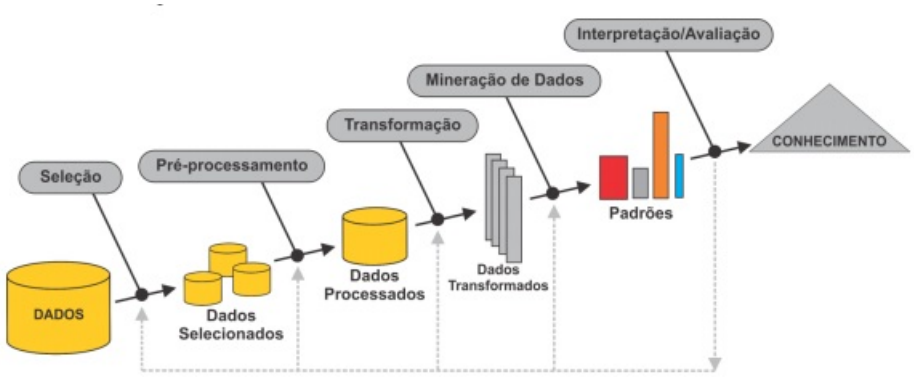

Figura 2.10: Fases do processo de KDD [35].

Seleção de Dados

Nesta fase são selecionados os dados relacionados a área do problema. Aqui fica evidente a exigência da compreensão da esfera do problema e dos objetivos [59]. Este processo de seleção é realizado utilizando-se um banco de dados estruturado.

Pré-processamento

Esta etapa destina-se a "eliminar os dados incompletos, problemas de definição de tipos, eliminação de tuplas repetidas, etc" [20]. Resumidamente esta etapa efetua pequenas correções e executa uma varredura no banco de dados com o intuito de garantir a consistência e a remoção de dados irrelevantes.

São propostas duas classificações para as tarefas realizadas na fase de pré-processamento dos dados [22]. São elas:

- **Tarefas fortemente dependentes de conhecimento de domínio:** são aquelas que somente podem ser efetivamente realizadas com o uso de conhecimento específico do domínio tratado.
- **Tarefas fracamente dependentes de conhecimento de domínio:** as informações necessárias para tratar os problemas de pré-processamento dos dados podem ser extraídas dos próprios dados, apresentando um maior grau de automação.

Transformação

Depois do pré-processamento, a etapa de transformação é encarregada de realizar a persistência dos dados tratados, fazendo com que estes estejam prontos para a mineração de dados. A transformação está profundamente ligada à técnica de mineração de dados. O objetivo primordial dessa fase é "... facilitar a utilização das técnicas de mineração de dados" [20].

Mineração de dados

Também conhecida por *Data Mining*, esta fase busca por padrões através de tarefas como: regras de classificação ou árvores, regressão, agrupamento, entre outras [31]. Essas

buscas e análises por padrões geralmente são executadas sobre um grande volume de dados pré-processados e armazenados por *Data Warehouse/Data Marts* [35].

A mineração de dados é fracionada de acordo com os tipos de dados e objetivos em cinco grupos [36]. São eles: associação, sequência, classificação, *clusterização* e previsão. As associações procuram descobrir relações em determinados conjuntos de dados. Por exemplo, calibrar o pneu associa-se a trocar o óleo, encher o tanque, entre outros, que em alguns casos pode parecer óbvio, mas em outros nem tanto. As sequências identificam acontecimentos que levam a outros. Por exemplo, a compra de uma casa leva a compra de móveis. Embora as classificações e *clusterizações* tenham objetivos semelhantes, pois ambas repartem os dados em grupos com indivíduos similares através de reconhecimento de padrões, elas se diferenciam no instante em que a classificação é feita seguindo amostras das classes já oferecidas em um momento passado, enquanto que a *clusterização* é manejada a partir de dados desconhecidos inseridos no algoritmo com a finalidade que este reconheça padrões e extraia informações necessárias para desmembrar grupos de dados(*clusters*). Por fim, as previsões consideram uma série temporal e predizem os dados futuros seguindo as informações extraídas do conjunto.

Interpretação/Avaliação

Esta fase apresenta o resultado da descoberta de conhecimento para o usuário através da representação e visualização do conhecimento obtido durante o processo. "Os resultados do processo de descoberta de conhecimento podem ser mostrados de diversas formas, porém devem ser apresentados de forma que o usuário possa entender e interpretar os resultados" [20].

Através da Figura 2.10, é possível enxergar que *Data Mining* ou Mineração de dados é uma das fases do processo de KDD, talvez a mais importante, dado que é nessa fase que as informações são extraídas de fato. Entretanto, um cuidado especial deve ser tomado na fase de pré-processamento dos dados, que procura identificar e remover os problemas que normalmente são visualizados nos dados extraídos de bancos de dados reais, como grande quantidade de inconsistências, excesso de valores desconhecidos, dados irrelevantes, entre outros, pois a qualidade dos dados fornecidos como entrada está diretamente relacionada à qualidade do conhecimento extraído no processo de descoberta de conhecimento [22].

Assim, a Mineração de Dados em suas variadas classificações, pode ser usada para descoberta de conhecimento em inúmeras áreas, cooperando para as pesquisas científicas e metodológicas como também vem se mostrando uma ferramenta importante para obtenção de resultados satisfatórios nos diversos setores de aplicação possíveis [21] [19].

2.3.3 Descoberta de Conhecimento em Bases Textuais

O processo de KDD é um grande aliado na busca por conhecimento. Porém as ferramentas deste processo não são capazes de extrair conhecimento de informações não estruturadas. As informações não estruturadas possuem um potencial enorme, pois a maioria das informações na atualidade se encontram nesse formato.

Assim, surge de forma concomitante à web o processo de KDT, como consequência do alto índice de dados dispostos armazenados e também a necessidade em recuperar conteúdos relevantes no formato desses dados não estruturados.

A Mineração de textos, ou *Text Mining*, é considerada sinônimo de descoberta de conhecimentos em textos e assim algumas nomenclaturas podem ser encontradas na literatura [42], outras variantes são, Mineração de Dados em Textos (*Text Data Mining*) ou Descoberta de Conhecimento a partir de Bancos de Dados Textuais, tendo em vista que o processo de mineração de textos também pode ser empreendido com técnicas de Descoberta de Conhecimento em Bancos de Dados [58].

A Mineração de textos tenta recolher informações significativas a partir de uma grande quantidade de textos escritos em linguagem natural. Mineração de texto pode também ser definido como a aplicação de algoritmos e métodos de aprendizagem de máquina e estatísticos para textos como o objetivo de encontrar padrões úteis. Assim é demonstrada a necessidade de pré-processar os textos de uma maneira eficiente [38].

O KDT "baseia-se em técnicas específicas para tratamento de textos que devem ser utilizadas a fim de se obter conhecimentos implícitos em bancos de dados textuais" [20]. A KDT é um processo pertencente à uma área multidisciplinar, que envolve recuperação de informação, análises textuais, extração de informação, *clusterização*, categorização, visualização e mineração de dados [46].

Os documentos não estruturados são compostos puramente por texto, encontrados em linguagem natural, de forma não organizada, o que dificulta a extração de informações. Embora estas informações sejam de fácil entendimento por humanos, do ponto de vista computacional é difícil de serem interpretadas, considerando-se que o computador não encontra indicadores explícitos sobre o que o texto está se referindo.

KDT é considerado o conjunto de técnicas inteligentes e intuitivas que colaboram para análise de altos volumes de textos para extrair conhecimento útil e auxiliar os usuários de textos não estruturados em sua diversidade [23].

As informações textuais não estruturadas, que têm aumentado significativamente na Internet ou dispositivos pessoais, compreendem informações não muito acessíveis, que por sua vez, podem apresentar-se em um caráter valioso para o usuário. Dessa forma, com a necessidade de extração dessas informações foram criados métodos de descoberta de textos.

É importante notar que a massa de dados disponível se encontra sob a forma textual, o que evidencia ainda mais a relevância das técnicas de descoberta de conhecimento não estruturado.

Os textos normalmente apresentam mais dificuldade na extração de informações ao serem comparados a dados armazenados em bancos de dados, por estarem menos(ou de forma nenhuma) estruturados e mais complexos de serem pré-processados para a inserção nos algoritmos de mineração. Na atual conjuntura, é necessário lidar com dados em forma de texto, já que este é o veículo comumente utilizado para a troca de informações, tornando este campo de pesquisa muito interessante [31].

2.3.4 Fases do Processo de Descoberta de Conhecimento em Bases de Textos - KDT

O processo de KDT é semelhante ao KDD nas fases de mineração e interpretação/avaliação. Entretanto, os passos do KDT possuem pequenas adaptações para que possa ser aplicado em informações não estruturadas.

A grande diferença entre Mineração de dados e Mineração de textos não é apenas a fonte da informação, mas a característica de que na Mineração de dados, a informação extraída é desconhecida e não é explícita [63], enquanto que na Mineração de textos, a informação a ser extraída está explicitada no texto a ser minerado, porém a dificuldade está no volume de dados, pois com a grande quantidade de textos torna-se impraticável minerar estes de maneira não computacional.

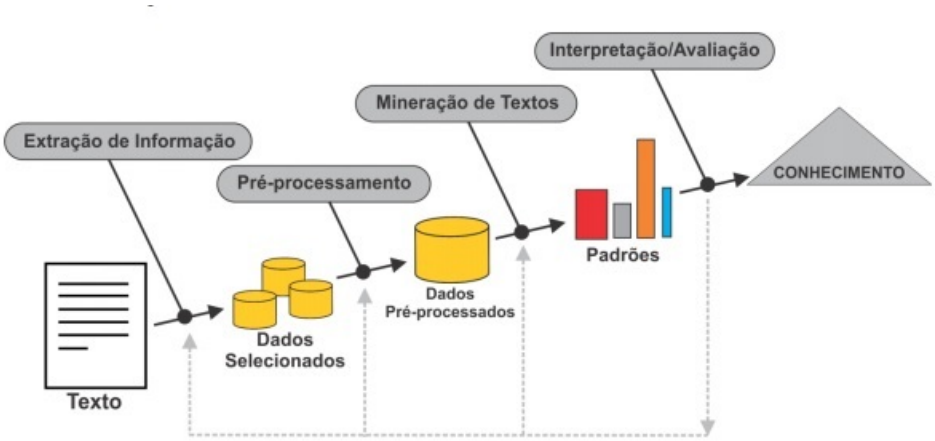

Figura 2.11: Fases do processo de KDT [35].

Extração de Informação

A etapa de extração se baseia na aquisição dos textos que serão usados no processo de mineração. Esta fase se constitui na escolha de preferência do conjunto de dados a serem analisados por um critério estabelecido previamente. "Considerando os objetivos que se deseja alcançar com o processo, o primeiro passo é eleger o conjunto de textos que será utilizado"[26].

Diversos documentos são analisados para que sejam coletados os dados que se unirão e formarão um *corpus*; esses dados podem ser extraídos do disco rígido de um computador, de bancos de dados, da web, etc. Esta etapa é crucial para o processo pois os dados refinados serão oriundos dos dados coletados neste momento. Neste trabalho será utilizada como fonte a web, que concerne a maior base de dados heterogêneos encontrada, sendo também uma base de dados complexa e consequentemente de difícil manejo.

Pré-processamento

"O objetivo dessa fase é eliminação de termos não relevantes(*stop-words*), redução das palavras aos seus radicais (*stemming*, correções ortográficas e outros aspectos morfológicos e também sintáticos que as expressões textuais possuem.)" [26].

Nesta etapa os dados não estruturados serão transformados em dados estruturados. Essa etapa atua como se fosse uma limpeza na fonte de dados para adaptá-la às etapas seguintes.

A reestruturação resultante do texto permitirá que o mesmo seja trabalhado pelo computador, e esta etapa pode ser realizada por diversas técnicas, onde elas se diferem

pela conveniência de ser mais útil em um determinado contexto. Essas técnicas podem ser usadas de maneira individual ou associada.

Dessa maneira, o Processamento da Linguagem Natural(PLN) é essencial nesta fase, pois examina justamente as estruturas implícitas, tendo como exemplo, a estrutura sintática [48]. No pré-processamento ocorre a transformação do texto em vetores, tabelas, matrizes, etc [26]. Neste momento a informação já se encontra de forma estruturada, e pronta para a fase de mineração.

A transformação de dados é vista como uma etapa no processo de KDD ilustrado na Figura 2.10, enquanto que no processo de KDT a transformação de dados está implícita na etapa de pré-processamento, na proposta de Mooney e Nahm, ilustrado na Figura 2.11.

Essa fase é uma das principais em todo o processo de mineração de textos [40], considerando que uma abstração ruim do documento pode impedir que este seja analisado corretamente.

Na etapa de pré-processamento, existem alguns procedimentos ao qual os documentos são submetidos. São eles:

- **Tokenização:** Refere-se à dispersão de unidades que formam o texto em unidades individuais mínimas – os *tokens* – que tomam forma e valor.
- **Correção Ortográfica:** É comum encontrar erros ortográficos em documentos textuais, devido a esses documentos terem sido concebidos por digitação manual de seres humanos. Assim, se faz necessário que haja correções textuais, para melhor estruturação do texto e otimização de resultado final da descoberta de conhecimento.
- **Redução do léxico:** Nem todas as palavras do texto tem valor real [55]. Algumas podem estar repetidas e outras nem tem representação. É importante, então, diminuir o número de termos no texto, tendo o cuidado para que o mesmo mantenha o seu sentido original.

Como dificuldade, este processo de conversão de textos para informação estruturada pode promover custos adicionais ao processo de KDT em relação ao KDD.

Mineração

Nessa fase é aplicado o algoritmo para que possa ser extraído o conhecimento desejado, utilizando alguma abordagem que mais se adapta ao objetivo proposto. As abordagens são:

- Classificação: A classificação consiste em organizar os documentos em classes, seguindo alguma semelhança entre os documentos de cada classe.
- *Clusterização*: A *clusterização* é uma estratégia de agrupamento que procura por padrões similares nos documentos, agrupando os textos em categorias, sendo necessário, ao final, informar a quantidade de categorias presentes e sua relação com a coleção de documentos.
- Sumarização: A sumarização é realizada com o objetivo de resumir o conteúdo do texto sem que este perca seu significado.

- Análise: A última etapa é a análise, realizada por pessoas especializadas que interpretarão os resultados e utilizarão os mesmos de maneira tal que o objetivo de se realizar a Mineração de texto seja atingido.

2.3.5 Mineração na Web

O número de usuários de Internet ao redor do mundo tem aumentado consideravelmente ao longo dos anos. Em 1995, existiam cerca de 45,1 milhões de usuários, no ano 2000 esse número se expandiu para 420 milhões, em 2005 a barreira dos bilhões foi rompida com 1,08 bilhões de usuários e em setembro de 2009 já existiam 1,73 bilhões [29]. Esse número continuou a crescer e em 19 de junho de 2016 haviam aproximadamente 3,4 bilhões de usuários da Internet, segundo o site www.internetlivestats.com [47].

Atualmente o aumento constante do número de usuários fez com que a maior parte das aplicações e informações estejam disponíveis na web, assim, a tentativa de extração de conhecimento nesse meio se tornou muito interessante já que muitas destas informações podem ser acessadas de forma pública e se trata de um conjunto de dados bastante rico e diversificado. Dá-se o nome de Mineração na web (*Web Mining*) a tentativa de extração de conhecimento apoiado na Mineração de dados obtidos da web [41] [30].

Figura 2.12: Ideia Geral da Mineração na Web [6].

O processo de Mineração na Web pode ser dividido em três categorias que se subdividem em outras. Essas categorias são: *Web mining* de conteúdo, de estrutura e de uso.

- *Web mining* de conteúdo: Consiste em minerar o conteúdo extraído de textos, imagens, áudios, vídeos e registros estruturados como listas e tabelas. Empresas importantes do ramo de buscas na Internet executam esse tipo de mineração a fim de relacionar os parâmetros de busca dos usuários com o conteúdo extraído e tratado [61].
- *Web mining* de estrutura: Trata da tentativa de extração de informação baseada nas ligações entre as páginas através dos *hiperlinks* contidos nas mesmas, com a possibilidade de verificar a popularidade de uma página em um determinado contexto a partir da quantidade de referências dessa página por outras. Além disso outras características podem ser extraídas a partir da análise da estrutura de grafos da web, por exemplo a classificação por importância de páginas [60] [56].

- *Web mining* de uso: Visa utilizar técnicas de mineração para observar padrões de uso das aplicações na tentativa de antecipar o comportamento do usuário a medida que esse interage com a web, partindo dos dados gerados pelas transações cliente-servidor. As informações observadas podem ser aplicáveis desde a sugestão de melhorias na estrutura de um site(como o posicionamento de *hiperlinks*) a indicações de produtos ao usuário, tendo como base a frequência em que o mesmo realiza compras pela Internet e nos produtos que ele compra, por exemplo [65].

Capítulo 3

Desenvolvimento do TwitSisbra

Este capítulo apresenta o TwitSisbra, um sistema capaz de trazer informações sobre eventos sismológicos, conhecidos por terremotos ou tremores, divulgados por usuários da rede social *Twitter*. Este projeto foi desenvolvido de forma conjunta com SIS-UnB, com a finalidade de ajuntar notícias e sentimentos providos por usuários sobre desastres sísmicos e informar aos analistas do SIS, através de uma interface simples e amigável. Este capítulo está dividido nas seguintes seções: Contextualização e Implementação do TwitSisbra, esta última que se divide em Arquitetura do Sistema, Tecnologias Utilizadas, Funcionamento do Sistema, Interface e Emissão de alerta de desastre.

3.1 Contextualização

O Observatório Sismológico da Universidade de Brasília (SIS)[8] é responsável por coletar, armazenar, analisar e comunicar sobre os diversos tipos de eventos sismológicos no território brasileiro. Os dados são coletados através de instrumentos chamados sismógrafos espalhados em várias estações sismológicas pelo Brasil.

Para auxiliar nos processos de manuseio dos dados coletados, foi criado o WebSisbra. O WebSisbra[37] possui uma interface que auxilia na análise dos dados coletados, sendo uma ferramenta de apoio tanto para o pessoal capacitado para analisar esses dados, que são os analistas do SIS, quanto para pessoas leigas que acessam o Website do sistema.

As redes sociais permitiram um avanço na troca de informações de forma global. Muitos usuários compartilham notícias, sentimentos, acontecimentos, a todo momento em redes sociais. A velocidade em que a informação é repassada para outros usuários da rede é grande. Isto também pode ser observado ao se tratar de eventos sísmicos, como tremores de terra. Desta forma as informações das redes sociais podem chegar mais rápido do que informações coletadas oficialmente pelos sismógrafos mencionados. A rede social *Twitter* possui ferramentas que auxiliam a extração de dados, enviados pelos usuários dessa rede, de forma colaborativa.

Assim, o objetivo deste trabalho está em identificar os *tweets* - que são as postagens dos usuários dessa rede - que tratam de informações sobre desastres ou eventos sismológicos. Além disso, pretende-se trazer essas informações coletadas aos analistas e também aos leigos, de forma clara e visual. Por fim, este trabalho também trata da emissão de alertas ao pessoal autorizado do SIS, quando há a incidência de muitos *tweets* numa determinada região.

O sistema TwitSisbra proposto neste trabalho, foi projetado seguindo a arquitetura em camadas. As camadas abordadas foram: camada de interface, aplicação e persistência. Além disso foi utilizada a arquitetura cliente e servidor, onde as funcionalidades são divididas entre esses dois modulos. Mais detalhes da arquitetura abordada estão descritos na seção 3.2.1.

Para proceder à implementação deste trabalho foram utilizadas diversas ferramentas, que foram:

- Javascript;
- Node.js;
- Express;
- Socket.io;
- *Twitter streaming API*;
- Google Maps;
- HTML;
- CSS.

Essas ferramentas abrangem desde linguagens de programação até *frameworks* que permitem a transmissão de dados em tempo real. Cada uma delas é explanada na seção 3.3.2.

O funcionamento do TwitSisbra se dá basicamente seguindo as etapas do processo de mineração de textos. As etapas compreendidas foram:

- Coleta(ou Extração da informação);
- Pré-processamento;
- Mineração;
- Análise(Através da Interface).

As primeiras três etapas são responsabilidade do próprio sistema, enquanto que a última será executada pelos analistas do SIS. Cada uma dessas etapas possui suas responsabilidades descritas na seção 3.2.3, que trata do funcionamento do sistema.

Além da mineração de textos, a outra parte que sustenta o trabalho é a visualização dos dados coletados. Essa parte é compreendida pela Interface. A Interface do TwitSisbra é baseada na exibição dos *tweets* na forma de mapa de calor. A seção 3.2.4, trata da interface do TwitSisbra

A Análise será executada pelos responsáveis no SIS através da visualização da interface do TwitSisbra juntamente com os dados coletados oficialmente nos sismógrafos.

Por fim, este trabalho compreende a emissão de alertas de desastres quando houverem muitos usuários divulgando informações sobre terremotos, numa mesma região. As informações sobre a implementação deste alerta estão descritos na seção 3.2.5.

3.2 Implementação do TwitSisbra

3.2.1 Arquitetura do Sistema

A arquitetura abstrata proposta para o SIG Twitsisbra, é ilustrada na Figura 3.1, sendo composta por três grandes módulos: Camada de interface, responsável pela interação do sistema com o usuário através dos mapas de calor; A Camada de aplicação é responsável pelo mecanismo de geração dos mapas; e por último a camada de persistência, onde os dados coletados são tratados e armazenados em um banco de dados.

Figura 3.1: Arquitetura abstrata do TwitSisbra.

A camada de Interface compreende a exibição dos *tweets* coletados em suas localizações obtidas ou o mais próximo possível de sua localização de origem. A camada de Aplicação é dividida em duas partes: o lado servidor(*Server-side*) e o lado cliente(*Client-side*). O lado servidor é responsável por carregar as páginas da web, configurar e inicializar as ferramentas utilizadas na implementação do TwitSisbra. Já o lado cliente configura o mapa e abre uma conexão com o servidor. Mais detalhes de como cada um dos lados da camada de aplicação funcionam são dados na seção 3.2.3. A Camada de Persistência é encarregada de armazenar e consultar os dados do sistema. Os *tweets* com sua localização são obtidos no formato JSON.

Além de exibir um mapa de calor com a localização dos eventos sísmicos detectados pelo *Twitter*, o TwitSisbra também tem a função de notificar os administradores do Observatório Sismológico de Brasília quanto à ocorrência desses fenômenos. Essa função está especificada na camada de persistência na Figura 3.2.

A Figura 3.2 compreende as camadas da arquitetura citada e suas responsabilidades. A camada de aplicação inicia o sistema através da inicialização dos ouvintes. O primeiro ouvinte é inicializado pelo lado cliente quando este se conecta com o servidor, sendo utilizado para estabelecer a conexão com o *Twitter*. Outro ouvinte é inicializado pelo servidor e uma mensagem é enviada deste ao cliente confirmando que a conexão foi estabelecida. A mensagem recebida pelo cliente indica que o servidor e cliente estão conectados.

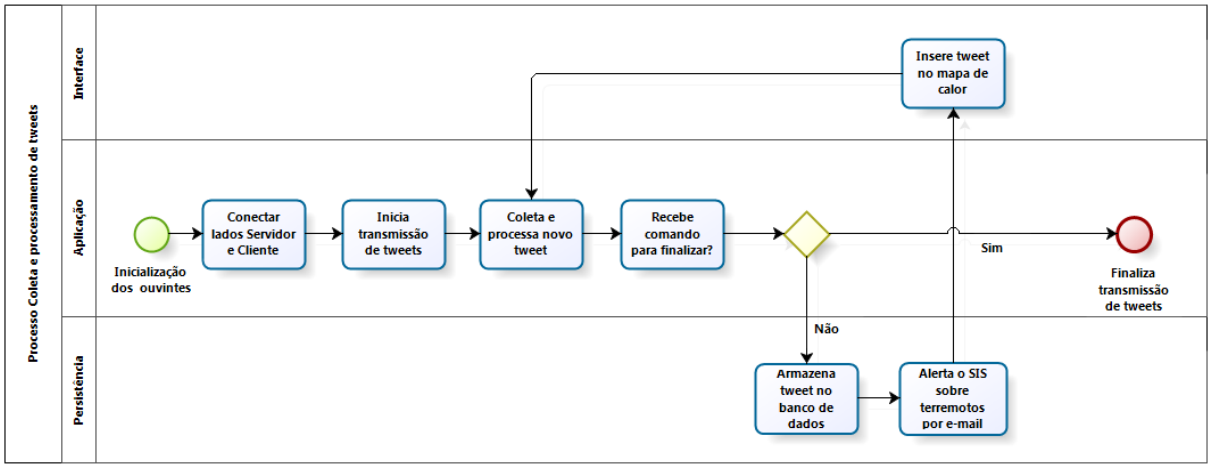

Figura 3.2: Processo designado para o TwitSisbra.

Quando os dois lados da aplicação estão conectados, é possível a transmissão dos *tweets*. Neste momento, as outras duas camadas são acionadas. Por um lado a camada de interface recebe os *tweets* desejados e sua localização aproximada, e insere um ponto no mapa de calor correspondente à sua localização.

De forma simultânea, a camada de persistência age recebendo os *tweets* e armazenado no banco de dados. Além disso, essa camada é responsável por testar se a cada *tweet* recebido, existem outros *tweets* com informações semelhantes na mesma região, no sentido de criar um alerta de desastres sismológicos.

Por fim, a Figura 3.2 retrata como o programa encerrará ao receber o comando específico para finalização.

3.2.2 Tecnologias Utilizadas

Javascript

Javascript é uma linguagem de alto nível, dinâmica, não-tipada e interpretada[34]. Juntamente com o HTML e o CSS, é uma das tecnologias centrais da produção de conteúdo da Internet, com a maioria dos *websites* empregando essa tecnologia e sendo suportada pelos navegdadores de Internet modernos[34]. Javascript é considerada uma linguagem multi-paradigmas, suportando estilos de programação orientados a objeto, imperativos e funcionais. Possui *API* para trabalhar com texto, vetores, datas e expressões regulares.

Embora haja muitas semelhanças entre JavaScript e Java, como o nome da linguagem, a sintaxe, e as bibliotecas padrão semelhantes, essas duas linguagens são diferentes[5]. O núcleo do JavaScript pode ser estendido para uma variedade de propósitos, complementando a linguagem:

- O lado cliente do JavaScript estende-se do núcleo da linguagem, fornecendo objetos para controlar um navegador web e seu *Document Object Model*(DOM). O DOM é uma multi-plataforma que denota como as marcações em HTML, e outras linguagens de marcação são organizadas e lidas pelo navegador utilizado. Como exemplo tem-se que extensões do lado do cliente permitem que uma aplicação coloque elementos

em um formulário HTML e responda a eventos do usuário, como cliques do mouse, entrada de formulário e de navegação de página;

- O lado servidor do Javascript se estende do núcleo da linguagem, fornecendo objetos relevantes à execução do JavaScript em um servidor. Por exemplo, as extensões do lado servidor permitem que uma aplicação comunique-se com um banco de dados, garantindo a continuidade de informações de uma chamada para outra da aplicação, ou executar manipulações de arquivos em um servidor.

Node.js

Node.js é um ambiente de execução *open source* e multi-plataforma para o desenvolvimento de aplicações web do lado servidor. Muitos dos módulos básicos do Node.js são escritos em JavaScript e os desenvolvedores podem criar novos módulos em JavaScript. O ambiente de execução interpreta JavaScript utilizando o motor(*engine*) JavaScript V8 do Google[7].

O motor JavaScript V8 é um mecanismo do JavaScript que o Google utiliza em conjunto ao navegador Chrome[7]. Com o V8, o Google criou um interpretador muito rápido escrito em C++ que possibilita fazer o *download* do motor e integrá-lo em qualquer aplicativo desejado.

Node.js permite a criação de serviços web e ferramentas de rede usando JavaScript e uma coleção de módulos que lidam com várias funcionalidades do núcleo. Os módulos são fornecidos para o sistema de arquivos, entrada e saída , rede, fluxos de dados e outras funções essenciais. Os módulos do Node.js utilizam uma *API* concebida para reduzir a complexidade de escrever aplicações de servidor[45].

Aplicações do Node.js podem ser executadas nos principais sistemas operacionais do mercado que são Mac OS X, Microsoft Windows, e servidores Unix. Essas aplicações podem, alternativamente, ser escritas com alternativas fortemente tipadas do JavaScript, ou qualquer outra linguagem que possa compilar para JavaScript[24].

Express

Express.js é um *framework* web para Node.js. Ele é usado para facilitar a criação de aplicações e serviços web através de vários recursos disponíveis para o desenvolvimento[3]. Express possibilita um desenvolvimento rápido de aplicações web baseadas no Node.js[4]. A seguir estão algumas das principais características do *framework* Express:

- Permite a criação de mecanismos para responder às solicitações HTTP;
- Permite processar páginas HTML dinamicamente com base na passagem de argumentos para modelos.

Socket.io

Socket.IO é uma biblioteca JavaScript para aplicações web em tempo real. Ela permite em tempo real, a comunicação bidirecional entre clientes e servidores [13]. Pode ser dividida em duas partes: uma biblioteca do lado do cliente que é executada no navegador, e uma biblioteca do lado do servidor para o Node.js. As duas partes possuem *APIs* muito semelhantes. Como o Node.js, Socket.io é orientada a eventos.

A principal ideia por trás do Socket.io é que seja possível enviar e receber qualquer evento e qualquer dado desejado. É possível assim enviar qualquer objeto que possa ser convertido para JSON, ou dados binários, que também são suportados.

O Socket.IO inicia uma con exão com o servidor possibilitando assim realizar a troca de mensagens entre cliente e servidor, sem a necessidade de atualizar(*refresh*) a página. Esta característica dispensa a necessidade de solicitar uma nova requisição para o servidor. Com isso é possível explorar o conceito de tempo real em uma aplicação, pelo qual basicamente o cliente envia uma mensagem, o servidor processa e responde utilizando *broadcast*(para todos os clientes de conexão aberta com servidor)[9].

Twitter streaming API

A *API* de *streaming*(*Twitter Streaming API*) fornece aos desenvolvedores acesso de baixa latência para o fluxo global dos dados dos *tweets* a partir do *Twitter*. Serão transmitidas mensagens com os *tweets* e a ocorrência de outros eventos à aplicação de um cliente de *streaming*, sem que haja com isso uma sobrecarga associada à emissão num terminal *REST*.

Há três variações da *API* de *streaming* do *Twitter*[17]:

- ***Stream* Público:** permite que sua aplicação monitore dados públicos no *Twitter*, como *tweets* públicos, filtros de tags, etc.;
- ***Stream* do Usuário:** permite rastrear o fluxo de *tweets* de um usuário, em tempo real;
- ***Stream* de Site:** permitem que sua aplicação monitore os *feeds* de vários usuários do *Twitter*, em tempo real.

Para usar a *API* de *streaming*, uma aplicação faz uma solicitação persistente HTTP. Ao contrário de uma requisição da *API* convencional *REST*, onde a conexão com o servidor é encerrada logo após os dados serem recebidos, a *API* de *streaming* deixa a conexão aberta pelo maior tempo possível e transmite continuamente novos dados à medida em que eles estiverem disponíveis. Os dados são enviados como *BLOB* no formato JSON que descrevem mensagens e eventos, tais como *retweets* e eliminação de mensagem. A estrutura dos dados da mensagem que é emitida pela *API* de *streaming* corresponde ao da *API REST*, o que significa que os desenvolvedores de aplicativos que estiverem usando o formato de saída JSON podem reutilizar o seu código existente de análise de mensagem em ambas *APIs*[17].

Embora o *Twitter* disponibilize gratuitamente informações dos *tweets*, para acessar a *API* de *streaming*, é necessário obter 4 credenciais de acesso[14] ao *Twitter* que são:

- *API key*;
- *API secret*;
- *Access token*;
- *Access token secret*.

Para isso é necessário ser cadastrado no *Twitter*, visitar a página https://apps.twitter.com/ e criar um novo aplicativo.

```
#Variables that contains the user credentials to access Twitter API
access_token = "ENTER YOUR ACCESS TOKEN"
access_token_secret = "ENTER YOUR ACCESS TOKEN SECRET"
consumer_key = "ENTER YOUR API KEY"
consumer_secret = "ENTER YOUR API SECRET"
```

Figura 3.3: Como as credenciais do aplicativo ficarão no código da aplicação.

HTML

HyperText Markup Language (HTML), é a linguagem de marcação padrão usada para criar páginas web. Junto com CSS e JavaScript, HTML é uma tecnologia fundamental usada para criar páginas web, bem como para criar interfaces de usuário para aplicações móveis e da web. Os navegadores da Web podem ler arquivos HTML e torná-los em páginas web visíveis ou audíveis. HTML descreve a estrutura de um site de forma semântica e, inicialmente, incluía sugestões para a apresentação ou aparência do documento (página da web), tornando-se uma linguagem de marcação, em vez de uma linguagem de programação.

Os elementos HTML formam os blocos de construção de páginas. Essa permite que imagens e outros objetos possam ser incorporados e possam ser utilizados para criar formas interativas de páginas web. Essa linguagem fornece um meio para criar documentos estruturados denotando a semântica estrutural para o texto, como cabeçalhos, parágrafos, listas, links, citações e outros itens. Os elementos HTML são delineados por *tags*, escritas usando colchetes. Os navegadores não exibem as *tags* HTML, mas as usam para interpretar o conteúdo das páginas.

HTML pode incorporar *scripts* escritos em linguagens como JavaScript que afetam o comportamento das páginas web HTML. A marcação HTML também pode consultar o navegador por CSS para definir a aparência e o *layout* de texto e outras informações.

CSS

Cascading Style Sheets (CSS) é uma linguagem de folha de estilo usada para descrever a apresentação de um documento escrito em uma linguagem de marcação. Junto com HTML e JavaScript, CSS é uma tecnologia fundamental usada pela maioria dos sites para criar páginas web mais atraentes das quais criadas utilizando-se somente HTML, interfaces de usuário para aplicações web e interfaces de usuário para muitas aplicações móveis.

CSS é projetado principalmente para permitir a separação do conteúdo do documento da apresentação do documento, incluindo aspectos como o *layout*, cores e fontes. Esta separação pode melhorar a acessibilidade dO conteúdo, fornecer mais flexibilidade e controle na especificação das características de apresentação, permitir múltiplas páginas HTML compartilhando uma única formatação através da especificação do CSS em questão em um arquivo *.css* separado, e reduzir a complexidade e repetição no conteúdo estrutural.

Figura 3.4: Processos compreendidos no sistema.

3.2.3 Funcionamento do Sistema

A Figura 3.4 relaciona como serão empregados os processos da mineração de texto. A seleção dos dados é responsável por coletar os dados do *Twitter*. Em seguida o pré-processamento remove do *corpus* coletado os dados irrelevantes. A mineração de texto é feita para se obter os dados sobre a localização de cada *tweet* selecionado. Por fim, a análise é realizada por pessoas, diferente das etapas anteriores que são executadas pelo sistema.

Inicialização do sistema

O lado Servidor é responsável por carregar uma página da Web utilizando o *framework Express* do Node.js, que alimenta as páginas estáticas da web, carrega o módulo de *socket* web Socket.io e aciona o módulo da *API* de *streaming* do *Twitter*. Os passos do lado servidor são:

- A aplicação de mapeamento do cliente se conecta ao servidor *socket* web e aciona um ouvinte chamado *connection*.
- Um outro ouvinte, chamado *start tweets*, é inicializado e uma mensagem de conectado é enviada ao cliente dizendo à ele que os dois lados estão conectados e os dois lados da aplicação estão conectados.
- Quando o cliente recebe essa mensagem, ele envia outra mensagem ao ouvinte *start tweets* e os *tweets* começam a ser transmitidos. Os *tweets* são coletados em formato JSON. Também é necessário conferir se a transmissão realmente está aberta, pois, não é interessante abrir uma transmissão por conexão, pois a *API* do *Twitter* restringe um número máximo de 6 conexões por usuário.

O lado cliente tem a função de configurar o Mapa do *Google* e então abrir uma conexão com o servidor. A partir do momento que o servidor confirma que ele recebeu a conexão e está pronto para enviar *tweets*, o ouvinte *connection* é chamado e o cliente envia uma mensagem de volta para o servidor para dizer que ele está pronto(através do ouvinte *start tweets*). Finalmente o servidor responde com uma transmissão de *tweets* capturados pelo ouvinte *twitter-stream*, onde estes *tweets* são incorporados a camada de calor do *Google Maps*.

Seleção dos Dados

A coleta dos dados do *Twitter* compreende a primeira parte da execução deste trabalho. Os dados são coletados com a *API* de *streaming* do *Twitter* através de funções compreendidas na própria *API*.

A *API* de *streaming* funciona fazendo um pedido para um tipo específico de dados - filtrada por palavra-chave, usuário, área geográfica, ou uma amostra aleatória - e, em

seguida, mantém a conexão aberta, desde que não haja erros na conexão. Neste trabalho foi utilizado o filtro por palavra-chave.

Uma vez conectado à *API* do *Twitter*, seja por meio da *API REST* ou da *API* de *streaming*, começará assim a entrada de uma grande quantidade de dados de retorno. Os dados que serão obtidos estarão codificados em JavaScript Object Notation (JSON). JSON é uma maneira de codificar a informação em uma forma independente de plataforma. Além disto JSON é uma maneira simplista e elegante de codificar estruturas complexas de dados. Quando um *tweet* retorna da *API*, a estrutura a seguir mostra o que este parece:

```
{
    ...
    "text": "TeeMinus24's Shirt of the Day is Palpatine/Vader '12.
    Support the Sith.",
    ...
    "retweeted": false,
    "coordinates": null,
   ...
    "user": {
        ...
        "location": null
        "description": "We are a limited edition t-shirt company. ",
    },
    "geo": null,
    "created_at": "Tue Mar 01 05:29:27 +0000 2016",
    "place": {
     ...
         "id": "5a110d312052166f",
       "full_name": "San Francisco, CA",
       "place_type": "city"",
            }
    ...
}
```

A coleta de dados através da *API* de *streaming* do *Twitter* retorna muitos dados, grande parte deles não são utilizados no caso de detecção de eventos sísmicos. O processo de coleta de dados no TwitSisbra é contínuo, pois busca-se receber os *tweets* sobre informações geológicas,como por exemplo, terremoto ou tremores à medida que eles vão acontecendo em tempo real.

Quando a conexão está ociosa e não há outros dados para enviar, a *API* de *streaming* irá emitir uma linha em branco a cada 30 segundos, tempo estipulado pela própria *API*. A linha em branco é um sinal de conexão ativa que se destina a impedir que aplicativos clientes dêem finalização por tempo limite excedido e terminem a conexão.

Pré-Processamento

Estabelecer a conexão e receber os dados é mais fácil do que o processamento em si. Inicialmente, o *tweet* tem uma estrutura complexa, com várias dados irrelevantes para

atingir o objetivo deste trabalho. Considerando isto, os *tweets* foram convertidos para um forma simplificada, contendo apenas os campos significativos para realizar o objetivo proposto neste trabalho.

Os dados relevantes são:

- **Text:**Contém até 140 caracteres de texto, e contém as mais diversas postagens e informações descritas pelos usuários dessa rede.
- **Coordinates:**Contém informações(Latitude e Longitude) sobre a localização do usuário que postou o *tweet*. Não é um campo obrigatório.
- **Place:**Diz respeito à cidade do usuário que postou o *tweet*. Também não é obrigatório.
- **Created at:**Contém a data e hora no qual o *tweet* foi postado.

Para o desenvolvimento do TwitSisbra após reuniões com especialista do SIS foram definidas um conjunto de palavras-chaves relacionadas aos terremotos e tremores sentidos pela população, entretanto muitas pessoas relatam esses eventos como explosões, detonações, entre outros.

As palavras chave escolhidas para detectar desastres sismólogicos foram:

- Terremoto;
- Tremor;
- Abalo;
- Tremeu;
- Explosão;
- Detonação;
- Sismo;
- Estrondo.

Através da diretiva *statuses/filter* é possível filtrar os *tweets* pelas palavras que designam os fenômenos sísmicos. Esta diretiva retorna *tweets* públicos que correspondem a um ou mais parâmetros de filtro. De acordo o *Twitter*, vários parâmetros de filtro podem ser especificados, permitindo assim que a maioria dos clientes possam usar uma única conexão com a *API* de *streaming*. Entretanto, só puderam ser observados vários parâmetros de busca em uma mesma categoria, por exemplo Palavra-chave, ao invés de poder combinar várias categorias, como Local e Língua de Origem, por exemplo.

Inicialmente gostaríamos de filtrar os *tweets* que contivessem alguma das palavras-chave propostas e também aqueles que estivessem dentro de uma determinada localização geográfica. Entretanto, seria necessário que houvessem mais conexões sobre a *API* de *streaming*, contrariando um dos princípios do *Twitter*. Assim, neste trabalho o método utilizado para filtrar os tweets foi através das palavras-chave propostas pelos especialistas do SIS.

Mineração

Depois de filtrar os *tweets* que contém as palavras-chave relacionadas à terremotos, a próxima etapa é detectar aonde esses *tweets* se localizam com o intuito de trazer a informação desses eventos aos responsáveis o quanto antes.

A localização do *tweet* é identificada de três formas:

- **Através das coordenadas cedidas pelo usuário:** O usuário pode habilitar a função de GPS no dispositivo que está sendo usado para enviar a mensagem. Postar um *tweet* com a localização é o recurso de etiquetagem geográfica na *API* do *Twitter*. Este recurso ajuda a fornecer uma experiência mais significativa para os usuários, fazendo os *tweets* serem mais contextualizados. Esta é a forma mais fácil e precisa de se obter a localização de um *tweet*, e no caso do *Twitter* como um sensor de eventos sísmicos, seria a forma ideal. Entretanto, esse método tem suas dificuldades. Os usuários têm de optar por usar o recurso *Tweeting With Location*(Postar *tweets* com Localização). Os usuários devem dar permissão explícita para a sua localização exata a ser exibida com seus *tweets*. Infelizmente, não são muitos os usuários que utilizam esse recurso. Aproximadamente, 20% dos usuários permitem que o *Twitter* tenha acesso a sua posição geográfica [15]. Como este é um dos métodos empregados neste trabalho é preciso lidar com essa dificuldade sabendo que esta é a forma mais precisa de se obter a localização de um *tweet*.

- **Através da cidade do usuário:** Embora não tão precisa quanto à primeira, essa forma de localização é muito boa, pois é possível detectar os *tweets* de uma forma geral nas cidades, abrangendo cerca de 50% dos tweets. Através do campo *place* é possível detectar a localização do *tweet* com uma precisão um pouco menor do que a técnica anterior .

  Places são locais nomeados com coordenadas geográficas específicas. Eles podem ser ligados aos *tweets* especificando um *place_ id* ao postar um *tweets*. *Tweets* associados com lugares não são, necessariamente, enviados a partir desse local, mas é bem provável que possa ser sobre esse local. *Places* também têm vários campos atributos que descrevem ainda mais um lugar. Neste trabalho utiliza-se apenas o atributo *name*, que retrata a cidade do usuário que enviou o *tweet*. Esse método, embora menos preciso, abrange muito mais *tweets* e suas localizações geográficas.

- **Minerando o texto do *tweet* em busca da cidade:** Ao postar alguma informação sobre algum desastre sentido, o usuário do *Twitter* provavelmente irá mencionar o local do ocorrido, como por exemplo, "Senti um tremor aqui no centro de Montes Claros", ou "A terra tremeu na cidade de Itumbiara, Goiás". De forma semelhante a quando usuário informa sua cidade na conta de usuário do *twitter* precisamos usar um banco de dados com as cidades brasileiras e suas latitudes e longitudes.

  O banco de dados das cidades demonstrado na Figura 3.5, é composto de três informações para encontrar a localização de um *tweet*. Essas informações são: a cidade, a latitude e a longitude da respectiva cidade.

  Quando um *tweet* com informação de evento sísmico acontece, identifica-se no conteúdo do texto daquela postagem se há alguma informação sobre a cidade. Esse método têm a capacidade de obter praticamente todos os *tweets* com informações

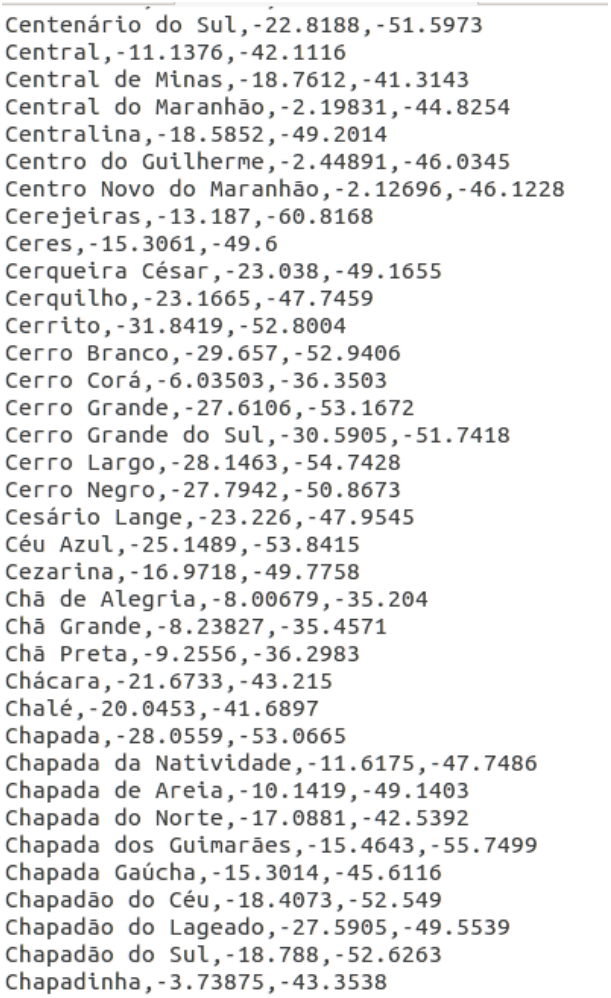

Figura 3.5: Modelo do Banco de Dados das cidades e suas latitudes e longitudes.

sísmicas, pois como dito anteriormente, dificilmente o usuário enviará uma postagem que não contenha o local do evento ocorrido. Esse método, entretanto, é oneroso para o desempenho do sistema, pois será necessária uma varredura no banco de dados em busca de informações sobre a cidade de origem do *tweet*.

A Figura 3.6 resume a inter-relação dos métodos empregados, a precisão e a quantidade de *tweets* coletados em cada um dos métodos em relação aos outros. A mais alta precisão se caracteriza por obter a latitude e longitude exata de onde o *tweet* foi postado. Já a menor precisão registrada é observada por eventos registrados à no máximo 20 quilômetros de distância de onde o *tweet* foi realmente postado. Este desvio acontece devido a localização do *tweet* ser aproximada no centro da cidade.

Assim as técnicas acima foram empregadas seguindo a ordem descrita, pois assim o sistema não fica ocupado desnecessariamente. Por exemplo, se o *tweet* possui as suas localizações geográficas de forma explícita, não é necessário minerar o texto em busca da cidade.

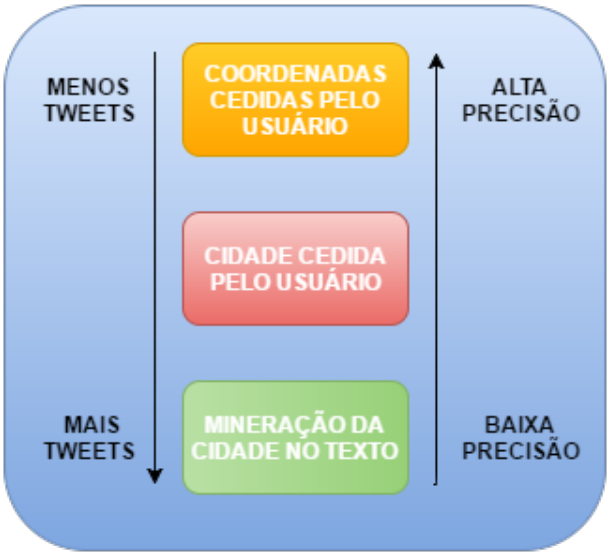

Figura 3.6: Inter-relação entre os métodos de descoberta do local da postagem.

3.2.4 Interface

A Interface do TwitSisbra foi construida utilizando bibliotecas de mapas do Google, a linguagem de marcação HTML, juntamente com CSS, e a linguagem de programação JavaScript. Para haver uma integração do TwitSisbra com o WebSisbra é necessário uma adaptação pois este foi construido usando a linguagem de programação *PHP* e o *framework Codeigniter*. A interface consiste em mostrar os *tweets* sobre eventos sísmicos no território brasileiro, que contenham algum tipo de identificação do local onde ocorrem. O método empregado de exibição foi o de mapa de calor. Um mapa de calor é uma representação bidimensional de dados, na qual os valores são representados por cores. Um mapa de calor simples fornece um resumo visual imediato da informação. Mapas de calor mais elaborados permitem que o usuário compreenda conjuntos de dados complexos. Pode haver muitas maneiras de exibir mapas de calor, mas todos eles compartilham algo em comum - eles usam cores para comunicar relações entre valores de dados que seriam muito mais difíceis de entender se apresentados numericamente em uma planilha ou de outra forma.

A interface do TwitSisbra foi projetada para permitir uma análise dos dados coletados ao invés de apenas armazená-los num banco de dados. A análise é feita a partir dos especialistas na área, do Observatório Sismológico de Brasília, que poderão receber os dados através da Rede Social, analisar se os casos apresentam um risco real à população de algum modo e avisar as autoridades responsáveis sobre um alerta. Além disso, essa interface poderá ser utilizada por usuários como agentes da comunicação, para que a informação possa realmente ser confirmada ou não. A interface é simples, mas provê recursos para a análise de eventos sísmicos. Primeiramente na Figura 3.7 pode-se ver o mapa vazio antes de iniciar as conexões com *Twiter*. Em seguida nas Figuras 3.8 e 3.9 é demonstrado como é o funcionamento do TwitSisbra.

Em seguida nas Figuras 3.8 e 3.9 é demonstrado como é o funcionamento do TwitSisbra com algumas indicações de *tweets* espalhados pelo Brasil.

Figura 3.7: Mapa onde serão exibidos os *tweets*.

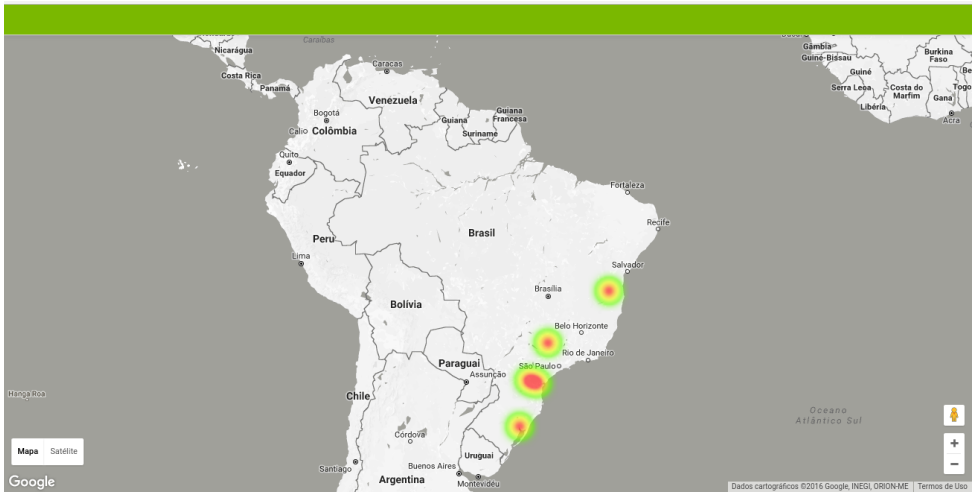

Figura 3.8: Exemplo do funcionamento do Mapa de calor do TwitSisbra 1.

3.2.5 Emissão de Alerta de Desastre

Além da análise visual realizada pelo setor especializado na área de sismologia, outra estimativa feita de forma automática é a de emitir um alerta caso haja uma certa quantidade de *tweets* num certo raio. Essa medida visa facilitar o trabalho dos analistas do SIS, com a finalidade de não ser necessária a visualização constante da tela de interface do TwitSisbra, ou a busca constante no banco de dados de *tweets* armazenados. Além disso essa medida serve para filtrar possíveis enganos na percepção de eventos sísmicos. Para estimar o raio no qual será feito as estimativas, obtemos a distância entre dois extremos da maior cidade do Brasil, São Paulo, e observamos que essa distância é de aproximadamente 40 quilômetros. Assim com um raio de 20 quilômetros é possível abranger informações sobre qualquer cidade do Brasil. A quantidade de *tweets* dentro deste raio foi estimada em 5 *tweets*, sabendo que este valor pode ser alterado de acordo as necessidades do SIS. Essa quantidade foi estimada tendo em vista que o *Twitter* não é a rede social mais utilizada no Brasil, e um valor abaixo desse poderia produzir a geração de alertas inverídicos.

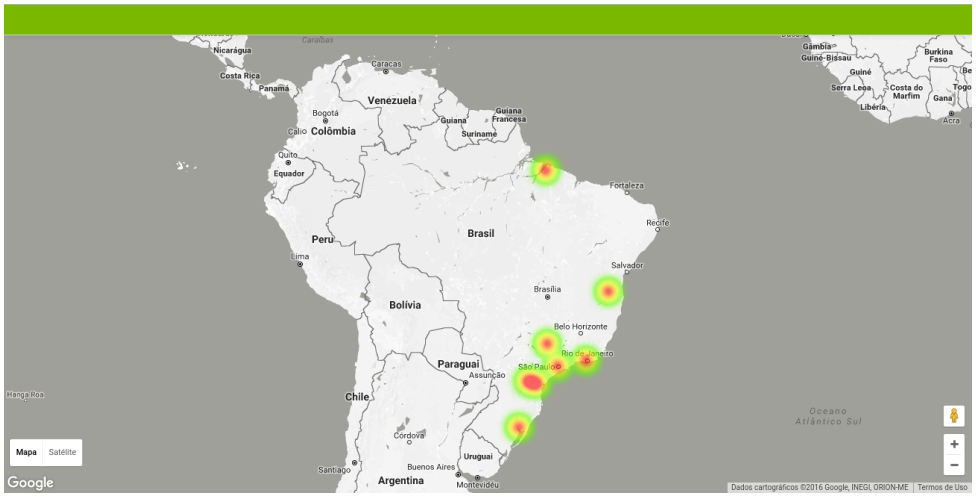

Figura 3.9: Exemplo do funcionamento do Mapa de calor do TwitSisbra 2.

Dessa forma, sempre que um *tweet* é recebido, ele é armazenado no banco de dados e se houverem 5 ou mais *tweets* referentes às informações sismológicas, armazenados no banco de dados, num raio de 20 quilômetros é emitido um alerta aos responsáveis no SIS. Este alerta será enviado através de e-mail aos analistas do SIS com as informações da região onde o evento ocorreu e quantos *tweets* foram recebido a partir daquele local.

Capítulo 4

Conclusão

A diferença entre uma situação de emergência e um desastre catastrófico é a comunicação. Obter os transmissores corretos neste contexto e mobilizar esforços de ajuda exigem uma comunicação imediata e simultânea para atingir um grande alcance da população. Agora, existe um canal de comunicação que combina os elementos mais eficazes de transmissão de emergência, transmissão de TV, rádio amador, entre outros.

O *Twitter* pode enviar mensagens curtas para muitos receptores específicos e diferentes ao mesmo tempo. Como a transmissão de TV, ele pode enviar alertas de desastres para o público em geral com imagens, sons e vídeo em tempo real. Como o rádio amador, o alcance da mensagem é global e pode ser operado por qualquer pessoa, mesmo sem equipamentos de transmissão caros ou habilidades especiais.

Quase em qualquer lugar que ocorre uma catástrofe, natural ou provocada pelo homem, há usuários do *Twitter* com dispositivos móveis nas proximidades que podem tirar fotos, gravar vídeos, e informar sobre eventos que vão acontecendo, em pequenas quantidades de informações.

Na atualidade, o Observatório Sismológico de Brasília compreende informações detectadas e coletadas através dos sismógrafos espalhados pelo Brasil. Essas informações são armazenadas e gerenciadas com o auxílio do sistema WebSisbra. Entretanto, não existe a possibilidade de analisar informações geradas a partir de redes sociais, que são amplamente utilizadas pelos usuários na atualidade.

Dessa forma, foi proposta a criação de um sistema que armazenasse informações sobre eventos sísmicos, divulgados pelos usuários da rede social *Twitter*. Esse sistema tem como objetivo criar uma interface interativa de acontecimento das postagens com os eventos citados em tempo real, no local onde eles ocorrerem. Foram empregadas as técnicas de Descoberta de conhecimento utilizando da Mineração de dados e Mineração de textos. Com isso, foi proposta a técnica de mapa de calor e os diversos conceitos de Sistemas de informação geográfica. Além disso, o sistema visa alertar os responsáveis do SIS, da ocorrência de várias postagens numa mesma região, a fim de evitar tragédias ainda piores.

Como base deste trabalho foram consultados diversos outros projetos na área de Mineração de Dados e Sistemas de Informação Geográfica. Estes trabalhos serviram como estudo para o início deste projeto.

O WebSisbra[53], foi um dos projetos que serviram como base deste trabalho. Este sistema reúne as informações coletadas pelas estações sismográficas espalhadas pelo Brasil em uma interface dinâmica e atraente.

Assim, com a definição dos requisitos do sistema, deu-se início o processo de implementação do sistema TwitSisbra, com o objetivo de coletar, armazenar e divulgar informações sobre eventos sismológicos detectados através da rede social *Twitter*.

O sistema é composto por diversas ferramentas e tecnologias como: JavaScript, Node.js, *Twitter Streaming API*, Express.js. Todas as tecnologias utilizadas neste sistema são de caráter Código Aberto(*Open Source*).

Este trabalho teve como objetivo, promover as seguintes contribuições:

- Promover a coleta e o armazenamento de dados importantes sobre eventos sísmicos detectados na rede social *Twitter*;
- Permitir a visualização de dados sismológicos num mapa tanto para analistas da área, quanto para leigos que acessarem a página do TwitSisbra;
- Emitir alerta quando ocorrerem muitos *tweets* sobre catástrofes numa determinada região aos responsáveis do SIS, e também a orgãos que lidam com desastres, como o Centro Nacional de Gerenciamento de Riscos e Desastres (Cenad).

Todo o projeto com suas instruções de uso podem ser encontrados em `https://github.com/mandrade006/twitsisbra`.

No processo de continuação deste trabalho, é possível seguir nos seguintes passos:

- Aperfeiçoar a interface do sistema permitindo ainda mais interação com o usuário. Por exemplo: ao clicar num ponto, mostrar os *tweets* com seus textos sobre aquele local;
- Promover uma Mineração de textos mais efetiva, com o propósito de identificar com ainda mais precisão o local do acontecimento do evento sismológico.
- Garantir, através de técnicas de Processamento da linguagem natural, que as informações postadas e coletadas realmente se tratam de eventos sísmicos.
- Realizar a integração do TwitSisbra com o WebSisbra, com a finalidade de agrupar os dados de origem da rede social e das estações sismológicas.

Referências

[1] Arquitetura sig. http://www.dpi.inpe.br/gilberto/livro/introd/cap3-arquitetura.pdf. Acessado em: 15/07/15. vi, 10, 11, 12

[2] Camadas de um sig. http://www.ctmgeo.com.br/softwares-servicos.php?id=3. Acessado em: 07/07/2015. vi, 7

[3] Express - node.js framework. http://expressjs.com/. Acessado em: 24/07/2016. 31

[4] Express overview - tutorial's point. http://www.tutorialspoint.com/nodejs/nodejs_express_framework.htm. Acessado em: 24/07/2016. 31

[5] Introduction to javascript. https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Introduction. Acessado em: 24/07/2016. 30

[6] Mineração na web. http://teste.tinegociosse.com.br/2014/04/15/1074/. Acessado em: 15/07/2015. vi, 25

[7] O que é o node.js. http://imasters.com.br/artigo/22016/javascript/o-que-exatamente-e-o-nodejs. Acessado em: 24/07/2016. 31

[8] Obsis - unb. http://www.obsis.unb.br/obsis/index.php?lang=pt-br. Acessado em: 11/08/2016. 27

[9] Real-time com socket.io no node.js. https://udgwebdev.com/real-time-com-socket-io-no-nodejs. Acessado em: 24/07/2016. 32

[10] Representação matricial de um sig. http://gisedu.colostate.edu/WebContent/nr505/2013_Projects/Team03/GISConcepts.html. Acessado em: 07/07/15. vi, 9

[11] Representação vetorial de um sig. http://andersonmedeiros.com/conceitos-dados-geograficos/. Acessado em: 07/07/15. vi, 8

[12] Sig - sistema de informação geográfica. http://www.topografia-etc.pt/. Acessado em: 07/07/2015. vi, 5

[13] Socket.io. http://socket.io/. Acessado em: 24/07/2016. 31

[14] Twitter access tokens. https://dev.twitter.com/oauth/overview/application-owner-access-tokens. Acessado em: 24/07/2016. 32

[15] Twitter and privacy: Nearly one-in-five tweets divulge user location through geotagging or metadata. https://pressroom.usc.edu/twitter-and-privacy-nearly-one-in-five-tweets-divulge-user-location-through-geota Acessado em: 11/08/2016. 37

[16] Twitter apis. https://dev.twitter.com/streaming/overview. Acessado em: 15/07/2015. vi, 16, 17

[17] Twitter streaming api. https://dev.twitter.com/streaming/overview. Acessado em: 24/07/2016. 32

[18] Wikipedia - a enciclopedia livre. https://pt.wikipedia.org. Acessado em: 12/08/2016. 14

[19] O poder do data mining. http://http://www.clientesa.com.br/artigos/37322/o-poder-do-data-mining/ler.aspx, Setembro 2009. 21

[20] E. C. N. Barion and D. Lago. Mineração de textos. *Revista de Ciências Exatas e Tecnologia*, 3(3):123–140, 2015. 20, 21, 22

[21] G. M. Bastos. Algumas aplicações práticas da tecnologia data mining. *Sebrae/RJ*, 2001. 21

[22] G. E. A. P. A. Batista. Pré-processamento de dados em aprendizado de máquina supervisionado. *Instituto de Ciências Matemáticas e de Computação, ICMC. São Carlos, SP*, 2003. 20, 21

[23] M. D. Beppler and A. M. d. R. Fernandes. Aplicação de text mining para a extração de conhecimento jurisprudencial. *Anais SULCOMP*, 1, 2012. 22

[24] H. Bergius. *CoffeeScript on Node.js*. O'Reilly Media, Incorporated, 2013. 31

[25] R. J. Brachman and T. Anand. The process of knowledge discovery in databases. In *Advances in knowledge discovery and data mining*, pages 37–57. American Association for Artificial Intelligence, 1996. 19

[26] F. Ceci. Um modelo semi-automático para a construção e manutenção de ontologias a partir de bases de documentos não estruturados. 2010. 23, 24

[27] G. Câmara. *Representação computacional de dados geográficos*. INPE, 2005. 4, 5, 6, 13

[28] G. Câmara, M. A. Casanova, A. S. Hemerly, G. C. Magalhães, and C. M. B. Medeiros. Anatomia de sistemas de informação geográfica. 1996. 6, 10, 11, 12

[29] E. de uso da Internet no mundo. MS Windows NT kernel description. http://comunicadores.info/2010/02/25/estatisticas-do-uso-da-internet-no-mundo/, Outubro 2010. 25

[30] O. Etzioni. The world-wide web: quagmire or gold mine? *Communications of the ACM*, 39(11):65–68, 1996. 25

[31] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. From data mining to knowledge discovery in databases. *AI magazine*, 17(3):37, 1996. 19, 20, 22

[32] E. P. Fernandes and H. P. de Freitas. Sistema de informação geográfico web para a análise de fenômenos sismológicos. 2011. 2

[33] E. J. Ferreira. Monitoramento dos tremores de terra em montes claros a partir de dados provenientes da rede sismográfica provisória da cidade. *Forum FEPEG*, 2014. 1

[34] D. Flanagan. *JavaScript: The definitive guide: Activate your web pages*. "O'Reilly Media, Inc.", 2011. 30

[35] A. L. Gonçalves. Um modelo de descoberta de conhecimento baseado na correlação de elementos textuais e expansão vetorial aplicado à engenharia e gestão do conhecimento. 2006. vi, 20, 21, 23

[36] P. Gray and H. J. Watson. The new dss: Data warehouses, olap, mdd, and kdd. In *Proceedings of the AMCIS Conference*, 1996. 19, 21

[37] M. Holanda, H. Saatkamp, H. P. de F Filho, and G. S. L. Araújo. Websisbra: Geographic information system for seismological data analysis. In *2014 9th Iberian Conference on Information Systems and Technologies (CISTI)*, pages 1–5. IEEE, 2014. 27

[38] A. Hotho, A. Nürnberger, and G. Paaß. A brief survey of text mining. In *Ldv Forum*, volume 20, pages 19–62, 2005. 22

[39] T. Imielinski and H. Mannila. A database perspective on knowledge discovery. *Communications of the ACM*, 39(11):58–64, 1996. 19

[40] J. R. C. Junior. *Desenvolvimento de uma Metodologia para Mineração de Textos*. PhD thesis, PUC-Rio, 2007. 24

[41] R. Kosala and H. Blockeel. Web mining research: A survey. *ACM Sigkdd Explorations Newsletter*, 2(1):1–15, 2000. 25

[42] M. C. S. Lopes. *Mineração de dados textuais utilizando técnicas de clustering para o idioma português*. PhD thesis, UNIVERSIDADE FEDERAL DO RIO DE JANEIRO, 2004. 22

[43] E. V. Marcelino. Desastres naturais e geotecnologias: Conceitos básicos. *INPE*, 2008. 1

[44] C. J. Matheus, P. K. Chan, and G. Piatetsky-Shapiro. Systems for knowledge discovery in databases. *IEEE Transactions on knowledge and data engineering*, 5(6):903–913, 1993. 19

[45] S. Mohan. *Node. js Essentials*. Packt Publishing Ltd, 2014. 31

[46] E. A. M. Morais and A. P. L. Ambrósio. Ontologias: conceitos, usos, tipos, metodologias, ferramentas e linguagens. *Universidade Federal de Goiás*, 2007. 18, 22

[47] R. T. S. Project. Internet live stats. http://http://www.clientesa.com.br/artigos/37322/o-poder-do-data-mining/ler.aspx. Acessado em: 19/06/2016. 25

[48] M. Rajman and R. Besançon. Text mining: natural language techniques and text mining applications. In *Data mining and reverse engineering*, pages 50–64. Springer, 1998. 24

[49] H. d. S. C. Ramos and M. B. B. Medeiros. Aplicação da descoberta de conhecimento em textos para apoio à construção de indicadores infométricos para a área de c&t. 2009. 18

[50] M. Rodrigues. Introdução ao geoprocessamento. In *Simpósio Brasileiro de Geoprocessamento*, volume 1, pages 1–26. Sagres Editora São Paulo, 1990. 8

[51] R. Rosa. Análise espacial em geografia. 2011. 6

[52] M. A. Russel. *Mining the Social Web: Data Mining Facebook, Twitter, LinkedIn, Google+, GitHub, and More*. "O'Reilly Media, Inc.", 2013. 15

[53] H. D. Saatkamp. Websisbra: Sistema nacional de registros sísmicos. *Monografia - UnB*, 2013. 1, 7, 14, 15, 42

[54] S. Salustiano. Monitoramento de redes sociais: Muito mais que uma análise de sentimentos, 2011. 15

[55] L. M. Santos. Protótipo para mineração de opinião em redes sociais: estudo de casos selecionados usando o twitter. 2010. 24

[56] P. Sharma, D. Tyagi, and P. Bhadana. Weighted page content rank for ordering web search result. *International Journal of Engineering Science and Technology*, 1(2):7301–7310, 2010. 25

[57] S. B. Shum. Graphical argumentation and design cognition. 1997. 4

[58] A. Silberschatz, H. F. Korth, S. Sudarshan, and D. Vieira. *Sistema de banco de dados*. Elsevier, 2006. 22

[59] M. P. d. S. Silva. Mineraçao de dados: Conceitos, aplicaçoes e experimentos com weka. *Livro da Escola Regional de Informática Rio de Janeiro-Espírito Santo. Porto Alegre: Sociedade Brasileira de Computação*, 1:1–20, 2004. 19, 20

[60] A. K. Singh and P. R. Kumar. A comparative study of page ranking algorithms for information retrieval. *International journal of electrical and computer engineering*, 4(7):469–480, 2009. 25

[61] J. Srivastava, P. Desikan, and V. KUmar. Web mining: Accomplishments and future directions. In *National Science Foundation Workshop on Next Generation Data Mining (NGDM'02)*, pages 51–69, 2002. 25

[62] K. L. von Bertalanffy. *Teoria Geral dos Sistemas*. Vozes, 1975. 4

[63] I. H. Witten and E. Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2005. 23

[64] L. K. Wives. Tecnologias de descoberta de conhecimento em textos aplicadas à inteligência competitiva. *Exame de Qualificação EQ-069, PPGC-UFRGS*, 2002. 18

[65] S. Yi-Xing, W. Wei, and W. Zhen-Hua. Crm measurement indicator system of logistics enterprises for data-mining. In *Logistics Systems and Intelligent Management, 2010 International Conference on*, volume 2, pages 1072–1075. IEEE, 2010. 26