cap:1

Capítulo 1

Big Social Data: Princípios sobre Coleta, Tratamento e Análise de Dados Sociais

Tiago Cruz França, Fabrício Firmino de Faria, Fabio Medeiros Rangel, Claudio Miceli de Farias e Jonice Oliveira

Abstract

Online social networks have become a popular mean of sharing and disseminating data. From these data one can extract information about patterns of interpersonal interactions and opinions, aiding in the understanding of a phenomenon, an event prediction or decision making. Nowadays, the studies and techniques for social network analysis need to work with the increase of variety and volume of data besides quickly processing them. Therefore, new approaches are required to be employed in those analyzes. Data that have such characteristics (volume, variety and velocity) are called Big Data. This short course aims to present an approach for analyzing Big Data in online social networks (Social Big Data), including the collection and processing of large volumes of social data mining and analysis principles of social interactions.

Resumo

Os dados das redes sociais online podem ser usados para extrair informações sobre padrões de interações interpessoais e opiniões. Esses dados podem auxiliar no entendimento de fenômenos, na previsão de um evento ou na tomada de decisões. Com a ampla adoção dessas redes, esses dados aumentaram em volume, variedade e precisam de processamento rápido, exigindo, por esse motivo, que novas abordagens no tratamento sejam empregadas. Aos dados que possuem tais características (volume, variedade e necessidade de velocidade em seu tratamento), chamamo-los de Big Data. Este minicurso visa apresentar uma abordagem de análise de Big Data em redes sociais online (Big Social Data), incluindo a coleta e tratamento de grande volume de dados sociais, mineração e princípios de análise de interações sociais.

1.1. Introdução

A 'redescoberta' da importância da análise de redes sociais se deu pelo uso intensivo das mídias sociais. Nesta seção entenderemos os conceitos básicos de redes sociais, aplicações da análise de redes sociais e como a coleta e tratamento de dados das redes sociais *online* (ou mídias sociais) podem ser caracterizados como um problema de *Big Data.*

1.1.1 Conceitos de Redes Sociais e a Aplicabilidade da Análise de Redes Sociais

Por definição, uma rede social é um conjunto de atores que pode possuir relacionamentos uns com os outros [Wasserman 1994].

Redes Sociais são como um organismo vivo, onde cada célula é uma pessoa. Como em todo organismo, vemos toda a sorte de células: as que permanecem nele por alguns dias, outras por alguns meses e aquelas que perduram por anos. Porém, invariavelmente, essas células acabam por deixar de existir, dando lugar a células novas ou muitas vezes nem isso. Se sairmos do campo metafórico e buscarmos algo mais concreto, do nosso dia-a-dia, podemos observar o mesmo padrão: pessoas que ficam desempregadas, se aposentam, são promovidas, empresas que se fundem, falecimentos, novas amizades, namoros e muito mais.

Em nosso dia-a-dia não faltam exemplos práticos de redes sociais: nossa família, nossos amigos de faculdade, de academia, de trabalho ou até mesmo encontros casuais, imprevistos. Eles podem ser vistos e caracterizados como a criação de um relacionamento entre dois indivíduos (nós), ligando assim as redes já existentes de ambos. Tal relacionamento pode nunca mais ser nutrido ou, como em alguns casos, vir a se tornar algo mais forte do que todos os relacionamentos já existentes. A estrutura que advém dessas inúmeras relações, normalmente, se mostra complexa.

A análise das redes sociais é feita do todo para a parte; da estrutura para a relação do indivíduo; do comportamento para a atitude. Para isto é estudada a rede como um todo, usando uma visão sociocêntrica (com todos os elos contendo relações específicas em uma população definida) ou como redes pessoais, em uma visão egocêntrica (com os elos que pessoas específicas possuem, bem como suas comunidades pessoais) [Hanneman 2005].

Podemos citar alguns cenários onde é aplicada a análise de redes sociais. No caso de uma empresa, por exemplo, é importante saber como os seus funcionários se organizam, pois assim é possível evitar problemas característicos que dificultam a disseminação de conhecimento [Wasserman 1994]. No campo da Ciência, as redes sociais podem auxiliar no estudo de propagações de endemias ou mesmo de epidemias [Pastor-Satorras 2001; Mikolajczyk 2008], bem como serem utilizadas para entender [Albuquerque 2014; Stroele 2011] ou melhorar a formação de grupos [Souza et al. 2011; Monclar et al. 2012; Melo e Oliveira 2014; Zudio et al. 2014]. Como tática de propaganda e 'marketing' podemos usá-las como uma ferramenta para ajudar a propagar uma determinada marca ou conceito, podendo servir, também, para o estudo de um público-alvo relacionado, verificando em quais nós aquelas informações morrem e em quais elas seguem em frente [Kempe 2003; Santos e Oliveira 2014]. Além disso, podemos aproveitar uma das grandes questões do início do século XXI desde os eventos de 11 de setembro de 2001, os atentados, para pensar na detecção e identificação de terroristas.

Pode-se realizar uma verificação de pessoas com as quais os terroristas costumam se relacionar, traçando um padrão de redes que auxilie futuras investigações [Svenson 2006]. Outra aplicação advém do levantamento de relacionamentos entre '*Weblogs'*[1] que realizam propaganda de ideais terroristas, montando a respectiva rede social representada por eles, como visto em [Yang 2007].

A tendência de pessoas se unirem e formarem grupos é uma característica de qualquer sociedade [Castells 2000]. Esse comportamento é retratado, nos dias atuais, através do avanço das mídias sociais e comunidades *online* que evidenciam o poder de unir usuários ao redor do mundo. Conteúdos gerados por seus usuários atingiram alto grau de alcance através de seus comentários, relatos de acontecimentos quase em tempo real, experiências, opiniões, críticas e recomendações que são lidos, compartilhados e discutidos, de forma quase instantânea, em diversas plataformas disponíveis na *Web*.

1.1.2 Redes Sociais no Mundo Digital: Redes Sociais *Online*

O crescimento de usuários de telefones celulares e *tablets* com acesso à rede permite que as pessoas permaneçam conectadas ao longo do dia, aumentando a quantidade de informações disponibilizadas na Internet. De acordo com o relatório do instituto de pesquisa tecnológica Cozza *et al.* [2011], 428 milhões de dispositivos móveis, incluindo celulares, *smartphones* e *tablets* foram vendidos ao redor do mundo no primeiro semestre do ano de 2011 com 19% por cento de crescimento em relação ao ano anterior. Grande parte do acesso e disponibilização do conteúdo se deve à popularização das mídias sociais.

Mídias Sociais é um tipo de mídia *online* que permite que usuários ao redor do mundo se conectem, troquem experiências e compartilhem conteúdo de forma instantânea através da Internet. Elas são fruto do processo de socialização da informação nos últimos anos representado pela extensão do diálogo e do modo como as informações passaram a ser organizadas através da *Web*.

As mídias sociais e consequentemente o crescimento do seu uso pela população implicaram numa mudança de paradigma em relação à disseminação da informação. As grandes mídias como jornais, revistas e portais passaram a não ser os mais importantes provedores de informações para a população, ou seja, o modelo de disseminação "um para muitos" foi sendo substituído pelo modelo "muito para muitos" [Stempel 2000]. As mídias sociais deram espaço para que usuários gerassem e compartilhassem conteúdo de forma expressiva, deixando de lado o comportamento passivo de absorção da informação que possuíam, caracterizando uma forma de democratização na geração de conteúdo.

Segundo Solis (2007), o termo mídia social descreve tecnologias e práticas *online* que pessoas usam para compartilhar opiniões, experiências e perspectivas, podendo se manifestar em diferentes formatos incluindo texto, imagens, áudio e vídeo. A expansão dessas mídias tornou mais simples encontrar amigos, compartilhar ideias e opiniões e obter informações, ou seja, uma democratização do conteúdo em que todos participam na construção de uma comunidade virtual.

Existem diversos tipos de mídias sociais com os mais diferentes focos. As categorias mais conhecidas são:

[1] 'Weblogs' são páginas pessoais, ou sites sem fim lucrativos, dedicados a trazer informações sobre um determinado tema [Blood 2002].

• Colaboração: está relacionada às redes sociais colaborativas, ou seja, *sites* em que é importante a interação de diferentes usuários compartilhando informações a fim de atingir um objetivo comum. Como exemplo destacam-se a Wikipedia, Yelp e Digg.

• Comunicação: está relacionada ao fenômeno da conversação entre pessoas e o modo como essa conversa é percebida por seus participantes, que podem participar de forma direta, através da realização de comentários e produção de conteúdo, ou indireta, compartilhando e divulgando conteúdo e, consequentemente, ajudando a promover discussões. Podem-se citar, neste contexto, os *blogs* e *microblogs*, as redes sociais *online* (RSO) e os fóruns. Como exemplo, é possível mencionar WordPress[2], Twitter[3], Facebook[4] e GoogleGroups[5], respectivamente.

• Multimídia: refere-se aos componentes audiovisuais que ficam além do texto puro e simples como fotos, vídeos, *podcasts* e músicas. Alguns exemplos dessas mídias, respectivamente, são Flickr[6], YouTube[7], JustinTV[8] e Lastfm[9].

• Entretenimento: diz respeito aos conteúdos que geram um mundo virtual favorecendo o desenvolvimento da "gamificação", ou seja, ambientes focados em *games online* ou ainda atividades que podem ser transformadas em algum tipo de competição, nos quais seus usuários se juntam com o objetivo de jogarem juntos ou compartilharem informações a respeito do tema. Como exemplo podem ser citados o Second Life[10] e TvTag[11].

A popularidade dessas plataformas pode ser evidenciada através da capacidade que possuem de produzir enormes volumes de conteúdo. Conforme as estatísticas do Facebook [Statistics Facebook 2011], este possui mais de 800 milhões de usuários ativos e 50% destes usuários acessam o *site* todo dia. De acordo com o Compete[12], o Twitter possuía em Setembro de 2012 cerca de 42 milhões de usuários únicos registrados representando o vigésimo primeiro site da Internet em número de visitantes únicos e segundo Dale [2007], o YouTube concentra 20% de todo o tráfego de dados da Internet.

A monitoração destas mídias sociais tornou-se um problema de *Big Data*, onde precisamos tratar um volume grande de dados, variedade (já que tais mídias apresentam características distintas no que se relaciona à estrutura, dinâmica, uso e modelagem) e necessidade de velocidade em seu tratamento para que diferentes análises sejam viáveis.

1.2 Redes Sociais *Online*: Coleta de dados e Técnicas de Análise

Nesta seção, apresentamos os princípios da coleta dos dados e a análise das redes sociais extraídas a partir destes dados.

[2] https://wordpress.com
[3] http://www.twitter.com
[4] https://www.facebook.com
[5] https://groups.google.com
[6] https://www.flickr.com/
[7] https://www.youtube.com/
[8] http://www.justin.tv/
[9] http://www.lastfm.com.br/
[10] http://secondlife.com/
[11] http://tvtag.com/
[12] http://www.compete.com

1.2.1 A informação dos dados coletados na *Web*

A *Web* é composta por uma grande quantidade de dados que não possuem uma estrutura definida ou que não possuem uma semântica explícita. Por exemplo, uma página HTML possui uma estrutura, mas esta estrutura pouco atribui informações sobre os dados presentes na página. Sendo assim, ao encontrar uma tag "<img>" em um determinado ponto do arquivo HTML é possível afirmar que se trata de uma imagem, porém pouco se pode afirmar sobre o conteúdo desta imagem. Ainda que a tag "<img>" possua um atributo que seja um texto relativo a essa imagem, textos são dados não estruturados e é necessário aplicar técnicas em mineração de dados para extrair informações do texto.

Segundo Chen (2001), estima-se que 80% de todo conteúdo mundial *online* são textos. Considerando que dados não estruturados englobam textos, imagens, vídeos e músicas, pode-se perceber que realmente grande parte da *Web* é composta de dados não estruturados.

Devido à necessidade de aperfeiçoar os mecanismos de busca e utilizar a *Web* como plataforma de integração, por meio de serviços, há uma busca crescente em estruturar os dados. Essa estruturação, entretanto, deve ter uma flexibilidade dado a própria natureza da *Web* em que há uma vasta variedade de dados. Com base nesse problema, alguns padrões foram criados. Entre eles, os padrões XML e JSON são os mais utilizados. Entretanto, nada impede de que outros padrões sejam criados ou aplicados. Essa diversidade cria um problema no processamento dos dados, pois é necessário criar uma aplicação para cada padrão de representação. Além disso, um mesmo padrão de representação pode estruturar o mesmo conjunto de dados de diferentes formas, como ilustrado na Figura 1.1.

1.2.2 Coleta de Dados: Crawler de páginas e APIs de RSO

Atualmente, as principais redes sociais *online* (RSO) provêem interfaces ou serviços para a captura parcial ou total de seus dados. Nesta seção, comentaremos os principais desafios e recursos para se trabalhar com as principais (RSO) existentes atualmente.

1.2.2.1 Coletando Dados de Páginas Estáticas e Dinâmicas da *Web*: Ferramentas e Desafios

Considera-se dinâmica uma página cuja atualização dependa de uma aplicação *Web*, enquanto páginas estáticas costumam não ter seu conteúdo modificado. Quando se faz necessário coletar dados de páginas, sejam estáticas ou dinâmicas, é necessário entender a estrutura dos dados contidos nessa página a fim de desenvolver um *crawler* capaz de buscar e armazenar esses dados. Quando a página não possui API de consumo ou a API possui limites indesejados, é possível utilizar ferramentas para capturar as páginas e extrair os dados sem a utilização de APIs. Um exemplo dessas ferramentas é o Node.js. O Node.js é uma plataforma construída na máquina virtual Javascript do Google para a fácil construção de aplicações de rede rápidas e escaláveis [Node 2014].

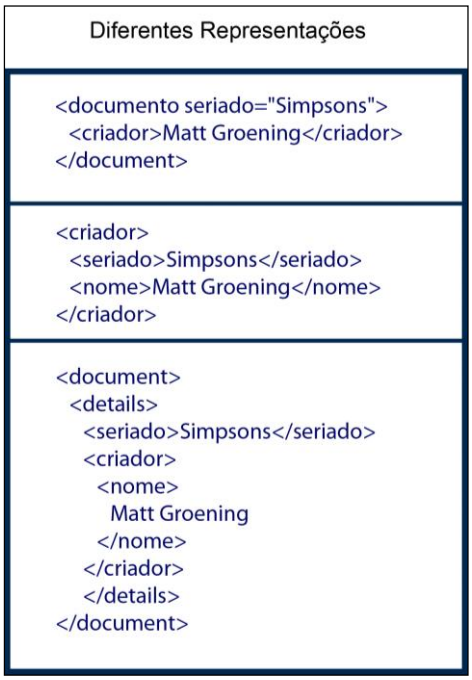

Figura 1.1. Exemplos de conjuntos de dados em diferentes representações.

1.2.2.2 Coletando Dados do Twitter, Facebook, YouTube e Foursquare

Normalmente, há duas formas diferentes de coleta de dados das redes sociais *online*. A primeira forma consiste em determinar termos e coletar por citações destes termos no passado. Desta forma, existe a possibilidade de restrições na obtenção de dados antigos, pois normalmente há um período de tempo viável para a coleta dos dados. A segunda se baseia em um conceito de *streaming*, onde a aplicação criada funciona como um "ouvinte" da rede e captura os dados à medida que estes surgem.

Twitter

O Twitter é uma rede social *online* que possui duas APIs diferentes para a captura dos seus dados: REST API e *Streaming API*. Para a utilização de ambas APIs é necessário inicialmente que o usuário tenha uma conta no Twitter. Acessando a página https://dev.twitter.com, é possível autenticar-se com a conta do Twitter e cadastrar uma aplicação. Após o cadastro da aplicação é necessário gerar o *Access Token* da mesma. Importante destacar que, tanto o *API Secret* como o *Access Token Secret* da sua aplicação não devem ser divulgados por questões de segurança. Essas chaves serão utilizadas na autenticação da sua aplicação de captura de dados.

O Twitter trabalha com o padrão de arquivo JSON. Todos os dados são recebidos nesse formato. Um exemplo da utilização do *Streaming* API pode ser visto na Figura 1.2.

Os programas usados como exemplo de *crawlers* para o *Twitter* foram codificados em Python 2.7. No primeiro exemplo (Figura 1.3), foram utilizadas as bibliotecas "Auth1Session" e "json". A Auth1Session é responsável pelo estabelecimento da conexão com o Twitter e a biblioteca JSON é responsável por transformar o texto recebido em um objeto Python cuja estrutura é no formato JSON. Assim, é possível manipular o arquivo JSON. Um exemplo utilizando a REST API do *Twitter* pode ser visto na Figura 1.3.

```
import json
from requests_oauthlib import OAuth1Session
key = "{sua API key}"
secret = "{sua API secret}"
token = "{seu Acess Token}"
token_secret = "{seu Acess Token Secret}"
requests = OAuth1Session(key, secret, token, token_secret)
r = requests.post('https://stream.twitter.com/1/statuses/filter.json',
data={'track': 'bom dia'},
stream=True)
for line in r.iter_lines():
   if line:
      print json.loads(line) # tweet retornado
```

Figura 1.2. Exemplo usando a *Streaming* API.

```
import oauth2 as oauth
import json
import time
CONSUMER_KEY = "{sua API key}"
CONSUMER_SECRET = "{sua API secret}"
ACCESS_KEY = "{seu Acess Token}"
ACCESS_SECRET = "{seu Acess Token Secret}"
consumer = oauth.Consumer(key=CONSUMER_KEY, secret=CONSUMER_SECRET)
access_token = oauth.Token(key=ACCESS_KEY, secret=ACCESS_SECRET)
client = oauth.Client(consumer, access_token)
q = 'israel' #termo a ser buscado
url = "https://api.twitter.com/1.1/search/tweets.json?q="+str(q)+"&count=100"+"&lang=pt"
response, data = client.request(URL, "GET")
tweets = json.loads(data)
for tweet in tweets['statuses']:
     print str(tweet)
```

Figura 1.3. Exemplo usando a REST API.

No código apresentado na Figura 1.3, a variável "q" representa a consulta buscada no *Twitter*. Neste caso, buscou-se pelo termo "israel". A API retorna os 100 *tweets* mais recentes que possuem esse termo, que foi passado como parâmetro na URL de requisição através do parâmetro "count" igual a 100. Ainda sobre a consulta, a API retorna *tweets* com todos os termos presentes na string "q" separados por espaços. Ou seja, se "q" possui "israel guerra", todos os *tweets* retornados possuirão as duas palavras. Exemplo: "Há guerra em Israel" seria um possível texto do *tweet* retornado.

Utilizando a REST API, existem algumas restrições impostas pelo Twitter. Ao que se sabe, o Twitter não permite que a API busque por *tweets* mais antigos do que 7

dias e ainda bloqueia a aplicação caso ultrapasse o número de requisições permitidas, sendo necessário um intervalo de 15 minutos para que a aplicação seja desbloqueada.

Facebook

Uma das APIs de consumo disponibilizadas pelo Facebook se chama Graph API. Existe outra API chamada Public Feed API, porém esta possui acesso restrito a um conjunto de editores de mídia e seu uso requer aprovação prévia do Facebook [Facebook 2014].

```
import httplib
app_id = "{sua app id}"
app_secret = "{seu app secret}"
client_credentials = "{label para de identificação do cliente}"
url = "/oauth/access_token?client_id="+app_id+"&client_secret="+app_secret+"&grant_type=client_credentials"
conn = httplib.HTTPSConnection("graph.facebook.com")
conn.request("GET", str(url))
response = conn.getresponse()
print response.read() #access token
```

Figura 1.4. Exemplo usando a Graph API.

Primeiramente, para utilizar a Graph API é necessário criar uma aplicação. Essa aplicação pode ser criada no site https://developers.facebook.com/ e será vinculada a uma conta no Facebook. A segunda etapa consiste em gerar um *access token* para aquela sessão. Um exemplo, utlizando Python 2.7, de como adquirir o *access token* pode ser visto na Figura 1.4. Neste exemplo, utilizou-se a biblioteca *httplib* para executar a requisição.

É importante destacar que algumas requisições necessitam de um *app access token* e outros necessitam do *user access token*. Este último deve ser criado no próprio site do Facebook através do endereço https://developers.facebook.com/tools /accesstoken/.

Com o Graph API é possível buscar no Facebook por certos objetos que possuam um determinado termo. Esses objetos podem ser usuário, páginas, eventos, grupos e lugares. Porém, uma limitação da API é que não é possível procurar por *posts* públicos onde um determinado termo aparece. Entretanto, a busca por páginas e lugares requer um *app access token*, enquanto buscas pelos outros objetos utilizam o *user access token*.

A Figura 1.5 mostra a utilização do Graph API na busca por usuários com o termo “Fabio”, codificado utilizando Python 2.7.

As requisições ao Facebook do tipo *search* retornam objetos no formato JSON.

```
import httplib
user_access_token = "{seu user access token}"
q = "Fabio"
url_consulta = "/search?q="+str(q)+"&type=user&access_token="+str(user_access_token)
conn = httplib.HTTPSConnection("graph.facebook.com")
conn.request("GET", str(url_consulta))
response = conn.getresponse()
print response.read()
```

Figura 1.5. Segundo exemplo usando a Graph API.

Youtube

A API do Youtube atual é a versão 3.0. Nesta versão, é possível buscar por informações de vídeos e canais com base nos seus *ids*. Entretanto, não é possível buscar por usuários que comentaram em um determinado vídeo ou mesmo por comentários de um vídeo. Isso pode ser uma grande limitação quando se pensa em análises utilizando o Youtube. Mesmo assim, dado um *id* de um determinado vídeo e requisições feitas em determinados intervalos de tempo, é possível conhecer a progressão de visualizações de um vídeo e acompanhar sua divulgação em outra rede social, correlacionando esses dados e estudando a interatividade entre as redes.

Para utilizar a API do youtube, é necessário a criação de uma aplicação na Google, no site https://console.developers.google.com/. Após a criação, é necessário a ativação da API do Youtube v3.0. Para a consulta de dados de vídeos ou vídeos de um determinado canal, utiliza-se o protocolo HTTPS. Uma *uri* exemplo para uma consulta é: https://www.googleapis.com/youtube/v3/videos?part=statistics&id=ZKugnwXU5_s&key={Key da sua aplicação}.

Nesse caso, serão retornadas as estatísticas do vídeo cujo id foi passado como parâmetro, no formato JSON. Mais informações e exemplos de requisições estão disponíveis em https://developers.google.com/youtube/v3/docs/.

Foursquare

O Foursquare pode ser definido como uma rede geossocial, onde seus usuários podem indicar onde se encontram ou procurar por outros usuários que estejam próximos geograficamente. A API de consumo do Foursquare permite buscas por uma determinada latitude e longitude a fim de retornar locais de interesse público daquela região. No retorno das requisições é possível verificar a quantidade de check-in desses locais, além da quantidade de pessoas que recentemente efetuaram um check-in naquele local. Além desta busca, a API permite outra busca por *tips* (dicas), que tem como retorno dicas dos usuários sobre locais em uma determinada região. A API pode ser acessada no *site* https://developer.foursquare.com e o usuário precisa criar uma aplicação no *site* para começar a utilizar. Um exemplo de requisição da API do Foursquare, codificada em Python 2.7, pode ser visto na Figura 1.6.

Dentre os parâmetros da requisição HTTP, "v" é um apenas um controle de versão, onde o desenvolvedor da aplicação pode informar ao foursquare se está utilizando uma versão anterior da API. No caso, este parâmetro recebe uma data no formato YYYYMMDD.

```
import urllib2


client_id = "{seu client id}"
client_secret = "{seu client secret}"
par_v = "20141107"
location = "40.7,-74"
query = "sushi"
response = urllib2.urlopen("https://api.foursquare.com/v2/tips/search"+
"?client_id="+client_id+
"&client_secret="+client_secret+
"&v="+par_v+
"&ll="+location+
"&query="+query)
html = response.read()
print str(html)
```

Figura 1.6. Exemplo usando a API do Foursquare.

1.2.3 Princípios de Análises de Redes Sociais

Após a leitura, desambiguação[13] dos nós e montagem do sociograma (grafo de relacionamentos), chegou a hora de analisarmos a rede social.

Em um sociograma, podemos representar os relacionamentos como arestas, as quais podem ser direcionadas (João envia um e-mail para Maria) ou não (João e Maria participaram de uma reunião). Nesta seção, apresentamos apenas métricas de grafos não direcionados. Vale lembrar que para todas as métricas apresentadas neste capítulo existem suas respectivas versões para grafos direcionados. Maiores detalhes sobre tais métricas podem ser encontradas em [Newman 2010]. Casos e exemplos de uso podem ser obtidos em [Easley e Kleinberg 2010].

1.2.3.1 Principais Métricas de Análise de Redes Sociais

A seguir, resumidamente foram descritas algumas métricas divididas naquelas que são referentes aos nós de modo individual e aquelas que medem o comportamento da rede como um todo.

[13] Quando estamos lidando com mídias sociais, normalmente temos diferentes usuários associados à uma única entidade. Por exemplo, imagine o João da Silva (entidade) que possui várias contas. No Twitter pode ser reconhecido como JS_RJ. No Facebook ele possui duas contas, uma de caráter profissional (Prof. João da Silva) e outra pessoal (João Carioca). O processo de desambiguação dos nós significa identificar as diferentes contas de uma mesma entidade e associá-las. Neste exemplo, identificaríamos que “JS_RJ”, “Prof. João da Silva” e “João Carioca” são diferentes contas de uma mesma pessoa.

Métricas Individuais

A importância de um vértice pode ser identificada através do cálculo de centralidade. Uma destas centralidades é o *grau do vértice* (também conhecida como grau de centralidade ou ainda centralidade local por Wasserman e Faust (1994)), ou seja, o número de arestas conectadas diretamente a ele. A *centralidade global* de um vértice, também conhecida como *closeness* ou grau de proximidade é a soma do menor caminho entre um vértice e os demais vértices da rede. Um vértice que tenha a menor soma das menores distâncias está mais perto dos demais. Ou seja, quanto maior a centralidade global, maior será a distância de um membro para com os demais. Isto significa que o trajeto de um dado, informação ou conhecimento para chegar a um destes membros isolados é maior, e consequentemente, pode demorar mais, como também podem chegar deturpados e com ruído.

Outra medida de centralidade muito utilizada é o *betweenness* ou grau de intermediação, que está relacionado ao número de caminhos mínimos aos quais um vértice pertence. O grau de intermediação revela o quanto um vértice está no caminho entre os outros vértices numa rede. Quanto maior for o valor deste grau, significa que este vértice é uma "passagem obrigatória", muitas vezes conectando diferentes grupos.

Seja $\mathbf{x_i}$ a quantidade de todos os caminhos mínimos entre **s** e **t** (caso exista mais de um) cujo vértice **i** está incluído, então, sendo $\mathbf{n_{st}^i}$ um destes caminhos mínimos entre os vértices **s** e **t**, $\mathbf{n_{st}^i = 1}$, se o vértice **i** pertence a este caminho mínimo e $\mathbf{n_{st}^i = 0}$ caso contrário [5], como está representado na equação (3.2):

$$x_i = \sum_{st} n_{st}^i$$

Seja $\mathbf{g_{st}}$ o número total de caminhos mínimos entre **s** e **t**, então o grau de intermediação do vértice **i** será $\mathbf{X_i}$ calculado através da equação (3.3):

$$X_i = \frac{x_i}{g_{st}}$$

Outra métrica, conhecida como *coeficiente de agrupamento*, pode informar para cada vértice e para a rede como um todo, como uma rede se apresenta em termos de grupos. Desta análise vem a definição de coeficiente de agrupamento: a probabilidade de que dois vizinhos de um vértice serão vizinhos entre si. Para calcular o coeficiente de agrupamento $\mathbf{C_i}$, são contados todos os pares de vértice que são vizinhos de **i** e sejam conectados entre si, então, divide-se este valor pelo número total de vizinhos de **i**, ou $\mathbf{k_i}$, que é o grau de **i**:

$$C_i = \frac{NúmeroDeParesDeVizinhosDeiQueSãoConectados}{NúmerodeParesdeVizinhoDei}$$

Outras centralidades igualmente importantes são o PageRank e o cálculo de Hubs, que podem ser encontrados em [Newman 2010].

Métricas da Rede

Watts e Strogatz (1998) propuseram o cálculo do coeficiente de agrupamento médio, $\mathbf{C_m}$, para a rede através da média do coeficiente de agrupamento local para cada vértice. Ou seja, C_m é o somatório dos coeficientes de agrupamento de cada vértice do grafo, normalizado pelo número total de vértices.

$$C_m = 1/n \sum_{1}^{n} C_i$$

Outra medida é o diâmetro da rede, que é a distância máxima entre dois vértices, é o maior caminho mínimo (geodésico) entre dois vértices da rede, simbolizando o nível de ligação entre os vértices da rede. Newman (2010) define a densidade de uma rede utilizando o grau médio da rede em questão. Primeiramente calculando o valor de **M,** o número máximo possível de arestas numa rede de **n** vértices.

$$M = \binom{n}{2} = \frac{1}{2}\mathrm{n}(n-1)$$

Sendo **G** o grau médio já calculado para esta rede com **m** arestas, a densidade ρ é obtida fazendo-se:

$$\rho = \frac{m}{\binom{n}{2}} = \frac{2\mathrm{m}}{n(n-1)} = \frac{G}{(n-1)}$$

Se a densidade de uma rede varia no intervalo [0,1], quanto mais próximo de zero, menos conectada é a rede. O contrário é válido, quanto mais próximo de um, a rede é mais densa. A vantagem do uso desta medida está na simplicidade de seu cálculo, no entanto, para redes com extenso número de nós, torna-se custoso realizar tal cálculo.

1.2.3.2 Principais Ferramentas

Existem diversas ferramentas (muitas delas gratuitas!) que automatizam a análise, possuindo as principais métricas implementadas e ferramentas de visualização e extração de relatórios. Várias delas lêem e exportam para diferentes formatos. Em [Huisman e Van Duijn 2005] vocês têm uma análise comparativa destas principais ferramentas. Os autores mantêm o site http://www.gmw.rug.nl/~huisman/sna/software.html atualizado com novos programas destinados à análise de redes sociais.

Neste curso a ferramenta Gephi[14] será utilizada. Essa ferramenta se apresentou como uma boa opção quando se analisa redes não muito grandes.

1.2.4 Princípios de Mineração de Informação

Mineração de dados é o processo de explorar dados à procura de padrões consistentes. Na análise de redes sociais, esses padrões descrevem como os indivíduos interagem ou as características (regras) que dão origem às redes sociais. Identificar fatores e as tendências-chave dos dados que a rede produz também são aplicações possíveis de mineração de dados em redes sociais.

Grafos representam estruturas de dados genéricas que descrevem componentes e suas interações, sendo assim são adotados para representar as redes sociais. Como consequência, os métodos de mineração de dados apresentados a seguir terão como foco a mineração em grafos. Segundo a taxonomia apresentada por Getoor e Diehl [Getoor e Diehl 2005], a mineração para grafos pode ser dividia em três grandes grupos: mineração orientados a objetos; mineração orientada a links; e mineração orientada a grafos. Os grupos e os métodos de cada grupo são: (i) Tarefas relacionadas a objetos; (ii) Tarefas

[14] https://gephi.github.io/

relacionadas a ligações e (iii) Tarefas relacionadas a grafos. Por sua vez Tarefas relacionadas a objetos se dividem em: (i) *Ranking* baseado em ligação de objetos (RBLO); (ii) Classificação baseada em ligação de objeto (CBLO); (iii) Agrupamento de objetos e (iv) Identificação de objetos. As tarefas relacionadas a ligações resumem-se à predição de ligações. Por fim, as tarefas relacionadas a grafos se dividem em: (i) Descoberta de subgrafos e (ii) Classificação de grafos. Nas próximas subseções será feito o detalhamento de cada método. Essa taxonomia de tarefas de mineração em grafos foi retirada de [Getoor e Diehl 2005].

1.2.4.1 *Ranking* baseado em *link* de objetos

Uma das tarefas mais comuns na mineração de dados em grafos, o *ranking* baseado em *link* de objetos explora a estrutura dos *links* de um grafo para ordenar e priorizar o conjunto de vértices. Dentre os métodos para RBLO, os algoritmos HITS [Kleinberg 1999] e PageRank [Ranking e Order 1998] são os mais conhecidos.

O PageRank funciona contando o número de ligações entre os vértices de um grafo orientado para estimar a importância desse vértice. Ligações de vértices importantes (onde a importância é definida pelo valor de PageRank desse vértice) para um vértice, fazem com que esse último melhore seu *ranking*. As ligações que um vértice faz com outros, ponderado pela importância desse vértice, faz com que o primeiro diminua sua importância. O balanço entre quais vértices apontam e são apontados determinam o grau de importância de um nó.

O algoritmo HITS é um processo mais complexo se comparado com o PageRank, modelando o grafo com dois tipos de vértices: *hubs* e entidades. *Hubs* são vértices que ligam muitas entidades e por consequência, entidades são ligadas por *hubs*. Cada vértice do grafo nesse método recebe um grau de *hub* e de entidade. Esses valores são calculados por um processo iterativo que atualiza os valores de hub e entidade de cada vértice do grafo baseado na ligação entre os vértices. O algoritmo de HITS tem relação com o algoritmo de PageRank com dois *loops* separados: um para os *hubs* e outro para as entidades, correspondendo a um grafo bi-partido.

1.2.4.2 Classificação baseada em ligação de objetos

A classificação baseada em ligação de objetos (CBLO) tem como função rotular um conjunto de vértices baseado nas suas características, diferindo dessa forma das técnicas de classificação tradicionais de mineração de dados por trabalhar com estruturas não homogêneas. Apesar da sua importância, é uma área de mineração não consolidada, apresentando como desafio o desenvolvimento de algoritmos para classificação coletiva que explorem as correlações de objetos associados [Getoor e Diehl 2005].

Dentre as propostas da CBLO, destacam-se os trabalhos de Lafferty *et al* (2001), que é uma extensão do modelos de máxima entropia em casos restritos onde os grafos são cadeias de dados. Taskar et al. (2002) estenderam o modelo Lafferty et al (2001) para o caso em que os dados formam um grafo arbitrário. Lu e Getoor (2003) estenderam um classificador simples, introduzindo novas características que medem a distribuição de classes de rótulos em uma cadeia de Markov.

1.2.4.3 Agrupamento de objetos

O objetivo do Agrupamento de Objetos é o agrupamento de vértices de um grafo por meio de características comuns. Várias técnicas foram apresentadas em várias comunidades para essa finalidade. Entretanto, o desenvolvimento de métodos escaláveis adequados para exploração de grafos complexos em tempo hábil ainda é um desafio.

Para grafos com arestas e vértices de um único tipo e sem atributos, pode-se utilizar técnicas de agrupamento aglomerativo ou divisivos. A tarefa de agrupamento envolve o particionamento de redes sociais em conjuntos de indivíduos que possuam um conjunto similar de *links* entre si. Uma medida de similaridade entre o conjunto de arestas e o agrupamento aglomerativo é definida e usada para identificar as posições. Métodos de separação espectral do grafo resolvem o problema de detecção de grupo, identificando um conjunto mínimo aproximado de arestas que devem ser removidas do grafo para atingir um determinado número de grupos.

Outras abordagens para detecção de grupo fazem uso das medidas de *betweenness* dos vértices. Um exemplo é o método Girva-Newman (2002), o qual detecta uma comunidade removendo progressivamente arestas do grafo original. Esses autores consideram que se dois vértices possuem uma aresta ligando-os e se esses nós apresentam um valor de *betweenness* alto, provavelmente a aresta que os liga é uma ponte. Se a ponte for removida, os agrupamentos tornam-se visíveis.

1.2.4.4 Identificação de Entidades

Denominada também de resolução de objetos, a identificação de entidades tem como objetivo determinar quais dados que fazem referência a entidades do mundo real. Tradicionalmente, a resolução de entidades é vista como um problema de semelhança entre atributos de objetos. Recentemente, houve significante interesse em usar *links* para aperfeiçoar a resolução de entidades com o uso de ligações entre vértices [Getoor e Diehl 2005].

A ideia central é considerar, em adição aos atributos dos vértices de um grafo que representa uma rede social, os atributos dos outros vértices que estão ligados com ele. Essas ligações podem ser, por exemplo, coautorias em uma publicação científica, onde os atributos dos indivíduos (nome, área de pesquisa) seriam utilizados em conjunto com os atributos dos coautores que trabalham com ele [Alonso et al. 2013].

1.2.4.5 Predição de Ligações

A predição de *links* trata do problema de predizer a existência de ligações entre dois vértices baseado em seus atributos e nos vértices existentes. Exemplos incluem predizer ligações entre atores de uma rede social, como amizade, participação dos atores em eventos, o envio de email entre atores, chamadas de telefone, etc. Na maior parte dos casos, são observados alguns *links* para tentar predizer os *links* não observados ou se existe algum aspecto temporal.

O problema pode ser visto como uma simples classificação binária: dado dois vértices **v1** e **v2**, preveja quando um vértice entre **v1** e **v2** será 1 ou 0. Uma abordagem é fazer a predição ser inteiramente baseada em propriedades estruturais da rede. Liben-Nowell e Kleinberg (2007) apresentaram um *survey* sobre predição de *links* baseado em diferentes medidas de proximidade. Outra abordagem fazendo uso de informações dos

atributos é a de Popescul *et al* (2003), que propõe um modelo de regressão logística estrutural que faz uso das relações para predizer a existência de novas ligações. O'Madadhain *et al* (2005) propõe a construção de um modelo local de probabilidades condicionais, baseado nos atributos e na estrutura. A respeito de predição de ligações, podemos ainda referenciar o minicurso de Appel e Hruschka (2011), ministrado no próprio SBBD.

1.2.4.6 Descoberta de subgrafos

A determinação de subgrafoss frequentes é um importante instrumento para a análise de redes sociais, pois permite caracterizar e discretizar conjuntos de grafos, criar mecanismos de classificação e agrupamento, criar índices de busca, etc. Por exemplo, em um conjunto de grafos que representam uma rede colaborativa de coautoria em trabalhos científicos, os subgrafos frequentes podem identificar os grupos de pesquisas ou grupos de um mesmo laboratório.

Um grafo **g'** é definido como subgrafo de um grafo **g** se existir um padrão isomórfico entre **g'** e **g**, ou seja, o conjunto de vértices e arestas de **g'** são subconjuntos dos conjuntos de vértices e arestas de **g** respectivamente. Dentre os métodos de mineração de subgrafoss frequentes, duas abordagens são as mais empregadas: a abordagem baseada no método apriori e a abordagem baseada em padrões de crescimento. Ambas possuem em comum buscar estrutura frequentes nas bases de grafos. A frequência mínima que um subgrafo deve ocorrer na base **D** para que seja considerado frequente recebe o nome de suporte.

A busca por grafos frequentes, utilizando o método apriori, começa por subgrafos de menor "tamanho" e procede de modo *bottom-up* gerando candidatos que possuam um vértice, aresta ou caminho extra que possuam um valor de suporte maior do que um valor pré-definido (min_sup). Para determinar se um grafo de tamanho **k+1** é frequente, é necessário checar todos os subgrafos correspondentes de tamanho **k** para obter o limite superior de frequência (busca em largura).

A abordagem baseada em padrões de crescimento é mais flexível se comparada com as abordagens baseadas no método apriori, pois é possível tanto fazer buscas em largura quanto busca em profundidade. Essa abordagem permite um menor consumo de memória, de acordo com o método de busca empregado, pois não é preciso gerar todo o conjunto de grafos candidatos de mesmo tamanho antes de expandir um grafo. Entretanto, a abordagem apresenta como limitação ser menos eficiente já que um mesmo grafo pode ser gerado mais de uma vez durante o processo de busca. O método inicia com um conjunto de subgrafos iniciais que são expandidos por meio da adição de arestas válidas. Grafos frequentes, com suporte igual ou maior que um valor pré-definido, são selecionados para dar origem à próxima geração de grafos frequentes. A busca continua até que seja gerado um grafo com suporte inferior ao mínimo definido, o último grafo gerado é armazenado para ser apresentado no conjunto de respostas. A busca prossegue com os outros candidatos.

1.2.4.7 Classificação de Grafos

Classificação de grafos é um processo supervisionado de aprendizagem. O objetivo é caracterizar um grafo como todo como uma instância de um conceito. A classificação de

grafos não exige inferência coletiva - como é necessário para classificar vértices e arestas - devido ao grafo ser geralmente gerado de forma independentemente.

Existem basicamente três abordagens que foram exploradas pelas comunidades de mineração de dados: Programação de Lógica Indutiva (PLI); Mineração de Características (MC); e definição de Kernel do Grafo (DK). A MC está relacionada com as técnicas de descoberta de grafo. A MC é usualmente feita encontrando as subestruturas informativas do grafo. Essas subestruturas são usadas para transformar os dados do grafo em uma tabela de dados e então aplicar classificadores tradicionais. A PLI usa relações como vértice (grafo_id, vértice_id, vértice_label, vértice_atributos) e aresta (grafo_id, vértice_id_1,vértice_id_3,aresta_label) para então aplicar em um sistema de PLI para encontrar um hipótese no espaço.

Encontrar todas as subestruturas frequentes em um grafo pode ser um processo computacionalmente proibitivo. Uma abordagem alternativa faz uso dos métodos de *kernel*. Gartner [2002] e Kashima e Inokuchi [2002] proporam métodos baseado em medidas de caminhos no grafo para obter o *kernel*. Gartner [2002] conta caminhos com rótulos iguais no inicio e no fim, enquanto Kashima e Inokuchi [2002] faz uso da probabilidade de caminhar aleatoriamente em uma sequência de rótulos identicos.

1.3 Redes Sociais *Online* e Big Data: Métodos de Tratamento de Grande Volume de Dados

Redes Sociais *Online* (RSOs) como Facebook, YouTube e Twitter estão conectando pessoas que estão produzindo *exabytes* de dados em suas interações [Tan et al. 2013]. O volume, a velocidade de geração e processamento dos dados de diferentes fontes criam grandes desafios isolados ou combinados a serem superados, tais como: armazenamento, processamento, visualização e, principalmente análise dos dados.

A quantidade de dados produzidas na rede aumenta a cada dia e novas unidades de medida surgem para tão grande volume de dados. Para ilustrar este fato, previu-se que o valor chegue próximo a uma dezena de *zettabytes* em 2015 [Oliveira et al. 2013]. Tamanho crescimento faz com que muitas das soluções existentes para manipulação de dados (armazenamento, visualização e transmissão) não sejam úteis nesse cenário. Somadas as RSOs, outras fontes de dados também contribuem para o aumento do volume de dados: sensores, medidores elétricos inteligentes, dados convencionais de aplicações da Internet, dentre outros.

O grande volume de dados heterogêneos produzidos por diferentes fontes autônomas, distribuídas e descentralizadas que geram rapidamente dados com relações complexas e em evolução é chamado *Big Data* [Silva et al. 2013]. O termo *Big Data* é frequentemente associado a 3Vs: i) Volume, relacionado a um grande conjunto de dados; ii) Velocidade, relacionado a necessidade de processo rápido dos dados; e iii) Variedade por provir de fontes diversas de dados [Kwon 2013 *apud* Oliveira 2013].

A grande quantidade de usuários das RSOs tem atraído a atenção de analistas e pesquisadores que desejam extrair ou inferir informações, podendo estar relacionadas a diversas áreas como predição de comportamento, marketing, comércio eletrônico, entre outras interações [Tan et al. 2013]. As análises devem ser eficientes, realizadas quase em tempo real e capazes de lidar com grafos com milhões de nós e arestas. Além disso, existem outros problemas, como falhas e redundâncias.

Esta seção aborda algumas das principais questões relacionadas ao tratamento de grandes massas de dados produzidas nas RSOs mais utilizadas no Brasil e no mundo. São apresentadas discussões sobre termos relacionados à *Big Data*, tecnologias utilizadas e características particulares das RSOs.

1.3.1 Armazenamento e Gerência de Grandes Volumes de Dados de RSO

Quando se fala da gerência dos dados, o volume varia de acordo com a capacidade das ferramentas utilizadas em cada área de aplicação. Por exemplo, as informações de um grafo com milhões de nós e bilhões de arestas podem ser armazenadas em um arquivo de alguns *gigabytes*. O tamanho desse arquivo pode não ser grande do ponto de vista de armazenamento, porém o processamento desse grafo (a aplicação de técnicas de análises) pode exceder a capacidade das ferramentas utilizada com tal finalidade. Percebe-se que, apesar do tamanho ser a parte mais evidente do problema, a definição de *Big Data* deve observar outras características, as quais podem não estar diretamente associados ao tamanho absoluto dos dados [Costa et al. 2012].

Além de observar a capacidade das ferramentas utilizadas, segundo [Costa et al. 2012] há outros pontos a serem observados no cenário de *Big Data*, tais como: a velocidade de geração e de processamento dos dados, além da quantidade de fontes de geração desses dados. Sob este ponto de vista, pode-se citar como exemplo o Twitter com milhões de usuários ao redor do mundo. O Twitter recebe mensagens enviadas em uma frequência muito alta. Apesar de uma mensagem individual ser pequena, a quantidade de mensagens enviadas por diferentes usuários (fontes) gera um grande volume de dados. Cada dado precisa ser armazenado, disponibilizado e publicado para outros usuários dessa mídia. Ou seja, o dado precisa ser armazenado, processado, relacionado a outras informações (que usuários seguem?, quem publicou o dado?) e transmitido. O mesmo pode ser observado em outras RSOs como Facebook ou YouTube.

Outro aspecto a ser observado é a estrutura (ou a sua ausência) dos dados das RSOs, os quais possuem formatos diferentes. O Twitter armazena mensagens textuais pequenas (de 140 caracteres, no máximo), além de outras informações como identificação da mensagem, data da postagem, armazena uma cópia das *hashtags* em um campo específico, posição geográfica do usuário ao enviar a mensagem (quando disponível), entre outros. A própria mídia trata com dados heterogêneos. O Facebook por, outro lado, armazena mensagens textuais, imagens, etc. O YouTube, além dos vídeos, mantém os comentários dos usuários relacionados ao conteúdo de multimídia. Além, dessas informações, essas mídias armazenam dados sobre os usuários, sobre suas interações na rede (seus amigos, curtidas, favoritismo, comentários, citações, dentre outros) e páginas ou canais mais acessados (no caso do Facebook e YouTube, respectivamente). Do ponto de vista da estrutura, percebe-se que as informações dessas redes podem ser armazenadas, parcialmente, em estruturas/formatos e tipos pré-definidos, enquanto outra parte não tem um tipo pré-estabelecido (não são estruturados). Do ponto de vista do relacionamento de dados de diferentes fontes, observa-se que as dificuldades são aumentadas. Um exemplo seria identificar, relacionar e analisar conteúdo dos perfis dos usuários do Twitter e do Facebook.

A observação sobre o conteúdo de mídias sociais permite que se perceba que os dados gerados são diversificados, estão relacionados (ex: um vídeo no YouTube está relacionado aos comentários, curtidas, etc.) e fazem parte de um repositório comum de

cada mídia. Ou seja, o YouTube possui a sua base, o Twitter a sua e assim por diante. Antes da correlação dos dados, é necessário extrair os dados das fontes heterogêneas, cada uma com suas particularidades.

Três exemplos de RSOs cujos dados são utilizados em muitos estudos são o Facebook, o YouTube e o Twitter. Segundo o serviço Alexa[15], no *ranking* dos *sites* mais acessados no mundo, essas três mídias estão entre as 10 mais: O Facebook é o segundo *site* mais acessado no mundo atualmente, o YouTube é o terceiro e o Twitter ocupa o sétimo lugar. No Brasil, o Facebook é o segundo mais acessado, o YouTube o quarto e o Twitter o décimo segundo, segundo o *ranking* do Alexa Brasil[16]. Além dessas mídias digitais, existem outras redes bastante utilizadas no Brasil como LinkedIn[17], Google+[18] e o Foursquare[19].

Os dados das RSOs são abundantes. Para se ter uma ideia, o Facebook é acessado por mais 1 bilhão de usuários a cada mês [Zuckerberg 2012; Facebook Data Center 2013] e registrou uma média de 829 milhões de usuários ativos por dia no mês de junho de 2014 (624 milhões em dispositivos móveis), chegando a passar de 1,32 bilhões no dia 30 de junho [Facebook NewsRoom 2014]. A média de *likes* (curtidas) registrada por dia passa de 2,7 bilhões e quantidade total de itens (texto ou conteúdo multimídia como fotos e vídeos) compartilhados entre amigos é superior a 2,4 bilhões. Em 2011, o espaço ocupado pelas fotos compartilhadas no Facebook já ultrapassava 1,5 petabyte de espaço, sendo mais de 60 bilhões de fotos. Em 2013, o Instagram (aplicativo de compartilhamento de fotos e vídeos curtos, pertencente ao Facebook) registrou uma média de 100 milhões de usuários por mês. Diferentes tipos de relacionamentos acontecem entre os usuários do Facebook formando redes. Por exemplo, rede amizades, citações em mensagens ou marcações em imagens [Facebook NewsRoom 2014].

Outro exemplo que pode ser citado é o YouTube, o qual possui uma taxa de *upload* de vídeo superior a 100 horas de vídeo por minuto, sendo acessado por milhões de usuários mensalmente [ComScore 2014; YouTube Statistics 2014]. O conteúdo de 60 dias do YouTube equivale a 60 de vídeos televisionado pela emissoras norte-americanas NBC, CBS e ABS juntas [Benevenuto et al. 2011]. Dados mais recentes sobre essa mídia informam que mais de 1 bilhão de usuários visitam o YouTube mensalmente, e mais de 6 bilhões de horas de vídeo são assistidas a cada mês. A identificação de mais de 400 anos de vídeo são verificados diariamente devido as buscas por conteúdo e milhões de novas assinaturas feitas todos os meses [YouTube Statistics 2014].

O Twitter, por sua vez, possui mais de 600 milhões de usuários, recebe mais de 500 milhões de mensagens por dia e tem uma média de 271 milhões de usuários ativos por mês. Em julho, o total de atividades registradas era de 646 milhões e 2,1 bilhões de consultas foram realizadas em média. No Twitter, as redes podem ser formadas observando quem segue quem, quem mencionou quem ou quem fez um *retweet* (republicou a mensagem de) quem [Twitter Statistics 2014; About Twitter 2014].

As grandes empresas como Facebook, Google (proprietária do YouTube) e Twitter possuem centros de dados espalhados pelo mundo. Alguns desses centros de

15 http://www.alexa.com/topsites
16 http://www.alexa.com/topsites/countries/BR
17 https://linkedin.com/
18 https://plus.google.com
19 https://foursquare.com/

dados ocupam grandes áreas e custam milhões para serem implantados e mantidos. Por exemplo, o centro de dados do Facebook em Iowa (o quarto da empresa) foi construído após o centro de dados da empresa na Carolina do Norte nos Estados Unidos, o qual custou aproximadamente 450 milhões [Online Tech 2011]. Empresas como as mencionadas acima possuem políticas próprias de gerência dos dados e definem as tecnologias a serem utilizadas. Além disso, elas também impõem uma série de restrições para acesso aos dados da sua base. Por esse motivo, os interessados em realizar análises precisam coletar (geralmente em períodos próximos à publicação), armazenar e gerenciar os dados do seu interesse [Costa et al. 2013]. Um exemplo são os dados publicados pelo Facebook sobre as publicações relacionadas à Copa do Mundo de Futebol de 2014. Segundo dados do próprio Facebook, mais de 1 bilhão de interações (publicações, comentários e curtidas) ocorreram durante este evento [Facebook World Cup 2014]. Aqueles que conseguiram coletar dados sobre esse evento nessa mídia precisam armazenar e gerenciar esses dados (um desafio à parte).

Costa et al. (2012) apresentam uma discussão sobre o ciclo de vida dos dados por meio da comparação com o ciclo de vida biológico. Os autores observaram as seguintes fases: geração (nascimento), agregação (crescimento com a agregação de valores ao dado), análise (reprodução, quando a combinação de novos dados traz significado sobre os dados iniciais) e apagamento (morte). O apagamento pode não ser uma tarefa tão simples, pois não é simples definir quando um conjunto de dados não possui mais valor para ser analisado. Esse valor pode ser finalizado em um contexto, mas sob outros pontos de vista os dados podem possuir valor em novas análises. Por esse motivo, definir quanto tempo os dados devem permanecer armazenados (ou pelo menos parte dele) não é trivial, um dado pode ficar armazenado mais do que o seu valor consumindo recursos valiosos. Porém, descartar um dado valioso por causa das restrições de infraestrutura pode ser lamentável. Finalmente, não é possível definir valores fixos (prazos ou períodos exatos) de validade dos dados. Cabe aquele que gerencia o dado tomar a decisão de descartá-lo ou não. É um consenso que sempre que possível os dados devem ser mantidos (ou seja, a sua remoção deve ser evitada).

Somadas aos desafios de armazenar esses volumes de dados, também existe o desafio de recuperar e analisar os dados dessas mídias digitais. Os problemas relacionados ao armazenamento, recuperação e análise são agravados por novas variações dos dados decorrente das alterações nas mídias digitais ocasionadas por novas tendências, pelo surgimento de novas mídias digitais com características novas e por comportamentos diferentes por parte dos usuários. Vale ressaltar que outras características desses dados são: redundâncias, inconsistências, dados com algum tipo de falha, etc. Todavia, apesar de todas essas dificuldades, as grandes massas de dados impulsionam a necessidade de extrair sentido dos mesmos. Correlacioná-los para compreendê-los apesar das constantes alterações dos dados podem trazer a tona informações preciosas, podendo se tornar essencial no futuro.

Observamos que as tecnologias de bancos de dados utilizados nas RSOs devem ser capazes de atender os requisitos de armazenamento e processamento de *Big Data*, como alta velocidade e a capacidade de lidar com dados não relacionais, executando consultas em paralelo [Oliveira et al. 2013a]. Pesquisadores ou qualquer interessado em analisar esses dados também precisam de tecnologias adequadas. Do ponto de vista do armazenamento, os Sistemas de Gerenciamento de Banco de Dados (SGBDs) convencionais disponíveis comercialmente não são capazes de lidar com volumes de

dados na ordem de *petabytes* [Madden 2012]. Ao observar a velocidade e variedade os sistemas de banco de dados também podem não ter um bom desempenho, sobretudo quando são feitas recuperações textuais, de imagens ou vídeos. A análise de grandes volumes de dados requer SGBDs especializados capazes de processar dados estruturados e não estruturados distribuindo dados a fim de escalar grandes tamanhos [Begoli 2012 *apud* Oliveira et al. 2013a]. Dados como grafos, documentos hierárquicos e dados geo-espaciais são dados úteis para diversos tipos de análise no contexto das RSOs, mas não podem ser modelados em bancos de dados relacionais. Para esses dados existem ferramentas especializadas como: PostGIS [20]e GeoTools[21] para dados geo-espaciais; HBase[22] ou Cassandra[23] para organização hierárquica de dados no formato chave-valor; e um exemplo de ferramenta para analisar grafos é o Neo4j[24].

Porém, mesmos essas tecnologias ainda não são suficientes para suprir todos os desafios citados até aqui. Faz-se necessário explorar devidamente e de forma plena as informações disponíveis nas RSOs. Várias tecnologias estão sendo desenvolvidas e adaptadas para manipular, analisar e visualizar *Big Data*. As seções 1.3.2 e 1.3.3 apresentam uma discussão sobre técnicas e tecnologias apropriadas para se trabalhar com grandes volumes de dados.

Neste trabalho, abordaremos em maiores detalhes o Hadoop (uma implementação do MapReduce) no domínio do armazenamento e análise de dados. O Hadoop permite que aplicações escaláveis sejam desenvolvidas provendo um meio de processar os dados de forma distribuída e paralela [White 2012; Shim 2012 *apud* Oliveira et al. 2013a].

1.3.2 Tratamento de Grande Volume de Dados: Quando Processar se Torna Difícil?

Uma série de desafios vem à tona quando o volume de dados excede os tamanhos convencionais, quando esses dados são variados (diferentes fontes, formatos e estruturas) e são recebidos em uma velocidade maior do que a capacidade de processamento. Por exemplo, ao extrair uma rede de *retweets* do Twitter e formar uma rede a partir desses *retweets* de um grande volume de dados, pode-se obter um grafo que excede a capacidade de tratamento em ferramentas convencionais de análise de redes sociais (como Gephi, por exemplo). Ou quando se deseja realizar processamento de linguagem natural de um texto muito grande a fim de realizar análises estatísticas do texto, o processamento e memória necessários excede a capacidade de computadores pessoais convencionais. Ou seja, os recursos de *hardware* (como a memória RAM, por exemplo) não comportam o volume dos dados [Jacobs 2009].

Jacobs (2009) apresentou um exemplo de difícil tratamento de um grande volume de dados usando um banco de dados relacional com 6,75 bilhões de linhas, com sistema de banco de dados PostgreSQL[25] em uma estação com 20 *megabytes* de memória RAM e 2 *terabytes* de disco rígido. O autor apresentou que obteve vários problemas de falhas e um alto tempo de processamento para as consultas realizadas.

[20] http://postgis.net/
[21] http://www.geotools.org/
[22] http://hbase.apache.org/
[23] http://cassandra.apache.org/
[24] http://www.neo4j.org/
[25] http://www.postgresql.org/

A velocidade do processamento, armazenamento, leitura e transferência de dados nos barramentos frequentemente fazem com que apenas extratos (amostras) dos dados sejam analisados o que não permite que todos os detalhes daquele conjunto de dados sejam observados [DiFranzo et al. 2013]. O desejo dos analistas é estudar as bases de dados por completo, não apenas uma amostra, ou ao menos aumentar as amostras o máximo possível. A necessidade de novas técnicas e ferramentas é reforçada pelo atual interesse em se empregar técnicas de análises que excedam as técnicas tradicionais de *business intelligence.* Extrair conhecimento a partir de grandes massas de dados é de fato desafiador como discutido até aqui, pois além de serem heterogêneos em sua representação, os dados das RSOs são de conteúdo multidisciplinar [Lieberman 2014].

As técnicas convencionais são utilizas em dados estruturados com formatos padronizados. As soluções de *Big Data* tratam com dados brutos, heterogêneos com e sem estrutura e sem padrão. Entender como tratar os desafios de *Big Data* é mais difícil do que entender o que significa o termo e quando empregá-lo. Apesar dos bancos de dados convencionais apresentam bons desempenhos no tratamento de dados estruturados e semiestruturados, as análises no contexto de *Big Data* requerem um modelo iterativo (de consultas recursivas) para análise de redes sociais e emprego de técnicas de clusterização como K-Mean ou PageRank [Silva et al. 2013]. O desafio do processamento dos grandes volumes de dados está relacionado a três aspectos: armazenamento dos dados na memória principal, a grande quantidade de iterações sobre os dados e as frequentes falhas (diferente dos bancos de dados convencionais onde as falhas são tratadas como exceções, no contexto de *Big Data*, as falhas são regras) [Silva et al. 2013].

O processamento intensivo e iterativo dos dados excede a capacidade individual de uma máquina convencional. Nesse contexto, *clusters* (arquiteturas de aglomeração) computacionais possibilitam a distribuição das tarefas e processamento paralelo dos dados. Em alguns cenários, não será possível processar e armazenar todos os dados. Nesse caso, é possível utilizar técnicas de mineração de dados para manipular os dados, sumarizando-os, extraindo conhecimento e fazendo predições sem intervenção humana, visto que o volume dos dados, seus tipos e estruturas não permitem tal intervenção.

Muitas empresas têm apresentado requisitos de gerenciar e analisar grande quantidade de dados com alto desempenho. Esses requisitos estão se tornando cada vez mais comuns aos trabalhos de análise de redes sociais [DiFranzo et al. 2013]. Diferentes soluções têm surgido como proposta para esses problemas. Dentre as propostas, destaca-se o paradigma MapReduce implementado pelo Hadoop, o qual permite o processamento distribuído de grandes conjuntos de dados em *clusters* de computadores [White 2012]. O Hadoop é uma poderosa ferramenta para a construção de aplicações paralelas que fornece uma abstração da aplicação do paradigma do MapReduce para processar dados estruturados e não estruturados em larga escala, sendo essa sua grande vantagem [Silva et al. 2013].

Muitos algoritmos de mineração de dados utilizados na descoberta automática de modelos e padrões utilizam técnicas como classificação, associação, regressão e análise de agrupamento podem ser paralelizados com MapReduce [Shim 2012 *apud* Oliveira et al. 2013]. Segundo Silva et al. (2013), os projetos de mineração de dados no contexto de *Big Data* precisam de três componentes principais. O primeiro é um cenário de aplicação que permita que a demanda por descoberta de conhecimento seja identificada. O segundo é um modelo que realize a análise desejada. O terceiro é uma implementação adequada

capaz de gerenciar um grande volume de dados. Além desses componentes fatores como a complexidade do dado, o tamanho da massa de dados, a dificuldade de transporte dos dados e a possibilidade de paralelização dos algoritmos empregados no processamento devem ser observados.

O processamento do volume de dados variados em tempo hábil exige tecnologias de software e hardware adequados. O Hadoop pode ser utilizado para distribuir e paralelizar dados em diferentes estações de trabalhos comuns, aumentando a capacidade de hardware por meio da clusterização de máquinas comuns. Devido a essas características e as demais utilizadas, o Hadoop tem sido adotado em muitos trabalhos.

1.3.2.1 Capacidade de Processamento de Hardware *vs* Volume de Dados

Como apresentado, o volume de dados na Internet cresce vertiginosamente. Não diferente, as RSOs acompanham esse crescimento à medida que novas mídias surgem e novos usuários começam a participar de RSOs possuindo contas em diferentes redes. As empresas que gerenciam as grandes RSOs possuem centros de dados gigantescos, distribuídos e gerenciam os seus dados contando com equipes especializadas. O sucesso dessas mídias permite que elas alcancem grandes ganhos financeiros, possibilitando-lhes a manutenção dessa estrutura.

Empresas como a IBM, EMC, entre outras, são fornecedores de *hardware* e tecnologias para tratar grandes volumes de dados. Apesar de tratarem com volumes de dados na ordem de *petabytes* ou até *exabytes*, as grandes mídias sociais possuem condições para adquirir, criar e manter tecnologias para armazenar e gerenciar essas grandes massas de dados.

Os analistas que coletam dados nas RSOs para analisá-los, como discutido nas seções anteriores, desejam armazenar, gerenciar e analisar esses dados. Todavia quando o volume dos dados ultrapassa a medida de *gigabytes* e passa a ser medida em *terabytes* ou dezenas de *terabytes*, muitos grupos já se veem na necessidade de excluir partes dos dados ou armazenar parte dos seus dados em tecnologias de mais baixo custo que normalmente tornam o acesso a esses dados mais difíceis. Sob o ponto de vista da análise, outros fatores se somam ao volume, como o tipo de análise que é realizada: grafos com milhões de nós e de centenas de milhões ou bilhões de áreas, processamento de linguagem natural de textos diferentes que exigem grande quantidade de processamento, entre outros exemplos. Nesses casos a dificuldade é saber qual (ou quais) plataforma(s) de hardware se deve utilizar para lidar com grandes massas de dados, os quais superam a capacidade de tratamento possibilitada pelos sistemas tradicionais.

Para serem mais abrangentes (analisar maiores amostras de dados), as análises de redes sociais precisam tratar com amostras que podem ser consideradas *Big Data* sob a ótica apresentada neste trabalho. Essas análises exigem mais capacidade de hardware do que um computador pessoal comum pode oferecer. Por esse motivo, a capacidade do *hardware* utilizado nas análises, assim como os *softwares* utilizados, é um aspecto que não pode ser ignorado.

O simples uso de servidores convencionais, antes empregados para abrigarem banco de dados relacionais, servidores *Web*, sistemas de *intranet*, entre outros, não são adequados para as tarefas de tratamento de enormes quantidades de dados. Estes servidores, ainda que possuam *hardware* superior aos computadores pessoais convencionais, podem ainda não ter a capacidade de *hardware* suficiente para algumas

análises, não fornecendo um rendimento adequado. Aumentar a capacidade desses servidores pode significar sua substituição. Quando adquirir equipamentos e *softwares* específicos para tratar grandes massas dados não são viáveis para os analistas por causa dos custos associados ou por causa da mão de obra para trabalhar com essas soluções, possíveis opções são: usar infraestruturas de nuvens privadas ou por contratação de provedores de serviços por meio do pagamento sob demanda de uso ou contratando pacotes de serviços específicos. Outra opção é distribuir e paralelizar o processamento e armazenamento dos dados que desejam manipular em cluster usando soluções como o Hadoop. A verdade é que qualquer solução exige conhecimento e investimento, porém o que deve ser considerado é o menor esforço e o menor custo de cada possível solução.

As soluções de nuvem exigem, no caso da nuvem privada, esforços por parte dos analistas para criar e manter suas nuvens de dados, além de custos relacionados à compra de ferramentas específicas (quando proprietárias) ou de *hardware*. Quando se contrata um serviço de nuvem de terceiros, além dos custos para pagar esses serviços, existem desafios relacionados à movimentação dos dados (*download* ou *upload*) quando, por exemplo, é necessário manipular dados com ferramentas específicas não fornecidas pelo provedor contratado. No caso de uma solução de *cluster* como o Hadoop, a infraestrutura já existente pode ser aproveitada, fazendo com que a capacidade de *hardware* subutilizada seja direcionada para o processamento e armazenamento de dados. Faz-se necessário elaborar algoritmos adequados de MapReduce, tolerar ruídos e falhas existentes no mundo real. O Hadoop o primeiro passo para a análise dos dados [Silva et al. 2013; Oliveira et al. 2013 a].

1.3.2.2 Processamento Paralelo e Distribuído: Técnicas e Ferramentas

Essa seção visa apresentar algumas técnicas e ferramentas para o processamento distribuído de dados de RSO. Entre elas podemos citar o uso de *clusters*, sensoriamento participativo, computação em nuvem e técnicas de fusão de dados.

Entre as técnicas de processamento distribuído, o sensoriamento participativo ganha destaque. Os telefones celulares, cada vez mais se tornando dispositivos multi-sensores, acumulando grandes volumes de dados relacionados com a nossa vida diária. Ao mesmo tempo, os telefones celulares também estão servindo como um importante canal para gravar as atividades das pessoas em serviços de redes sociais na Internet. Estas tendências, obviamente, aumentam o potencial de colaboração, mesclando dados de sensores e dados sociais em uma nuvem móvel de computação de onde os aplicativos em execução na nuvem são acessados a partir de *thin clients* móveis. Tal arquitetura oferece poder de processamento praticamente ilimitado. Os dois tipos de dados populares, dados sociais e de sensores, são de fato mutuamente compensatórios em vários tipos de processamento e análise de dados. O Sensoriamento Participativo, por exemplo, permite a coleta de dados pessoais via serviços de rede sociais (por exemplo, Twitter) sobre as áreas onde os sensores físicos não estão disponíveis. Simultaneamente, os dados do sensor são capazes de oferecer informações de contexto preciso, levando a análise eficaz dos dados social. Obviamente, o potencial de combinar dados sociais e de sensores é alta. No entanto, eles são normalmente processados separadamente em aplicações em nuvem móvel e o potencial não tem sido investigado suficientemente. Um trabalho que explora essa capacidade é o *Citizen Sensing* [Nagarjaran et al. 2011]. O estudo introduz o paradigma da *Citizen Sensing*, ativada pelo sensor do celular e pelos seres humanos nos

computadores - os seres humanos agindo como cidadãos na Internet onipresente, que atuam como sensores e compartilham suas observações e visão através de *Web* 2.0.

Outra técnica promissora é o uso de Sistemas de fusão de dados escaláveis. Muitos trabalhos buscam o uso de técnicas de fusão de dados distribuídas para o processamento de dados de RSOs. Os autores em [Lovett et al. 2010] apresentam métodos heurísticos e probabilísticos para a fusão de dados que combinam calendário pessoal do usuário com mensagens das RSOs, a fim de produzir uma interpretação em tempo real dos acontecimentos do mundo real. O estudo mostra que o calendário pode ser significativamente melhorado como um sensor e indexador de eventos do mundo real através de fusão de dados.

Outra tendência é o uso da Nuvem (*Cloud*) que está começando a se expandir a partir da aplicação das TIC (Tecnologias da Informação e Comunicação) aos processos de negócios para a inovação, que se destina a aumentar as vendas e otimizar sistemas, identificando informações valiosas através de análise de dados das RSO agregados em nuvens. A inovação torna-se significativamente útil quando é aplicada diretamente no cotidiano das pessoas, auxiliando os usuários a tomarem decisões (por exemplo, que caminho tomar em um dia de engarrafamento [Lauand 2013; Sobral 2013]), e isso torna-se gradualmente claro enquanto os grandes dados coletados são analisados de diversas maneiras. Por essa razão, a análise dos dados deve ser repetida muitas vezes a partir de diferentes perspectivas e é necessária alta velocidade e processamento de baixo custo em todas as fases de desenvolvimento e operação. Os benefícios oferecidos pela nuvem, como a disponibilidade temporária de grandes recursos de computação e de redução de custos através da partilha de recursos têm o potencial de atender a essa necessidade.

Apesar dos *clusters* HPC tradicionais (*High Performance Computing*) serem mais adequados para os cálculos de alta precisão, um *cluster* HPC orientado a lotes ordenados oferece um potencial máximo de desempenho por aplicativos, mas limita a eficiência dos recursos e flexibilidade do usuário. Uma nuvem HPC pode hospedar vários clusters HPC virtuais, dando flexibilidade sem precedentes para o processamento de dados das RSO. Neste contexto, existem três novos desafios. O primeiro é o das despesas gerais de virtualização. A segunda é a complexidade administrativa para gerenciar os *clusters* virtuais. O terceiro é o modelo de programação. Os modelos de programação HPC existentes foram projetados para processadores paralelos homogêneos dedicados. A nuvem de HPC é tipicamente heterogênea e compartilhada. Um exemplo de um *cluster* HPC típico é o projeto Beowulf (2014).

1.3.3 Exemplo Prático: Analisando Dados de RSP Usando Processamento Paralelo e Distribuído com *Hadoop*

Hadoop é uma plataforma de *software* escrita em Java para computação distribuída. Essa plataforma é voltada para armazenar e processar grandes volumes de dados, tem como base o processamento com MapReduce [Dean e Ghemawat 2004] e um sistema de arquivos distribuído denominado Hadoop File System (HFS), baseado no GoogleFS (GFS) [Ghemawat et al. 2003]. Para a análise de redes sociais, o Hadoop apresenta como benefício: i) capacidade de armazenar grandes volumes de dados utilizando *commodity hardware* (no contexto de TI, é um dispositivo ou componente que e relativamente barato e disponível); ii) armazenar dados com formatos variados; iii) além de trazer um modelo de alto nível para processar dados paralelamente.

A arquitetura básica de um *cluster* Hadoop está apresentada na Figura 1.7. Nessa figura, cada retângulo representa um computador em um *cluster* fictício. No primeiro computador, da esquerda para direita, recebe a notação de NameNode, enquanto os demais recebem a denominação de DataNode. O NameNode é responsavel por gerenciar o espaço de nomes do sistema de arquivos e por regular o acesso de arquivos pelos clientes. O DataNode compõe a unidade de armazenamento do *cluster*, onde os arquivos estão distribuídos e replicados. Na imagem também estão apresentados os serviços de JobTracker, que gerencia as tarefas de MapReduce, coordenando sua execução e o TaskTracker que é o serviço de execução de tarefas do MapReduce.

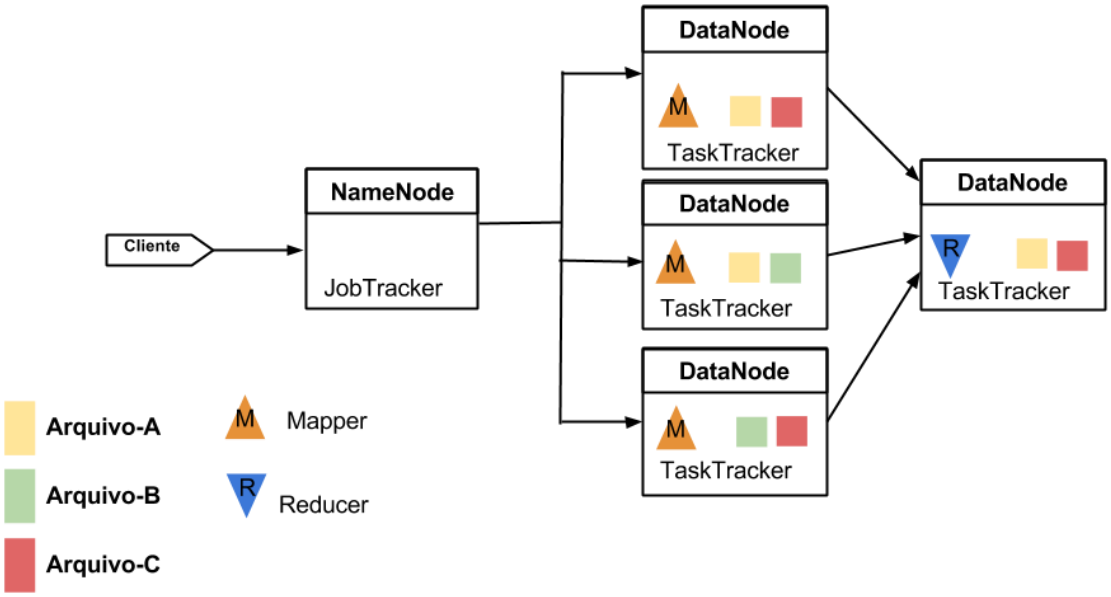

Figura 1.7: Arquitetura básica de um cluster Hadoop, adaptado de [Menon 2013].

O MapReduce é um modelo de programação para processar grandes volumes de dados de forma paralela. Um programa MapReduce é construído seguindo princípios de programaçao funcional para processar lista de dados. Nesse modelo, deve existir uma função de Mapeamento que processa partes de uma lista de dados paralelamente, ou seja, a função de mapeamento é executada por vários nós do *cluster*, processando partes independentes dos dados e uma função de Redução, que recebe os dados dos nós do *cluster* que estão executando a função de mapeamento, combinando-as. Por exemplo, considere um *corpus* textual onde se deseja obter as freqüências das palavras. Um programa em pseudocódigo, descrevendo a função de mapeamento e de redução, pode ser visto na Figura 1.8.

```
function map(String document):
   for each word w in document:
       emit (w, 1)
function reduce(String word, Iterator partialCounts):
 // word: a word
 // partialCounts: a list of aggregated partial counts
 sum = 0
 for each pc in partialCounts:
   sum += ParseInt(pc)
 emit (word, sum)
```

Figura 1.8: Exemplo Hadoop

1.3.3.1 Ecossistema Hadoop

Além da capacidade de armazenar grande volume de dados de forma distribuída, permitir que esses dados apresentem diferentes formatos e prover mecanismos para processamento desses dados por meio de programas MapReduce, o Hadoop apresenta um ecossistema de ferramentas e bibliotecas que auxiliam em tarefas administrativas para o *cluster*, no processamento e análise de dados e no próprio armazenamento de dados. A Figura 1.9 apresenta algumas ferramentas que compõem esse ecossistema.

As ferramentas apresentadas são:

- D3: é uma biblioteca JavaScript para visualização de dados [http://d3js.org/] .
- Tableau: plataforma de visualizaçao e análise de dados proprietária, entretanto, possui versões gratuitas para estudantes e para universidades. [http://www.tableausoftware.com/pt-br]
- Mahout: é um projeto da Apache Software Foundation para produzir implementações livres de algoritmos de aprendizado de máquina escaláveis, focados principalmente nas áreas de filtragem colaborativa, *clustering* e classificação [https://mahout.apache.org/].
- R: ambiente para análise estatísticas [http://www.r-project.org/]
- Java/Python/...: Java é a linguagem oficial para criar programas em um *cluster* Hadoop, entretanto, é possível utilizar outras linguagens como Python e Ruby.
- Pig: é uma plataforma para análise de dados que consiste de uma linguagem de alto nível para expressar uma análise de dados e a infraestrutura para executar essa linguagem. [http://pig.apache.org/]

Figura 1.9: Ecossistema Hadoop, adaptado de [Bidoop Layes 2014]

- Hive: fornece um mecanismo para projetar, estruturar e consultar os dados usando uma linguagem baseada em SQL, chamado HiveQL [http://hive.apache.org/]
- HDFS: Não é uma ferramenta nem uma biblioteca, mas é o cerne da plataforma Hadoop. HDFS é um sistema de arquivos distribuído projetado para ser executado em *commodity hardware*
- HBASE: é um banco de dados orientado a colunas e foi construído para fornecer pedidos com baixa latência sob Hadoop HDFS. [http://hbase.apache.org/]
- MongoDB: banco de dados orientado a documentos no formato JSON. [http://www.mongodb.org/]
- Kettle: ferramenta de ETL que permite tratamento de dados construindo *workflows* gráficos. [http://community.pentaho.com/projects/data-integration/]
- Flume: ferramenta de coleta e agregação eficiente de *streams* de dados. [http://flume.apache.org/]
- Sqoop: ferramenta que permite a transferência de dados entre bancos relacionais e a plataforma Hadoop. [http://sqoop.apache.org/]
- Chukwa: sistema de coleta de dados para monitoramento de sistemas [https://chukwa.apache.org/]
- Oozie: é um sistema gerenciador de *workflows* para gerenciar tarefas no Hadoop. [http://oozie.apache.org/]
- Nagios: ferramenta para monitorar aplicativos e redes [http://www.nagios.org/]
- Zoo Keeper: é um serviço centralizador para manter informações de configuração. [http://zookeeper.apache.org/]

1.3.3.2 Exemplo de Análise de Redes Sociais

Como exemplo será apresentado um algoritmo baseado em MapReduce para contagem de frequência em grafos. Algoritmos como APRIORI necessitam contar a frequência de cada subgrafo em uma base de grafos. Como esse processo é computacionalmente custoso e deve ser executado em cada etapa do APRIORI, abaixo está apresentado um exemplo simplificado para contagem de frequência. Considere uma base de Grafos D e a sua representação como uma lista de vértices e arestas como exemplificado na Figura 1.10.

```
1 a,b,c,d,e,f,g | ab,de,gf,dc
2 h,i,c,d,e,f,n | hc,dc,ef, en
…
N a,g,h,j,n,r,h | ag,ah,aj,rh
```

Figura 1.10: Base de grafos D, contendo N grafos. O formato da base é id do grafo, vértices, arestas.

```
#MAP
import sys
sub_graf = #representação do subgrafo aqui
for ling in sys.stdin:
  is_subgraph = True
  lista_vertices, lista_arestas = parser_grafo(line) # transformando o arquivo do grafo em uma lista de vértices e outra lista de arestas
  #checando se todos os vértices de sub_graf_input são vértices em graf
  for v in sub_graf.vertices:
    if v not in lista_vertices:
      is_subgraph = False
      break
  #checando se todas arestas de sub_graf_input são arestas em graf
  if is_subgraph:
    for a in sub_graf.arestas:
      if a not in lista_arestas:
        is_subgraph = False
  if is_subgraph:
    print "%d\t%d"%(sub_graf.id, 1)
  else:
    print "%d\t%d"%(sub_graf.id, 0)
```

Figura 1.11: Algoritmo de mapeamento.

O primeiro passo é criar o algoritmo para mapeamento. Ele deve processar um conjunto de subgrafos, com representação idêntica da Figura 1.11, comparando cada um com um dos subgrafos representados na variável sub_graf. Esses subgrafos poderiam ser provenientes de arquivos ou outras estruturas de armazenamento, mas foram postos diretamente no código apenas para simplicar o exemplo. Para definir se um grafo é subgrafo de outro está sendo feito a comparação dos vértices e das arestas. Quando todos vértices e arestas estão contidos no grafo, o algoritmo de mapamento retorna o id do subgrafo e o valor 1. Quando existe uma ou mais arestas ou vértices que não fazem parte do grafo, o algoritmo retorna o id do sub_grafo e valor 0.

O algoritmo de redução agrega os resultados das tarefas de mapeamento, retornando o somatório dos valores para cada id do subgrafo (Figura 1.12).

```
#Reducer
import sys

current_graph = None
current_count = 0
subgraph_id = None
for line in sys.stdin:
   # removendo espaços em branco
   line = line.strip()

   # parseando o resultado produzido pelo mapper
   subgraph_id, count = line.split('\t')

   #convertento o contador para um inteiro
   count = int(count)

   #conta enquanto existirem valores para serem reduzidos
   if current_graph == subgraph_id:
      current_count += count
   else:
      #imprime o resultado caso outro id de sub_grafo esteja sendo processado
      if current_graph:
         print '%s\t%s' % (current_graph, current_count)
      current_count = count #zerando o contador
      current_graph = subgraph_id #atribuindo o novo id de sub_grafo
#necessário para imprimir o ultimo resultado
if current_graph == subgraph_id:
   print '%s\t%s' % (current_graph, current_count)
```

Figura 1.12: Reducer.

1.4 Principais Desafios e oportunidades de pesquisa

Analisar grandes volumes de dados extraídos das redes sociais *online* permite que novas informações sejam obtidas, as quais não eram possíveis de serem verificadas devido às amostras desses tipos de dados ser menor. Porém, o aumento dos dados a serem analisados somam novos desafios aos já existentes na área de análise de redes sociais. Agora esses desafios são tanto do ponto de vista da análise de redes sociais quanto do ponto de vista do avanço das tecnologias de *Big Data*. O aumento das massas de dados das redes sociais está fazendo com que as técnicas, metodologias e ferramentas de mineração de dados e análise de grafos sejam adaptadas, melhoradas ou soluções novas sejam criadas.

Alguns dos principais desafios (que trazem novas oportunidades de pesquisa) são:

- Algoritmos adequados e escaláveis para milhões ou até bilhões de elementos a serem analisados;

- Algoritmos que possam ser distribuídos, paralelizados e capazes de tratar de ruídos e falhas;
- Algoritmos que permitam análises rápidas de grandes massas de dados, sendo a análise quase em tempo real;
- Segurança dos dados, no contexto das RSOs, principalmente privacidade;
- Diminuir o consumo de recursos necessário para armazenar, gerenciar, processar e enviar grandes massas de dados;
- Segurança e confiabilidade da informação (publicação de informações íntimas por usuários leigos, geração e propagação de boatos nas RSOs, etc.);
- Desafios relacionados à multidisciplinaridade dos dados das RSOs que exigem conhecimentos de diferentes áreas do conhecimento sendo, quase sempre, necessário que profissionais de diferentes áreas consigam interagir e colaborar nessas análises;
- Analisar mensagens não estruturadas como análises de linguagem natural, de imagens e de vídeos; e
- Estruturas físicas para armazenar dados fornecendo acesso rápido aos mesmos.

1.5 Conclusão

A quantidade de dados produzidos na Internet aumenta diariamente. Novas aplicações usadas na rede, aliadas às aplicações existentes e ao aumento do uso de sensores e dispositivos eletrônicos (medidores elétricos, por exemplo) aumentam cada vez mais a quantidade de dados produzidos. As redes sociais *online* seguem essa tendência. À medida que novas mídias digitais surgem e se popularizam, novas funcionalidades são adicionadas as mídias e novos usuários participam dessas redes, levando ao aumento da quantidade de dados oriundos de interações sociais. As informações das RSOs são multidisciplinares, em grandes quantidades, produzidos rapidamente e em diferentes fontes.

Esses dados, produzidos em grande volume, velocidade e de fontes variadas precisam ser armazenados, gerenciados e possivelmente analisados sob diferentes óticas para geração de novos conhecimentos. *Big data* é o termo empregado para esse grande volume de dados oriundos de fontes heterogêneas, produzidos, transmitidos e processados em altas velocidades.

O volume de conteúdo produzido e compartilhado nas redes sociais *online*, associado ao grande número de usuários (cidadãos de diferentes localidades), é fonte de diversas informações que se propagam e agregam novos valores às informações de diversas áreas. Atualmente, analisar essa grande massa de dados é um desafio, visto que as ferramentas utilizadas para mineração de dados, estudos de grafos, entre outras, podem não ser adequadas para tratar com grandes volumes de dados.

Este trabalho apresentou uma discussão sobre tecnologias e abordagens para análises de redes sociais *online*, contextualizou o problema de análise de grandes volumes de dados, abordou as principais abordagens existentes para se trabalhar com esses dados e apresentou um exemplo prático de análise de grandes volumes de dados extraídos de

redes sociais *online*. Trabalhar com amostras maiores de dados possibilita que informações antes ocultas sejam aproveitadas e tragam novas e melhores informações.

Ainda existem muitos desafios a serem enfrentados, porém a possibilidade de trabalhar com amostras maiores de dados das redes sociais *online* permite que novas informações sejam extraídas e que informações antes obtidas sejam mais consistentes, visto que a amostra analisada será maior. Além do desafio técnico de analisar grandes quantidades de dados, novos desafios surgem a partir dessa nova oportunidade de análise, visto que novas informações que antes não eram consideradas devido às limitações técnicas e humanas podem e devem ser agora consideradas. Tratar essas novas informações adequadamente extrapola as áreas técnicas da computação (e até da área das ciências exatas) visto que conhecimentos de áreas de humanas (como antropologia, sociologia, psicologia, entre outros) são necessários. Este trabalho introduziu o tema de *Big Data* e análise de redes sociais, permitindo que pesquisadores e analistas de redes sociais que desejam trabalhar com grande volume de dados conheçam as principais abordagens e desafios que existem atualmente. Ao mesmo tempo, este trabalho também pode ser utilizado por profissionais que estão trabalhando com *Big Data* e desejam agora analisar dados de redes sociais.

Referências

About Twitter (2014) "Our mission: To give everyone the power to create and share ideas and information instantly, without barriers", Disponível em: https://about.twitter.com/company, Acessado em: 19 de julho de 2014.

Albuquerque, R. P., Oliveira, J., Faria, F. F., Studart, R. M., Souza, J. M.(2014), "Studying Group Dynamics through Social Networks Analysis in a Medical Community", Social Networking, v. 03, p. 134-141.

Alonso, O., Ke, Q., Khandelwal, K., Vadrevu S. (2013), "Exploiting Entities In Social Media", Proceedings of the sixth international workshop on Exploiting semantic annotations in information retrieval (ESAIR), ACM, p. 9-12.

Appel, A. P., Hruschka, E.(2011), "Por dentro das redes complexas: detectando grupos e prevendo ligações", Anais do XXVI Simpósio Brasileiro de Banco de Dados, SBC.

Bidoop Layer (2014), "Soluções em Big Data Baseadas em Hadoop", Disponível em: http://www.bidoop.es/bidoop_layer, Acessado em: 10 de maio de 2014.

Benevenuto, F., Almeida, J., Silva, A. S. (2011), "Explorando Redes Sociais Online: Da Coleta e Análise de Grandes Bases de Dados as Aplicacões", Minicurso do XXVI Simpósio Brasileiro de Redes de Computadores, SBC.

Beowulf (2014), "The Beowulf Archives", Disponível em: http://www.beowulf.org/, Acessado em: 14 de fevereiro de 2014.

Castells, M. (1996), "Rise of the Network Society: The Information Age: Economy, Society and Culture", Vol. 1, John Wiley & Sons.

Chen, H. (2001), "Knowledge management systems: a text mining perspective", Arizona: Knowledge Computing Corporation.

ComScore (2014), "comScore Releases March U.S. *Online Video Rankings*". Disponível em: https://www.comscore.com/por/Insights/Press-Releases/2014/4/comScore-Releases-March-2014-US-Online-Video-Rankings, Acessado em 19 de julho de 2014.

Costa, L. H. M. K., Amorim, M. D., Campista, M. E. M., Rubinstein, M. G., Florissi, P., Duarte, O. C. M. B. (2012), "Grandes Massas de Dados na Nuvem: Desafios e Técnicas para Inovação", Minicurso do XXX Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos, SBC, p. 58.

Cozza, R., Milanesi, C., Gupta, A., Nguyen, T. H., Lu, C. K., Zimmermann, A., & De La Vernge, H. J. (2011), "Market Share Analysis: Mobile Devices", Gartner Report, Disponível em: http://www.gartner.com/newsroom/id/1689814, Acessado em 20 de julho de 2014.

Dale, C., Cheng, X., Liu. J. (2007), "Understanding the Characteristics of Internet Short Video Sharing : YouTube as a Case Study", Technical Report arXiv:0707.3670v1, Cornell University.

Dean, J., Ghemawat, S. (2008). "MapReduce: simplified data processing on large clusters". Communications of the ACM, v. 51, n. 1, p. 107-113.

DiFranzo, D., Zhang, Q., Gloria, K., Hendler, J. (2013). "Large Scale Social Network Analysis Using Semantic Web Technologies", AAAI Fall Symposium Series.

Easley, D., Kleinberg, J. (2010). "Networks, crowds, and markets: Reasoning about a highly connected world", Cambridge University Press.

Facebook (2014), "Public Feed API", Disponível em: https://developers.facebook.com/docs/public_feed, Acessado em 21 de janeiro de 2014.

Facebook Data Center (2014), "A New Data Center for Iowa", Disponível em: https://newsroom.fb.com/news/2013/04/a-new-data-center-for-iowa/, Acessado em 20 de julho de 2014.

Facebook NewsRoom (2014), "NewsRoom", Disponível em: http://newsroom.fb.com/company-info/, Acessado em 20 de julho de 2014.

Facebook World Cup (2014), "World Cup 2014: Facebook Tops A Billion Interactions", Disponível em: https://newsroom.fb.com/news/2014/06/world-cup-2014-facebook-tops-a-billion-interactions/, Acessado em: 20 de julho de 2014.

Gärtner, T. (2002), "Exponential and geometric kernels for graphs", NIPS Workshop on Unreal Data: Principles of Modeling Nonvectorial Data, Vol. 5, pp. 49-58.

Getoor, L., Diehl, C. P. (2005), "Link mining: A survey", ACM SIGKDD Explorations Newsletter, v. 7, n. 2, p. 3–12.

Ghemawat, S., Gobioff, H., Leung, S. T. (2003), "The Google file system", ACM SIGOPS Operating Systems Review, ACM , v. 37, n. 5, p. 29-43.

Girvan, M., Newman, M. (2002), "Community structure in social and biological networks", Proceedings of the National Academy of Sciences, v. 99, n. 12, p. 7821-7826.

Gomide, J., Veloso, A., Meira Jr, W., Almeida, V., Benevenuto, F., Ferraz, F., Teixeira, M. (2011), "Dengue surveillance based on a computational model of spatio-temporal

locality of Twitter", Proceedings of the 3rd International *Web* Science Conference, ACM, p. 3-11.

Hanneman, R. A., Riddle, M., "Introduction to social network methods", Disponível em: http://faculty.ucr.edu/~hanneman/, Acessado em 20 de julho de 2014.

Huisman, M., Van Duijn, M. A. J. (2005), "Software for Social Network Analysis". In Carrington, P .J., Scott, J., ,Wasserman, S. (Editors), "Models and Methods in Social Network Analysis", New York: Cambridge University Press, p. 270-316.

Jacobs, A. (2009) . "The pathologies of Big Data". Magazine Communications of the ACM - A Blind Person's Interaction with Technology, New York, NY, USA. v. 52, n. 8, p. 36-44.

Junior, E. A. S., Oliveira, J. (2013), "Hermes: Identificação de Menores Rotas em Dispositivos Móveis". Anais do XXVIII Simpósio Brasileiro de Banco de Dados - Demos e Aplicações, Pernambuco, SBC.

Kashima, H., Inokuchi, A. (2002), "Kernels for graph classification". Proceedings of ICDM Workshop on Active Mining.

Kempe, D., Kleinberg, J., Tardos, E. (2003), "Maximizing the Spread of Influence through a Social Network", Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, pp. 137-146.

Kleinberg, J. M. (1999), "Authoritative sources in a hyperlinked environment", Journal of the ACM, v. 46, n. 5, p. 604-632.

Krauss, J., Nann, S., Simon, D., Gloor, P. A., Fischbach, K. (2008), "Predicting movie success and academy awards through sentiment and social network analysis", Proceedings of European Coferance on Information Systems, p 2026–2037.

Lafferty, J., McCallum, A., Pereira, F. C. N. (2001), "Conditional random fields: Probabilistic models for segmenting and labeling sequence data", Proceedings of XVIII International Conference on Machine Learning, Morgan Kaufmann Publishers, p. 282-289.

Lam, C. (2010), "Hadoop in action", Manning Publications Co.

Lauand, B.; Oliveira, J. (2013), "TweeTraffic: ferramenta de análise das condições de trânsito baseado nas informações do Twitter". Anais do II Brazilian Workshop on Social Network Analysis and Mining (BraSNAM), SBC.

Liben-Nowell, D., Kleinberg, J. (2007), "The link-prediction problem for social networks", Journal of the American Society for Information Science and Technology, v. 58, n. 7, p. 1019–1031.

Lieberman, M., "Visualizing Big Data: Social network analysis", Disponível em: http://mvsolution.com/wp-content/uploads/Visualizing-Big-Data-Scial-Network-Analysis-Paper-by-Michael-Lieberman.pdf, Acessado em 20 de julho de 2014.

Lu, Q., Getoor, L. (2003), Link-based Classification, Proceedings of XX International Conference on Machine Learning, v. 20, n. 2, p. 1–42.

Nagarajan, M., Sheth, A., Velmurugan, S. (2011), "Citizen sensor data minning, social media analytics and development centric Web applications". Proceedings of the 20th international conference companion on World Wide Web, ACM, pp. 289-290.

Manovich, L. (2011), "Trending: the promises and the challenges of big social data", Minneapolis, MN: University of Minnesota Press.

Melo, H., Oliveira, J. (2014), "Ambiente Analítico Web para Análise da Colaboração Científica no Cenário Médico", Anais do X Simpósio Brasileiro de Sistemas de Informação, SBC, p.387-398 .

Menon, R. (2013), "Introducing Hadoop – Part II", Disponível em: http://rohitmenon.com/index.php/introducing-hadoop-part-ii/, Acessado em: 10 de dezembro de 2013.

Mikolajczyk, R. T., Kretzschmar, M. (2008), "Collecting social contact data in the context of disease transmission: Prospective and retrospective study designs", Social Networks, Elsevier, v. 30, n. 2, p. 127-135.

Monclar, R.S., Oliveira, J., Faria, F.F., Ventura, L.V.F., Souza, J. M., Campos, M.L.M. (2012), "The Analysis and Balancing of Scientific Social Networks in Cancer Control", Handbook of Research on Business Social Networking: Organizational, Managerial and Technological Dimensions, IGI Global, p. 915-941.

Nakamura, E. F., Loureiro, A. A. F., Frery, A. C. (2007), "Information fusion for wireless sensor networks: Methods, models, and classifications", Computing Surveys, ACM, v. 39, n. 3, p. 9-64.

Neto, B.; Oliveira, J.; Souza, J. M. (2010). "Collaboration in Innovation Networks: Competitors can become partners", Proceedings of International Conference on Information Society, IEEE, p. 455-461.

Newman, M. E. J. (2010), "Networks: An Introduction", Oxford: Oxford University Press.

Nodejs (2014), "Node.js", Disponível em: http://nodejs.org, Acessado em: 20 de junho de 2014.

O'Madadhain, J., Hutchins, J., Smyth, P. (2005), "Prediction and ranking algorithms for event-based network data", ACM SIGKDD Explorations Newsletter, 7(2), 23-30. Oliveira, A. C., Salas, P. R., Roseto S., Boscarioli C., Barbosa W., Viterbo A. (2013a), "Big Data: Desafios e Técnicas para a Análise Eficiente de Grandes Volumes e Variedades de Dados", Minicurso do IX Simpósio Brasileiro de Sistemas de Informação, SBC.

Oliveira, J., Santos, R. P. (2013b), "Análise e Aplicações de Redes Sociais em Ecossistema de Software", Minicurso do IX Simpósio Brasileiro de Sistema de Informação.

Online Tech (2014), "Cloud computing prompts 2012 data center expansion plans", Disponível em: http://resource.onlinetech.com/cloud-computing-prompts- 2012-data-center-expansion-plans/, Acessado em: 19 de julho de 2014.

Pastor-Satorras, R., Vespignani, A. (2001), "Epidemic spreading in scale-free networks", Physical Review Letters, v. 86, n. 14, p. 3200-3203.

Popescul, A; Ungar, L. H. (2003), "Statistical Relational Learning for Link Prediction", Proceedings of IJCAI workshop on learning statistical models from relational data.

Ranking, T. P. C., Order, B. (1998), "The PageRank Citation Ranking: Bringing Order to the Web". Technical Report, Stanford University, Disponível em http://ilpubs.stanford.edu:8090/422/1/1999-66.pdf , Acessado em: 23 de setembro de 2014.

Santos, F. B., Oliveira, J. (2014), "More than Just a Game: The Power of Social Media on Super Bowl XLVI", Social Networking, Scientific Research, v. 03, p. 142-145.

Silva, I. S., Gomide, J., Barbosa, G. A. R., Santos, W., Veloso A., Meira, W. Jr., Ferreira, R. (2011). "Observatório da Dengue: Surveillance based on Twitter Sentiment Stream Analysis". XXVI Simpósio Brasileiro de Banco de Dados-Sessão de Demos.

Silva, T. L. C., Araújo, A. C. N., Sousa, F. R. C., Macêdo, J. A. F., Machado, J. C. (2013), "Análise em Big Data e um Estudo de Caso utilizando Ambientes de Computação em Nuvem". Minicurso do XXVII Simpósio Brasileiro de Banco de Dados.

Solis, B. (2007), "Manifesto, The Social Media", Disponível em: http://www.briansolis.com/2007/06/future-of-communications-manifesto-for/ , Acessado em: 20 de junho 2014.

Souza, J. M., Neto, B., Oliveira, J. (2011), "Innovation Networks as a Proposal to Overcome Problems and Improve Innovation Projects". International Journal for Infonomics, Infonomics Society, v. 4, p. 623-632.

Statistics | Facebook (2014), Disponível em: http://www.facebook.com/press/info.php?statistics, Acessado em: 20 de junho de 2014.

Stempel, G. H., Hargrove, T., Bernt, J. P. (2000), "Relation of Growth of Use of the Internet to Changes in Media Use from 1995 to 1999", Journalism & Mass Communication Quarterly, SAGE Journals, v. 77, n. 1, p. 71-79.

Stroele, V., Silva, R., Souza, M.F., Mello, C. E., Souza, J. M., Zimbrao, G., Oliveira, J. (2011), "Identifying Workgroups in Brazilian Scientific Social Networks", Journal of Universal Computer Science, v. 17, p. 1951-1970.

Studart, R. M.; Oliveira, J.; Faria, F.F.; Ventura, L.V.F.; Souza, J. M.; Campos, M.L.M. (2011), "Using social networks analysis for collaboration and team formation identification", Proceedings of XV International Conference on Computer Supported Cooperative Work in Design, IEEE, p. 562-569.

Svenson, P., Svensson, P., Tullberg, H. (2006), "Social Network Analysis And Information Fusion For Anti-Terrorism", Proceedings of Conference on Civil and Military Readiness, Paper S3.1.

Lovett, T., O'Neill, E., Irwin, J., Pollington, D. (2010). "The calendar as a sensor: analysis and improvement using data fusion with social networks and location" Proceedings of XXII International Conference on Ubiquitous Computing, ACM, p. 3–12.

Tan, W., Blake, M. B., Saleh, I., Dustdar, S. (2013), "Social-Network-Sourced Big Data Analytics". Internet Computing. IEEE Computer Society, v. 17, n. 5, p. 62-69.

Taskar, B., Abbeel, P., Koller, D. (2002), "Discriminative probabilistic models for relational data", Proceedings of the Eighteenth conference on Uncertainty in artificial intelligence, p. 485-492

Twitter Statistics (2014), "Twitter Statistics Verification", Disponível em: <http://www.statisticbrain.com/twitter-statistics/>, Acessado em: 19 de julho de 2014.

Wasserman, S., Faust, K. (1994), "Social Network Analysis: Methods and Applications", Cambridge University Press.

Watts, D. J.; Strogatz, S. H. (1998), "Collective dynamics of small-world networks", Nature, v. 393, n. 6684, p. 440-442.

White, T. (2009), "Hadoop: The Definitive Guide", O'Reilly Media.

Yang, C.C., Ng, T.D. (2007), "Terrorism and Crime Related Weblog Social Network: Link,Content Analysis and Information Visualization", Intelligence and Security Informatics, IEEE. p. 55-58.

YouTube Statistics (2014), Disponível em: https://www.youtube.com/yt/press/statistics.html, Acessado em: 19 de julho de 2014.

Zikopoulos, P., Eaton, C. (2011), "Understanding Big Data: Analytics for enterprise class Hadoop and streaming data", McGraw-Hill Osborne Media.

Zuckerberg, M. (2014), "One Billion People on Facebook", Disponível em: https://newsroom.fb.com/news/2012/10/one-billion-people-on-facebook/, Acessado em: 20 de julho de 2014.

Zudio, P., Mendonca, L., Oliveira, J. (2014), "Um método para recomendação de relacionamentos em redes sociais científicas heterogêneas" , Anais do XI Simpósio Brasileiro de Sistemas Colaborativos, SBC.

Sobre os Autores

Tiago Cruz França é professor assistente da Universidade Federal Rural do Rio de Janeiro (UFRRJ) e aluno de doutorado no Programa de Pós-Graduação em Informática, onde desenvolve pesquisas nas áreas de análise de redes sociais e *big data*. Tem interesse em tecnologias *Web*, Engenharia de *Software* e Análise de Redes Sociais. Nos últimos anos, Tiago tem atuado nos seguintes temas: Serviços *Web*, *Mashups Web*, *Web* das Coisas, Análise de Sentimentos, Fusão de Dados *Web* e dados oriundos de dispositivos inteligentes.

Fabrício Firmino de Faria é professor substituto da Universidade Federal do Rio de Janeiro (UFRJ), possui mestrado em Informática pela Universidade Federal do Rio de Janeiro, durante o qual realizou intercâmbio no Digital Enterprise Research Institute (DERI, Irlanda). Atualmente atua em pesquisas com *Web* Semântica, *Data Warehousing*, Análises de Redes Sociais e *Big Data*. Nos últimos anos trabalhou com processamento de linguagem natural para análise de dados textuais e com o desenvolvimento de plataformas para captura e armazenamento de dados produzidos por sensores.

Fabio Medeiros Rangel é graduando da Universidade Federal do Rio de Janeiro, possui experiência em Análise de Redes Sociais, com foco em visualização de dados e desenvolvimento de algoritmos para cálculo de métricas em ambientes distribuídos. Possui interesse nas áreas de *Data Mining* e *Big Data*.

Claudio Miceli de Farias possui graduação em Ciência da Computação pela Universidade Federal do Rio de Janeiro (2008), mestrado (2010) e doutorado (2014) em Informática pela Universidade Federal do Rio de Janeiro. Atuou como professor substituto no Departamento de Ciência da Computação da UFRJ durante o período de 2012 a 2014. Atualmente é professor do Colégio Pedro II e professor visitante no laboratório de Redes e Multimídia do iNCE-UFRJ. Atua como revisor no SBSEG e SBRC. É também membro do comitê de programa das conferências Wireless Days e IDCS. As principais áreas de atuação são: Redes de Sensores sem Fio, Redes de Sensores Compartilhadas, Fusão de Dados, Escalonamento de tarefas, *Smart Grid*, Análise de Dados e Segurança.

Jonice Oliveira obteve o seu doutorado em 2007 em Engenharia de Sistemas e Computação, ênfase em Banco de Dados, pela COPPE/UFRJ. Durante o seu doutorado recebeu o prêmio IBM Ph.D. Fellowship Award. Na mesma instituição realizou o seu Pós-Doutorado, concluindo-o em 2008. Atualmente é professora adjunta do Departamento de Ciência da Computação da UFRJ, coordenadora do curso de Análise de Suporte à Decisão (habilitação do Bacharelado em Ciências da Matemática e da Terra) e atua no Programa de Pós-Graduação em Informática (PPGI-UFRJ). Em 2013, tornou-se Jovem Cientista do Nosso Estado pela FAPERJ. Coordena o Laboratório CORES (Laboratório de Computação Social e Análise de Redes Sociais), que conduz pesquisas multidisciplinares para o entendimento, simulação e fomento às interações sociais. É coordenadora de Disseminação e Parcerias do Centro de Referência em Big Data, da UFRJ. Suas principais áreas de pesquisa são Gestão do Conhecimento, Análise de Redes Sociais, Big Data e Computação Móvel.