INSTITUTO DE GESTÃO E TECNOLOGIA DA INFORMAÇÃO

Coleta e Obtenção de Dados

Fernanda Farinelli

2020

Coleta e Obtenção de Dados

Fernanda Farinelli

Sumário

Capítulo 1. Conceitos Fundamentais

O Capítulo 1 apresenta alguns conceitos fundamentais que são relevantes para compreensão de diversas questões relacionadas com a formação em análise de dados.

A análise de dados ou *Data Analytics* é uma forma de retroalimentar os planejamentos e iniciativas da empresa para que sinais, indicativos e insights se transformem em insumos para a definição dos rumos do empreendimento. Entretanto, para que seja realizada análise de dados, primeiro deve-se selecionar ou coletar os dados que serão usados na análise.

1.1. Conceito de dados

Nas últimas três décadas, os dados assumiram um papel vital na estratégia das empresas, tornando-se um dos grandes ativos existentes no patrimônio das organizações (DAMA, 2009, p. 1). No final dos anos 90 e até meados da década seguinte, muitos pesquisadores se preocuparam em definir o que são dados, e avançavam nesta definição ao diferenciar dados de informação, conhecimento, porém não existe propriamente um consenso quanto à sua diferenciação ou definição. Para simplificar, cita-se as definições de Davenport (1998), conforme o Quadro 1, que demonstra a complementaridade entre esses três conceitos.

Quadro 1 - Dados, informação e conhecimento.

Dado	Informação	Conhecimento
Simples observações sobre o estado do mundo	Dados dotados de relevância e propósito	Informação valiosa da mente humana
Facilmente estruturado Facilmente obtido por máquinas Frequentemente quantificado Facilmente transferível	Requer unidade de análise Exige consenso em relação ao significado Exige necessariamente a mediação humana	Inclui reflexão, síntese, contexto De difícil estruturação De difícil captura em máquinas Frequentemente tácito De difícil transferência

Fonte: Retirado de DAVENPORT (1998, p. 18).

Tal definição enfatiza o papel dos dados em representar fatos sobre o mundo, pois se baseiam na definição da palavra latina *datum* que significa “um fato”, ou seja, *dados são fatos capturados, armazenados e expressos como texto, números, gráficos, imagens, sons ou vídeos* (DAMA, 2009; 2017). Na área de TI, os dados são armazenados em formato digital e não se resumem a fatos, podem ser objetos, localizações, quantidades, textos, imagens e áudios, ou qualquer coisa que possa ser digitalizada e consequentemente armazenado em alguma tecnologia de bancos de dados (DAMA, 2017).

Os dados podem ser obtidos pela digitalização de fatos, observações, medições, cadastro e outros. Este processo de transformação de “tudo” em um formato digital é conhecido como dataficação ou datificação (MAYER-SCHÖNBERGER; CUKIER, 2013). Dataficação (*datafication*) refere-se à capacidade de tomar informações sobre todas as coisas e transformá-las em um dado formato para torná-lo quantificado, permitindo assim monitoramento e análise. O processo de dataficação envolve ainda o uso de tecnologias digitais para adquirir os dados e tecnologias de análise de dados para gerar conhecimento (LYCETT, 2013; MAYER-SCHÖNBERGER, CUKIER, 2013; VAN DIJCK, 2014).

Atrelado ao processo de dataficação, podemos considerar a evolução da internet e dos diversos dispositivos que hoje funcionam conectados à rede, que vão desde as redes sociais, fotos, áudios, vídeos e cada vez mais dispositivos “inteligentes” conectados à internet que compõe o que chamamos de internet das coisas (IoT, do inglês *internet of things*) (ATZORI et al., 2010; GUBBI et al., 2013; MIORANDI et al., 2012).

Tipos de dados

A disponibilidade dos dados oferece oportunidades para a obtenção de informações, ou seja, ao submeter os dados a processos de análise obtém-se informação que pode ser útil nos processos decisórios das organizações. Observamos no nosso dia a dia que negócios em geral se tornam negócios de dados. Imagens, palavras, localizações, tempo, compromissos, entretenimento, interações em geral se tornam dados. Dados podem ser nomes, endereços, datas de

nascimento, operações de compra e venda, produtos e sua respectiva descrição, a quantidade de produtos em estoque, saldos em contas e aplicações, comentários feitos em redes sociais, vídeos e áudios gravados, etc.

Acontece que a forma como os dados são gerados, processados e armazenados mudou drasticamente ao longo das décadas principalmente nos últimos anos. Como consequência, os dados passaram a assumir diferentes formatos ou tipos. Os tipos de dados encontrados atualmente são estruturados, semiestruturados e não estruturados:

Dados Estruturados: São assim denominados pois possuem uma organização ou estrutura para serem armazenados e recuperados definida previamente. Estes dados que se caracterizam por apresentar um formato ou esquema de representação e com um domínio pré-definido, com tamanho e formato definidos. Como exemplo geral, temos os dados organizados nos bancos de dados relacionais. A Figura 1 apresenta um modelo de dado estruturado, referente à tabela Cliente que contém o cadastro dos clientes. Observe que a tabela representa uma estrutura prévia definida.

Figura 1 - Exemplo de dado estruturado.

Tabela Cliente

CPF	Nome	Telefone	UF_Nascimento
15745265423	Joselito Gomes	98532615	SP
98569841254	Mariana Souza	975468000	MG
00414675821	Renato Arantes	978454587	RS

Dados semiestruturados: Dados que não apresentam uma estrutura homogênea, são irregulares ou incompletos não necessariamente de acordo com um esquema, compreensíveis por máquinas, mas não por seres humanos. Podem apresentar uma estrutura pré-definida, mas em regra geral não é rígida quanto a formato, tamanho ou domínio, ou seja, sua estrutura é heterogênea. Estes dados quando são coletados, trazem sua estrutura acoplada a eles para poderem ser lidos

conforme tal esquema. Como exemplo deste tipo de dado podemos citar os dados armazenados em arquivos XML (Figura 2).

Conforme Figura 2, a linha 1 apresenta dados relacionado à estrutura, além disso, as *tags* do arquivo também representam sua estrutura, e os valores podem ser no mesmo formato ou não. Por exemplo, compare os valores da tag CPF nas linhas 3, 9 e 15. Todos os valores se referem ao CPF de cliente, mas a o valor da linha 15 não possui caracteres especiais e os outros dois possuem. Além disso, se observar o terceiro cliente do arquivo, ele não possui UF de nascimento informada. Tais diferenças podem levar a tratamento de dados de forma diferente.

Figura 2 - Exemplo de dado estruturado.

```
1  <?xml version="1.0" encoding="UTF-8"?>
2  <cliente>
3      <CPF>157452654-23</CPF>
4      <nome>Joselito Gomes</nome>
5      <telefone>9853-2615</telefone>
6      <uf_nascimento>SP</uf_nascimento>
7  </cliente>
8  <cliente>
9      <CPF>985.698.412-54</CPF>
10     <nome>Mariana Souza</nome>
11     <telefone>975468000</telefone>
12     <uf_nascimento>MG</uf_nascimento>
13 </cliente>
14 <cliente>
15     <CPF>00414675821</CPF>
16     <nome>Renato Arantes</nome>
17     <telefone>978454587</telefone>
18 </cliente>
```

Dados não estruturados: São aqueles que como regra geral não seguem um formato ou esquema especificado, ou seja, não possuem um domínio, tamanho nem uma estrutura pré-definida. Fazem parte desse tipo de dados os dados originários de textos, páginas web, as mídias sociais, celulares, imagens, vídeos, sensores, microfones e áudios. A maior parte dos dados que são coletados atualmente são dados não estruturados. Acredita-se que 95% dos dados gerados hoje são em formato não estruturado (MAYER-SCHÖNBERGER, CUKIER, 2013:47). Este tipo de

dados exige um pré-processamento para recuperar seu conteúdo, por exemplo, uma rotina de PLN (processamento de linguagem natural) para retirar o conteúdo relevante de postagens em uma rede social.

A Figura 3 apresenta um quadro sumarizando comparativamente estes três tipos de dados.

Figura 3 - Comparativo dos tipos de dados.

Fonte: Da autora.

Vivemos cercados por uma grande quantidade de dados e informação, que apoiam a nossas decisões (HEATH, BIZER, 2011). Além de reconhecer os tipos de dados, deve-se entender que estes dados podem ser originários de diversas fontes.

Fonte de dados

Uma fonte de dados é simplesmente o local de onde o dado é coletado ou adquirido. Pode ser um arquivo, um banco de dados, um portal de notícias ou até mesmo um *feed* de notícias. Ou seja, uma fonte de dados pode ser qualquer dispositivo ou estrutura que pode fornecer dados, pode estar localizada no mesmo computador ou dispositivo que o programa de coleta ou em outro computador disponível em algum lugar da rede, seja uma intranet ou internet.

Nossos dados são provenientes de uma infinidade de aplicações e dispositivos: sistemas transacionais como ERP's, CRM's, redes sociais, aplicativos de dispositivos móveis, cookies, internet das coisas (IoT), e-mails, documentos, bancos de dados, arquivos de diversos formatos, dados públicos (abertos), da web semântica e muitas outras fontes. Conforme a origem dos dados, este podem assumir formatos e estruturas diferentes. Como exemplos de dados conforme a fonte de dados podemos considerar:

- Dados provenientes de sistemas transacionais (Sistemas bancários, ERPs[1], CRMs[2]): compras de cartão de crédito, movimentação bancária, cadastros de pessoas, serviços ou produtos, registros de ligações e de reclamações nas empresas, etc.

- Dados referentes à biometria: características fisiológicas (DNA, impressões digitais, reconhecimento facial, padrão de íris, etc.) e características comportamentais (movimentos, escrita, voz).

- Dados pessoais sujeitos à proteção por legislação: dados pessoais, documentos eletrônicos, exames e registros médicos, ligações telefônicas, etc. Estes dados podem ser provenientes de sistemas transacionais.

- Dados originários da web e redes sociais: Histórico de buscas realizadas em portais de pesquisa como o Google®, dados recolhidos pelos cookies, dados de fluxo de cliques, blogs, posts, tweets, feeds de notícias, etc.

---

[1] ERP (sigla do inglês *Enterprise Resource Planning*): Sistema de informação que visam integra dados e processos de uma organização em um único sistema. Mais detalhes em: https://pt.wikipedia.org/wiki/Sistema_integrado_de_gest%C3%A3o_empresarial.

[2] CRM (sigla do inglês *Customer Relationship Management*): Sistema de informação que visão concentras as atividades relacionadas ao contato e acompanhamento da relação entre a empresa e seu cliente. Mais detalhes em: https://pt.wikipedia.org/wiki/Customer_relationship_management.

- Dados *machine-to-machine* (gerados diretamente por máquinas): dados de sensores, dispositivos de GPS, etiquetas RFID, dispositivos móveis e medidores.

- Dados abertos e dados abertos interligados: podem ser dados governamentais como gastos diversos e investimentos, dados climáticos, dados de pesquisas censitárias, informações geográficas como dados de georreferenciamento, mapas, endereços, etc.

- Repositórios e bibliotecas eletrônicas, como Portal CAPES: artigos, dados sobre patentes, livros e entrevistas.

- Dados provenientes de dispositivos IOT: dispositivos vestíveis conectados (*wearables*), carros conectados, residências inteligentes, cidades inteligentes e serviços de saúde conectados.

- Dados de sensores (IOT): como por exemplo sensores de temperatura (clima), pluviométrico, umidade, iluminação, som (ruído), distancia, pressão, presença e assim por diante, posição, ângulo, deslocamento, distância, velocidade, aceleração.

Para realizar a análise de dados, deve-se primeiro determinar quais os tipos de dados disponíveis e quais dados serão necessários para realizar as análises necessária. Em seguida, são definidas as fontes de dados que serão usadas para coletar ou adquirir dados, consequentemente, será determinado o tipo de análise a ser feito e a ferramenta a ser utilizada.

1.2. Web semântica

A ideia de criar uma Internet semântica foi trazida por um dos idealizadores da Internet, Tim Bernes-Lee. A web semântica pode ser entendida como uma extensão da web tradicional que estrutura o significado do conteúdo da web de forma clara e bem definida, permitindo aos computadores interagir entre eles trocando

informações. Sua principal motivação é ter uma web de dados, no qual tais dados que sejam significativos tanto para os humanos quanto para as máquinas.

A situação atual da web é que tem ocorrido um crescimento significativo na quantidade de conteúdo disponível. Entretanto, a quase totalidade do conteúdo está em linguagem natural, o que é compreensível apenas para seres humanos. Com a web semântica deseja-se que a informação disponível também seja compreensível para as máquinas, que terão a capacidade de processar e "entender" o que os dados significam.

Os idealizadores da web semântica concebem um mundo onde sistemas de informação e dispositivos computacionais (especializados e personalizados), por eles denominados de *agentes*, interajam através da troca de informação entre tais sistemas e os agentes utilizando para isso a infraestrutura de dados disponível na web, possibilitando assim a automatização de atividades do cotidiano dos usuários (BERNERS-LEE et al., 2001).

A proposta trazida por BERNERS-LEE et al. (2001) é que isto seja feito com a combinação do conteúdo disponível com "metadados". Dessa forma, a semântica será incorporada sem prejudicar o que já está disponível para as pessoas. Existem quatro princípios que regem a ideia de se criar a web semântica:

1º. Sempre atribua um nome a todas as coisas ou entidades publicadas na web;

2º. Atribua um endereço para os nomes na web, (URIs);

3º. Indique as conexões (relações) entre as coisas;

4º. Providencie informações úteis sobre as coisas, usando os padrões, que definem a semântica formal e explícita: i) associe classes às coisas; ii) associe tipos às relações; iii) organize as classes em uma hierarquia; e iv) defina as restrições.

Trazendo a discussão em torno da representação da informação, existem vários sistemas de organização do conhecimento (SOCs). Os SOCs são conhecidos

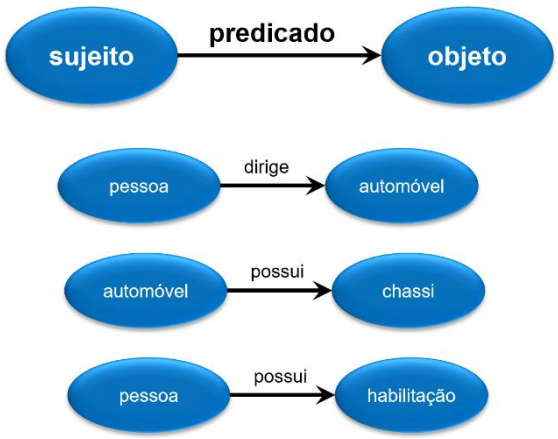
como produtos de representação (teorias, processos e instrumentos) gerados por meio de interpretações relevantes de um mundo escolhido ou domínio de informação, registrados em sistemas de informação e documentos. Os SOC podem contribuir na desambiguação de vocabulário e das estruturas necessárias para se garantir a semântica e compreensão dos termos, podendo assim ser aplicado para solucionar problemas relacionados à interoperabilidade semântica (SOUZA et al., 2010; 2012). A seguir são listados alguns SOCs:

- Vocabulários Controlados: corresponde a uma lista de termos;
- Glossários: equivalem a vocabulários controlados, mas com significado utilizando a linguagem natural, com conteúdo semelhante a um dicionário físico;
- Taxonomias: corresponde a uma estrutura hierárquica dos termos, onde tem-se uma relação generalização/especialização dos termos, para um domínio definido.
- Tesauros: corresponde a taxonomias onde tem-se também relações de equivalência (sinônimo), associação (relacionado a) e homônimos (possui a mesma grafia e significado diferente).
- Ontologias: possui as características de tesauros, incluindo funções, restrições e axiomas. Permite especificar formalmente um modelo conceitual que define um domínio, e pode ser compreendido pelos computadores.
- Folksonomias: corresponde à utilização de marcações sociais, as hashtags. Com isto, a folksonomia permite uma indexação social, através da busca pelas marcações. Ela surge como um mecanismo da inteligência coletiva, na medida que as pessoas classificam os conteúdos utilizando as hashtags.

Os elementos da representação do conhecimento são utilizados pela web semântica:

- Conhecimento declarativo: apresenta de forma explícita modelos representativos de uma parte do mundo real, com suas propriedades e relações. Por exemplo: automóvel, cor do automóvel, pessoa (condutor).
- Consulta: obter os elementos que possuem determinados valores para seus atributos. Por exemplo: selecione todos os automóveis da cor branca.
- Inferência: chegar a conclusões, inferir. Por exemplo: obter os anos dos automóveis que usam eletricidade.
- Afirmações como triplas: (sujeito) – (predicado) – (objeto) conforme ilustrado na Figura 4 a seguir.

Figura 4 - Exemplo de triplas.



Entretanto, são verificadas dificuldades ao modelar o conhecimento, pois, primeiro, o conhecimento muda ou evolui exigindo uma revisão da representação realizada; segundo, o conhecimento é dependente do contexto, do assunto, da linguagem, das habilidades de quem faz a análise, etc.; por fim, deve-se garantir a inexistência de ambiguidades na estrutura lógica do conhecimento representado.

A web semântica prevê a interoperabilidade para que ocorra o intercâmbio dos metadados. Esta interoperabilidade deve ocorrer em pelo menos 3 aspectos:

- Interoperabilidade semântica: que corresponde ao intercâmbio dos significados e relações;

- Interoperabilidade sintática: preocupa-se em compartilhar informações sobre a codificação;
- Interoperabilidade estrutura: está voltada para como o conteúdo está organizado.

1.3. Dados abertos e dados ligados

Uma das maneiras para se publicar dados na web de forma estruturada buscando estes modelos de interoperabilidade é pela adoção dos princípios chamados *Linked Data* (dados ligados) e *Open Data* (Dados Abertos).

Dados abertos (Open Data)

O conceito de dados abertos remete a ideia de conteúdo aberto, ou seja, disponível para todos. Dados abertos são dados que podem ser livremente publicados na web, seguindo alguns padrões pré-definidos, e a partir de sua publicação podem ser reutilizados e redistribuídos por qualquer pessoa ou aplicativo desde que sigam a requisição de citar a fonte deste dado além de promover o compartilhamento destes dados seguindo as regras (ISOTANI, BITTENCOURT, 2015; OKI, 2019; SLTI, MPOG, 2019).

Segundo a *Open Knowledge Internacional*, existem três características básicas que direcionam o conceito para a adoção de dados abertos (OKI, 2019):

- **Disponibilidade e Acesso:** “Os dados necessitam ser disponibilizados por completo e sob um custo razoável de reprodução, e preferencialmente na internet” (OKI, 2019).
- **Reutilização e Redistribuição:** “Os dados disponíveis precisam permitir a reutilização, redistribuição, e a combinação com outras coleções de dados” (OKI, 2019).

- **Participação Universal**: “Qualquer um pode ser capaz de usar, reutilizar e redistribuir os dados, não sendo aplicável discriminação às áreas de atuação, pessoas ou grupos” (OKI, 2019).

A lista abaixo apresenta alguns portais onde o Governo Brasileiro disponibiliza seus dados:

- Portal Brasileiro de Dados Abertos: http://www.dados.gov.br/[3].
- Portal da Transparência: http://www.portaltransparencia.gov.br/[4]
- Portal de Dados Abertos/ Transparência da Dataprev: http://portal2.dataprev.gov.br/transparencia[5].

Outras fontes de dados abertos[6] disponíveis em diversos idiomas:

- Kaggle: https://www.kaggle.com/datasets.
- Data.world Community: https://data.world/company/about-us/.
- Wikipédia: https://dumps.wikimedia.org/.
- Portal de Dados Abertos da EU: https://data.europa.eu/euodp/pt/data/.
- Portal de Dados Abertos dos EUA: https://catalog.data.gov/dataset.

---

[3] Último acesso em: 25/05/2020.

[4] Último acesso em: 25/05/2020.

[5] Último acesso em: 25/05/2020.

[6] Para outras dicas sobre onde encontrar dados abertos veja o post: VASCONCELLOS, Paulo. *Os 7 melhores sites para encontrar datasets para projetos de Data Science*. On-line: 2017. Disponível on-line em: <https://paulovasconcellos.com.br/os-7-melhores-sites-para-encontrar-datasets-para-projetos-de-data-science-8a53c3b48329>. Acesso em: 09/04/2020.

- Portal de Dados Abertos do Reino Unido: https://data.gov.uk/.

Dados ligados (Linked data)

Uma das maneiras para se publicar dados na web de forma estruturada é pela adoção dos princípios chamados *Linked Data* (dados ligados). O conceito *Linked Data* também foi iniciado por Tim Berners-Lee e partiu da ideia de interligar dados na web ao invés de documentos. Em suma, *Linked Data* é uma forma de publicar dados na web de forma estruturada, de modo que uma pessoa ou máquina possa explorar estes dados. É um conceito relacionado a web semântica determinando um conjunto de melhores práticas ou princípios para que descrevem mecanismos para tanto a publicação quanto a vinculação de dados estruturados na web (BERNERS-LEE, 2006; BIZER et al., 2009).

De acordo com BERNERS-LEE (2006), HEATH e BIZER (2011), tais princípios são os seguintes:

i) Determinar URIs (*Uniform Resource Identifier*) como nomes para as coisas que se deseja representar;

ii) Definir URIs HTTP de modo que as pessoas procurarem por esses nomes em URI;

iii) Quando alguém procurar por um URI, prover informações úteis, utilizando os padrões (RDF*, SPARQL);

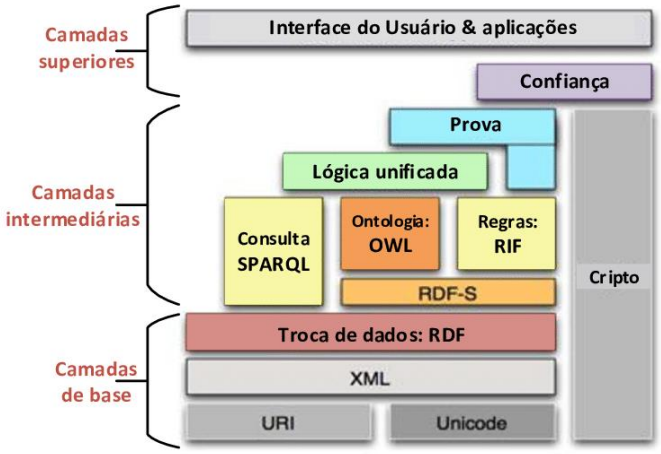
iv) Adicionar links para outras URIs visando encontrar mais coisas.

O *linked data* ainda propõe o uso de *Resource Description Framework* (RDF) como mecanismo padrão para interligar os dados. Onde seus URIs identificam unicamente qualquer tipo de objeto ou conceito. Para detalhe sobre o RDF consulte as seguintes referências: (W3C, 2004; 2014a; 2014b).

A web semântica tem sido desenvolvida utilizando-se das tecnologias existentes, em uma organização em camadas sobrepostas conforme apresentado inicialmente, este modelo já evoluiu, mas ainda representa um importante conceito

para compreensão do modelo. Cada uma das camadas pode ser desenvolvida separadamente, o que torna mais simples que desenvolver todas as camadas de uma só vez.

Figura 5 - Camadas da Web semântica.



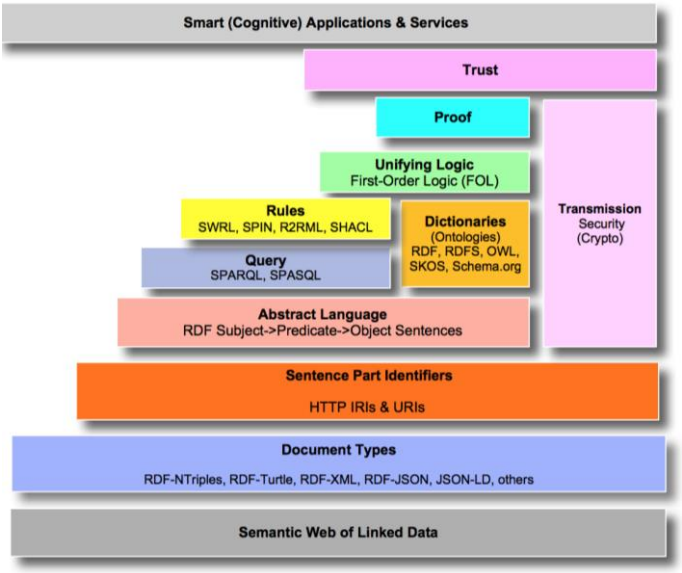
Fonte: Traduzido de BERNERS-LEE (2000)[7].

Em 2017, no artigo *Semantic Web Layer Cake Tweak*[8], Kingsley Uyi Idehen mostra um modelo atualizado que inclui a recente evolução das tecnologias da web semântica (Figura 6), entretanto o conceito das funções das camadas originalmente concebidos ainda persiste.

[7] Disponível em: <https://www.w3.org/2000/Talks/1206-xml2k-tbl/slide10-0.html>.

[8] Disponível em: <https://medium.com/openlink-software-blog/semantic-web-layer-cake-tweak-explained-6ba5c6ac3fab>. Acesso em: 25/05/2020.

Figura 6 - Modelo atualizado da web semântica.



A ideia de interligação da proposta pela web semântica e realizada pelo *linked data* permite aos usuários da web navegar entre diferentes fontes. Além disso, as ferramentas de busca ficam aptas a indexar a web e fornecer recursos de pesquisa mais sofisticados sobre o conteúdo rastreado (HEATH; BIZER, 2011).

Dados abertos ligados (Linked open data)

O conceito de dados abertos interligados ou *Linked Open Data* remete a ideia de conteúdo aberto ou disponível para todos, mas com interconexões entre os dados, ou seja, são dados interligados que se encontram disponíveis livremente na web, ou seja, dados abertos.

O *Linked Open Data* (LOD), traduzido como dados abertos vinculados por ZAIDAN e BAX (2013), é um projeto aberto comunitário mundial, de responsabilidade do W3C, que iniciou em 2007 e que visa à publicação de vários conjuntos de dados (*datasets*) de forma que as ligações sejam possíveis entre eles.

Diversas iniciativas surgiram para publicar os dados usando o padrão LOD. Estas iniciativas deram origem a nuvem de dados abertos ligados (*Linked Open Data Cloud*) apresentando todos os conjuntos de dados que foram publicados no formato

de dados vinculados e as ligações entre eles. Atualmente[9], os repositórios ou nuvem LOD contém 1.255 conjuntos de dados com 16.174 ligações. Você pode reutilizar o diagrama sob a Licença de Atribuição Creative Commons (*Creative Commons Attribution License*).

Além do diagrama de nuvem principal (

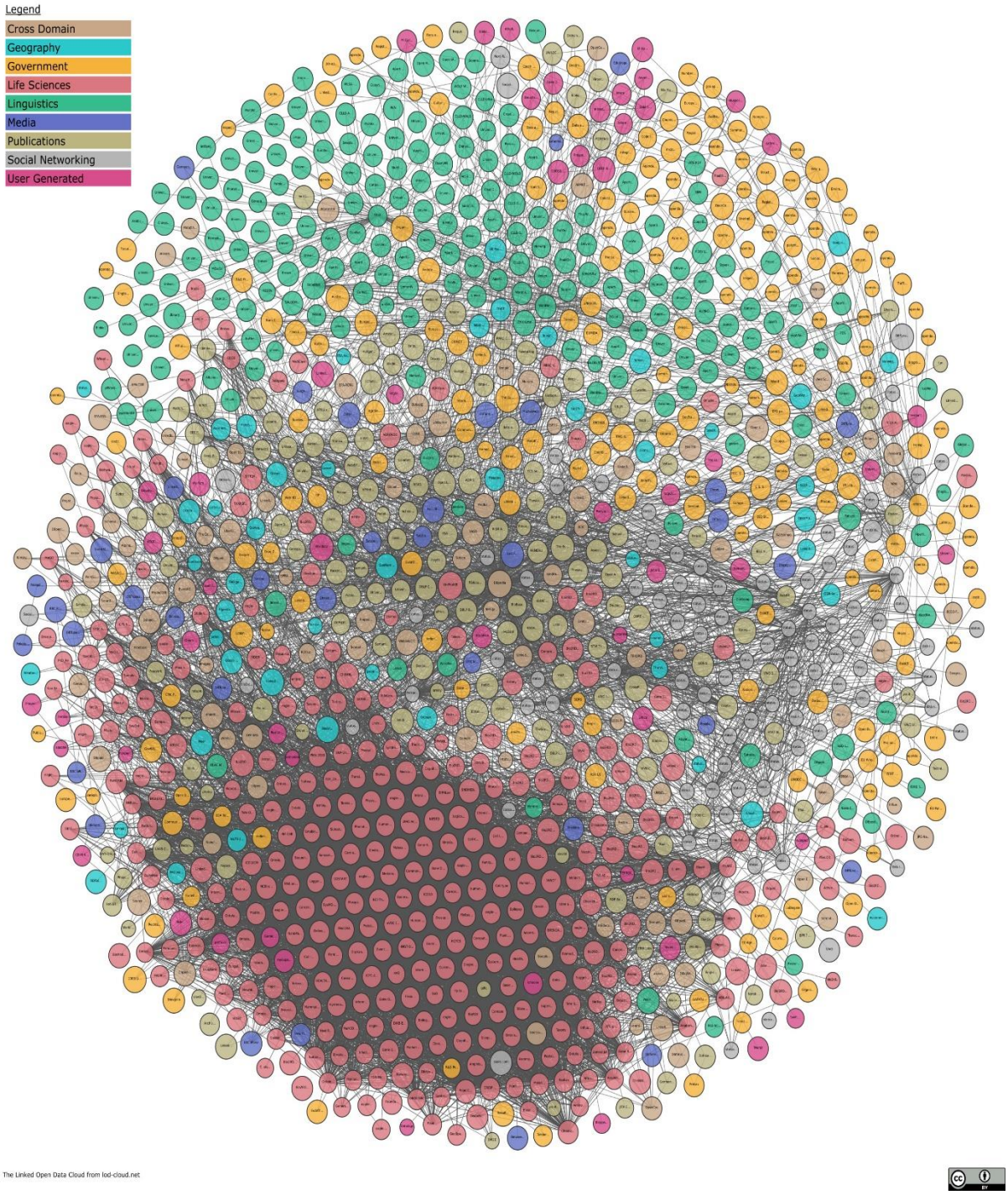
Figura 7), existem diagramas para tratar subclasses por domínio de conhecimento. Cada um dos domínios tratados corresponde à legenda da

[9] Maio/2020.

Figura 7, estes domínios são: Múltiplos domínios (domínio cruzado), Geografia, Governo, Ciências da Vida, Linguística, Meios de comunicação, Publicações, Rede social e dados gerado pelo usuário. A nuvem e as sub-nuvens se encontram disponíveis em: https://lod-cloud.net/[10].

[10] Último acesso em 25/05/2020.

Figura 7 - Linked Open Data Cloud.



Fonte: https://lod-cloud.net.

Para publicar seus dados, BAUER e KALTENBÖCK (2011) sugerem os seguintes passos:

- Analisar seus dados, selecionado aqueles que são úteis para serem publicados.
- Limpar e formatar os dados, considerando que os dados podem vir de várias fontes, alguns podem vir com informação a mais (desnecessária) para a publicação.
- Modelar seus dados para convertê-los facilmente para linguagem RDF e criar URIs para cada objeto.
- Escolher ou criar sob qual licença ficarão os dados, pode ser a *Creative Commons Attribution License.*
- Escolher, converter ou criar um vocabulário RDF apropriado.
- Ligar seus dados a outros dados antes de publicá-los. As ligações podem ser em qualquer outro dataset já criado e publicado.
- Publicar e promover seu conjunto de dados.

Além dos portais de dados abertos já citados anteriormente, a seguir são listados alguns outros datasets:

- Wikipédia, cujo dataset é o DBPedia[11] em inglês https://wiki.dbpedia.org/ e em português: http://pt.dbpedia.org/.
- Listas com diversos datasets:
  - https://datahub.io/collections/linked-open-data.

[11] Último acesso em: 25/05/2020.

– https://www.w3.org/wiki/TaskForces/CommunityProjects/LinkingOpenData/DataSets.

1.4. Fundamentos em Ontologias

Ontologia é um termo polissêmico e objeto de pesquisa em diversas área como: Filosofia, Ciência da Computação e Ciência da Informação. Ela pode ser entendida como disciplina filosófica ou como artefato representacional usada para representar compreensão acerca de vários domínios de conhecimento.

A palavra ontologia é derivada do grego:

- Onto - exprime a noção do ser, criatura.
- Logia - algo dito ou a maneira de dizer.

Como disciplina da Filosofia, a ontologia estuda a natureza da existência das coisas. Nas áreas de Inteligência Artificial e Web, o termo **ontologia** corresponde aos artefatos que descrevem domínios, como Medicina, Direito, etc., através da formalização das relações entre termos e conceitos.

Segundo Tim Berners-Lee, “Uma ontologia é um documento ou arquivo que define formalmente os relacionamentos entre termos”.

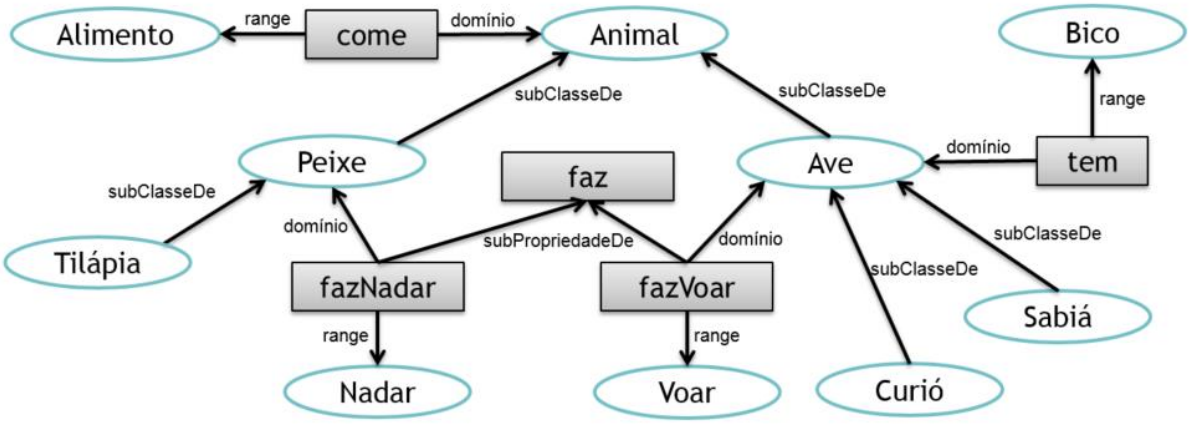
A especificação de uma ontologia inclui as descrições de:

- Conceitos e propriedades em um domínio.
- Relacionamentos entre conceitos.
- Restrições em como os relacionamentos podem ser usados.
- Indivíduos como membros de conceitos.

Uma ontologia pode ser muito complexa, com milhares de conceitos, ou muito simples, descrevendo apenas um ou dois conceitos. Um exemplo de ontologia é mostrado na Figura 8. Nesta ontologia é descrita uma pequena fração do domínio do

reino animal. Na Figura 8 é possível identificar as classes representadas por elipses e as propriedades representadas por triângulos.

Figura 8 - Exemplo de ontologia descrevendo um pequena fração do reino animal.



As propriedades em ontologias relacionam classes indicando qual é o domínio à qual pertence e qual é o seu alcance (*range*). Por exemplo, a propriedade “tem” do domínio “Ave” possui o alcance “Bico”.

As propriedades também podem ter características similares e, dessa forma, podemos identificar superpropriedades, que abrangem estas características comuns que são herdadas pelas subpropriedades. Na Figura 8 temos a propriedade “faz”, com duas subpropriedades “fazNadar” e “fazVoar”, com domínios e alcances diferentes.

Importante ressaltar que os nomes das classes devem iniciar com letras maiúsculas e os nomes das propriedades devem iniciar com minúsculas.

Avançando no conceito de ontologia, é importante ressaltar que uma ontologia define:

- Um vocabulário comum;
- Um entendimento compartilhado.

Os benefícios da ontologia incluem:

- Entendimento comum compartilhado entre pessoas e entre computadores;
- Reúso do conhecimento sobre o domínio representado;
- Interoperabilidade das aplicações que a utilizam.

No contexto da web semântica, as ontologias podem incrementar as funcionalidades da web e melhorar a qualidade das buscas. Elas são muito úteis quando existe ambiguidade nos termos utilizados em diferentes *datasets*.

Como exemplo, considere um revendedor de livros que deseja integrar dados de diferentes editoras. Os dados podem ser importados em um modelo RDF usando conversores. Entretanto, um *dataset* utiliza o termo “autor” enquanto o outro utiliza o termo “criador”. Para que se tenha a integração completa, deve-se acrescentar informação no RDF descrevendo o fato que “autor” é o mesmo que “criador”.

As ontologias podem ser classificadas em:

- Ontologias de alto nível: descrevem conceitos amplos independentes de um domínio particular. Ex.: relacionadas a espaço, tempo, eventos, etc.;
- Ontologias de referência: descrevem conceitos relacionados a atividade ou tarefas genéricas, independentes de domínio. Ex.: diagnóstico;
- Ontologias de domínio: descrevem conceitos relacionados a domínios específicos, como direito, computação, etc. É a categoria mais comum;
- Ontologias de aplicação: descrevem conceitos dependentes de um domínio e tarefa específicos.

As ontologias descrevem entidades sobre a perspectiva dos universais e particulares:

- **Particulares ou indivíduos**: ocorrências únicas de algo existente na realidade.

  - Exemplo: Cada um de nós é uma única ocorrência ou indivíduo de um "homo sapiens".

- **Universais ou tipos:** entidades reais que generalizam os particulares existentes no mundo. Existe apenas se existir pelo menos um particular desse universal.

  - Exemplo: "homo sapiens“ é uma entidade geral ou universal referente aos particulares que cada um de nós é.

Elementos de uma ontologia

- ***Entidades***: É algo que você deseja representar em um domínio particular. Qualquer coisa que exista, existiu ou irá existir. Ex.: eventos, processos, objetos inanimados ou vivos, etc.

- ***Classes***: Representam as entidades do domínio. O organizam as entidades de um domínio em uma taxonomia. Universais.

- ***Atributos de classe:*** Propriedades relevantes da classe que ajudam a descrevê-la.

- ***Instância:*** Representam uma unidade de objetos específicos de uma entidade, ou seja, indivíduos de um determinado universal.

- ***Atributos da instância:*** Essas são propriedades relevantes que descrevem as instâncias de uma entidade.

- ***Relacionamento:*** Descreve o tipo de interação entre duas classes, duas instâncias ou uma classe e uma instância.

- ***Cardinalidade:*** Uma medida do número de ocorrências de uma entidade associada a um número de ocorrências em outra.

- ***Axioma:*** Uma declaração ou proposição representada em um padrão lógico que é considerado verdadeiro. Restringem a interpretação e o uso das classes envolvidas na ontologia.

- Exemplo:

$$e\ instanceOf\ \mathrm{E} \quad \Rightarrow \quad \forall e \forall \mathrm{E} \left(inst(e, \mathrm{E}) \rightarrow p(e) \wedge u(\mathrm{E}) \right)$$

Onde:

– e , E são variáveis para instância e classe;

– inst, p , u são funções para instância, particular e universal;

– o símbolo ∧ significa conjunção;

– o símbolo ∀ é o quantificador universal;

– o símbolo → é uma implicação.

Web Ontology Language

A linguagem OWL (*Web Ontology Language*) é uma linguagem para representação de ontologias, projetada para o uso por aplicações que necessitam processar o conteúdo da informação, ao invés de apenas apresentar a informação para humanos.

A OWL permite uma capacidade de interpretação do conteúdo Web pelas máquinas maior do que a suportada pelo XML, RDF e RDF Schema (RDF-S), através do fornecimento de vocabulário adicional, juntamente com uma semântica formal. As recomendações para OWL estão em www.w3.org/TR/2004/REC-owl-features-20040210.

A OWL possui sub-linguagens, com algumas características que as diferenciam:

- OWL Lite: foi projetado para ser fácil de implementar, menos complexo e, dessa forma, auxiliar os usuários a iniciarem na OWL;

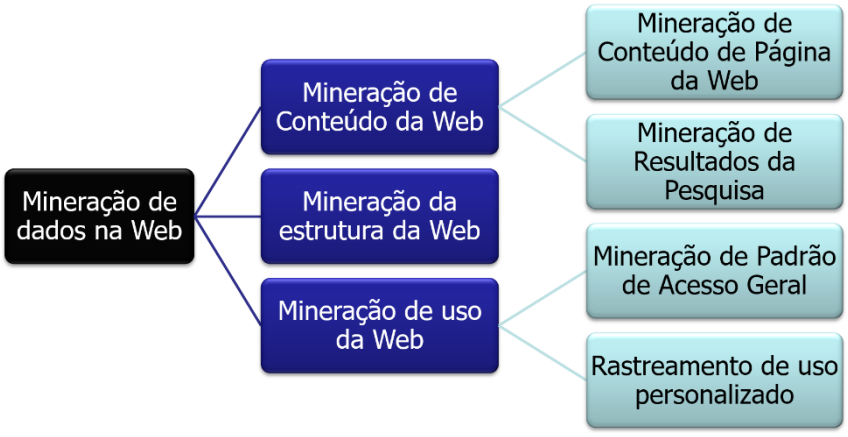
- OWL DL: permite a máxima expressividade, mas com a garantia que pode ser processada em computador e em tempo finito, o que exige algumas restrições sintáticas;
- OWL Full: permite a máxima expressividade e liberdade sintática, o que não garante a utilização por software.

1.5. Web Mining

Web Mining (Mineração na Web) corresponde à aplicação de técnicas de Data Mining (Mineração de Dados) à web. Ou seja, Web Mining é o processo de extração de conhecimento, a partir dos dados da web. Conforme Figura 9, os tipos de dados da web envolvidos, as respectivas categorias de Web Mining, são:

- O conteúdo da Web -> Web Content Mining;
- A estrutura da Web -> Web Structure Mining;
- A utilização da Web -> Web Usage Mining.

Figura 9 - Categorias de web mining.



Web Content Mining

A Web Content Mining extrai informação do conteúdo dos recursos web. Os recursos web podem ser texto, imagem, áudio, vídeo, etc.

Para recursos do tipo texto são utilizados algoritmos de Text Mining como:

- Classificação: a partir de um conjunto de recursos com classes previamente definidas é gerado um modelo que permite prever as classes de novos recursos ainda não classificados;
- Clusterização: agrupa os recursos conforme as similaridades entre si. Os recursos com maior similaridade ficarão nos mesmos grupos.

A classificação é considerada uma técnica supervisionada, pois utiliza uma base de recursos previamente classificados para treinar o modelo. Em seguida é realizado um teste com um grupo de recursos previamente classificados e que não foram utilizados no treinamento. Para a **classificação** há algumas abordagens, conforme características dos dados e modelo a ser obtido. São elas:

- Regressão: é utilizado quando a classe é um valor numérico que pode ser calculado como uma operação sobre os demais atributos numéricos do recurso a ser classificado;
- Regras de decisão: gera um modelo contendo regras de decisão que levam à classificação de novos recursos;
- Árvores de decisão: gera um modelo em formato de árvore de decisão para a classificação de novos itens;
- Redes neurais: gera um modelo que é treinado para fornecer a classificação. Redes neurais são considerados algoritmos que trabalham como caixa preta, pois fornecem bons resultados, mas não mostram o que leva às classes resultantes do seu processamento. Quanto mais recursos fornecidos no treinamento, melhores são os resultados da classificação.

A **clusterização** é utilizada para descobrir uma organização das informações (recursos web) em grupos, com base em alguma medida de similaridade. Ela corresponde a um processo não supervisionado, pois não necessita de treinamento ou pré-definição de classes dos dados.

O número de clusters pode ser definido *a priori*, na chamada k-clusterização. Na clusterização automática a obtenção do número de clusters da melhor configuração faz parte da solução.

Como medida de similaridade na clusterização normalmente são utilizadas as similaridades Euclidiana e a de Manhattan.

Web Structure Mining

A Web Structure Mining tem como objetivo principal extrair relacionamentos, previamente desconhecidos, entre recursos web.

A Estrutura Web é representada por um grafo com os vértices sendo os recursos web (como páginas, imagens, vídeos, etc.) e as arestas sendo os links que conectam os recursos.

A representação em grafo dos recursos e das conexões entre eles é utilizada para vários tipos de análise, normalmente usando técnicas matemáticas relacionadas à Teoria dos Grafos.

A principal técnica de mineração de dados utilizada em grafos é a clusterização, na qual são agrupados os recursos (vértices) de forma a maximizar a quantidade de arestas que ligam os recursos dentro de um mesmo cluster e minimizar a quantidade de arestas entre recursos de diferentes clusters.

Web Usage Mining

A Web Usage Mining utiliza técnicas de mineração de dados para encontrar analisar ou descobrir padrões de navegação do usuário nos sites. O objetivo é melhorar a experiência do usuário nas aplicações web.

Para a mineração na utilização da web, podem ser analisados:

- Dados do servidor web: os logs de acesso do usuário são coletados no servidor web e normalmente consistem do endereço IP do usuário, página acessada e tempo de acesso;

- Dados do servidor de aplicação: servidores de aplicação comerciais permitem rastrear vários tipos de eventos de negócio e armazená-los em logs;
- Dados da aplicação: outros eventos a serem rastreados podem ser definidos para a utilização da aplicação pelo usuário.

Algumas técnicas utilizadas na Web Usage Mining são:

- **Filtragem**: permite responder perguntas como “qual a quantidade de usuários que visitaram o site usando um determinado browser?”;
- **Análise estatística**: fornece dados estatísticos relacionados a page views, tempo de visualização, etc.;
- **Padrões sequenciais**: identifica padrões em acessos ao longo do tempo. Isto permite identificar usuários que estejam iniciando a execução do padrão e permite agir para evitar problemas ou aproveitar oportunidades;
- **Clusterização**: pode, por exemplo, agrupar usuários conforme a forma de acesso às páginas ou preferências no acesso;
- **Classificação**: a partir da clusterização de usuários, por exemplo, pode-se definir uma classe para cada cluster e aplicar uma posterior classificação para obter os modelos que permitam classificar novos usuários;
- **Regras de associação**: a obtenção de regras de associação identifica eventos que acontecem juntos para uma quantidade relevante de usuários. A relevância é obtida através de métricas.

Exemplos de regras de associação que podem ser obtidas a partir de Web Usage Mining são:

– Se (visitou página A) então (visitou página B);

– Se (visitou página X) e (visitou página Y) então (visitou página Z);

– Se (visitou página X) então (visitou página Y) e (visitou página Z);

– Se (acessou do Brazil) então (visitou página Y).

Vale ressaltar que na Web Usage Mining normalmente é necessário um grande esforço de **pré-processamento**, devido às características dos dados coletados. No pré-processamento são realizadas as tarefas de limpeza e integração dos dados.

1.6. Text Mining

A Mineração de Texto (Text Mining), consiste na aplicação de técnicas de mineração de dados para obtenção de informações importantes em um texto. É um processo que utiliza algoritmos capazes de analisar coleções de documentos texto escritos em linguagem natural com o objetivo de extrair conhecimento e identificar padrões. Dentre as técnicas utilizadas, destaca-se o processamento de linguagem natural.

Processamento de Linguagem Natural

O Processamento de linguagem natural (*Natural Language Processing* – NLP) é uma subárea da Inteligência Artificial que já está muito bem estabelecida. Ela envolve a análise computacional da linguagem natural e a identificação do seu significado, através de correlações estatísticas.

Através desta técnica os algoritmos são treinados a entenderem a linguagem natural. No contexto das mídias sociais, o treinamento é conduzido através de um processo em que mensagens ou palavras, são apresentadas a um analista que as classifica da melhor forma possível, dentro de um número de categorias previamente definidas. O algoritmo então avalia os atributos linguísticos da mensagem, que podem ser palavras, frases, a ordem em que aparecem, ou mesmo os *emoticons*, e passa a associar estes atributos à categoria específica. Assim, após a realização de muitas classificações pelo analista, o algoritmo “aprende” a classificar, de forma automática, novas mensagens.

Uma aplicação do NLP é a análise de sentimento, na qual a classificação de uma mensagem indica o sentimento do seu criador. Na análise de sentimento, as categorias normalmente utilizadas para o sentimento são: positivo, neutro e negativo.

Os dados obtidos pelo NLP podem ser armazenados em uma base de dados para outras análises posteriores, como a aplicação da mineração de dados.

Mídias Sociais

As mídias sociais são plataformas que permitem a criação e troca de conteúdo gerado pelos usuários. O conceito de mídias social on-line se baseia na ideia de serviços disponíveis na web que possibilita a um indivíduo (i) construir perfis públicos ou semi-públicos dentro de um sistema; (ii) estabelecer listas e/ou grupos de usuários conectados por alguma afinidade; e (iii) percorrer seus grupos de conexões assim como os grupos de outros usuários do serviço (BENEVENUTO et al., 2011).

Estas redes ou mídias sociais oferecem diversas oportunidades para análise de dados, pois configuram um ambiente onde seus usuários criam e compartilham conteúdos e interagem entre eles, criando uma rede de afinidades entre indivíduos e indivíduos e organizações/produtos/serviços. Os dados oriundos destas redes ganharam relevância nas análises de *Big Data*, pois constituem um ambiente rico para extração de informações que podem ser convertidos em conhecimento por meio da aplicação de técnicas de mineração de dados. As plataformas das mídias sociais são muito variadas. Em todas elas dados podem ser obtidos e analisados. Alguns exemplos de plataformas são:

- E-mails;
- Chats;
- Redes sociais;
- Blogs;
- Microblogs;
- Wikis;
- Sites de perguntas e respostas;
- Sites de bookmarks/listas compartilhados;
- Social News;
- Compartilhamento de mídias;
- Opiniões e avaliações.

Os dados existentes nas mídias sociais são diversos, variando conforme o propósito da rede. Entretanto, nem sempre todos os dados estão disponíveis para uso da comunidade em geral, principalmente para a comunidade de desenvolvimento de aplicações de análise de dados. Cada rede social oferece seu próprio mecanismo de coleta de dados, e os dados que estão disponíveis dependem de sua política de privacidade de dados, e da adequação às legislações como a LGPD[12] e a GDRP[13].

A seguir, destaca-se alguns dados que as redes sociais coletam de seus usuários com base no estudo de BENEVENUTO et al. (2011).

- ***Perfis dos usuários:*** Os perfis são usados para identificar o indivíduo na rede e para identificar indivíduos com interesses comuns visando articular novas conexões. Os dados dos perfis são compostos por dados demográficos do indivíduo – por exemplo: idade, sexo, localização – e conteúdo de seu interesse – como por exemplo: passatempos, bandas, filmes e livros favoritos, viagens – e às vezes por fotos. Permitem dados em formato diversos, como texto, imagens, áudio e vídeo. Os perfis geralmente podem ser acessados por qualquer pessoa com uma conta na rede social ou podem ser privados, conforme parametrizado pelo usuário seguindo as políticas de privacidades.

- ***Comentários:*** Em geral, as redes sociais on-line permitem que usuários comentem o conteúdo compartilhado por outros. Os usuários podem ainda adicionar comentários nos perfis de outros usuários de sua rede de contatos. Comentários também podem ser em formato diversos, como texto, imagens, áudio e vídeo e configuram como um elemento fundamental da comunicação nas redes sociais.

---

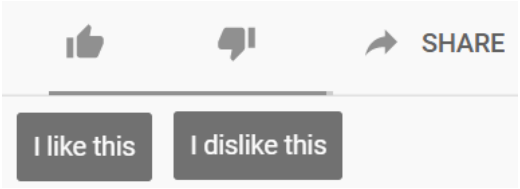
[12] Lei Geral de Proteção de Dados Pessoais, Lei Brasileira nº 13.709/2018, texto completo disponível em: <http://www.planalto.gov.br/ccivil_03/_ato2015-2018/2018/lei/L13709compilado.htm>. Acesso em: 25/05/2020.

[13] General Data Protection Regulation (Regulamento Geral sobre a Proteção de Dados, aplicável a todos os indivíduos na União Europeia e Espaço Económico Europeu que foi criado em 2018). Detalhes disponível em: <https://gdpr-info.eu/>. Acesso em: 25/05/2020.

- ***Avaliações ou reações:*** As redes sociais ainda permitem que seus usuários expressem suas reações em relação ao conteúdo compartilhado por um determinado usuário. Tais avaliações ou reações podem ser muito específicas conforme a plataforma da rede social. Por exemplo, a rede Facebook® permite aos usuários reagirem a uma postagem por meio de menu de opções de *emoticons* conforme Figura 10. Já no caso, do YouTube®, Figura 11, os vídeos podem ser avaliados como "Gostei" (*I like this*) e "Não gostei" (*I dislike this*) além de permitir o compartilhamento (*Share*).

Figura 10 - Menu de reações disponível no Facebook®.

Figura 11 - Menu de reações disponível no YouTube®.



- ***Listas de Favoritos:*** São listas de conteúdos favoritos que os usuários selecionam. Estas listas auxiliam aos usuários no gerenciamento do seu próprio conteúdo e são úteis para receber as recomendações oriundos da própria rede.

- ***Listas de Mais Populares (Top Lists):*** As *Top Lists* são criadas pelo próprio serviço da rede social a partir dos conteúdos ou usuários mais populares. Tais listas são formadas a partir das estatísticas baseadas nas avaliações dos usuários ao conteúdo.

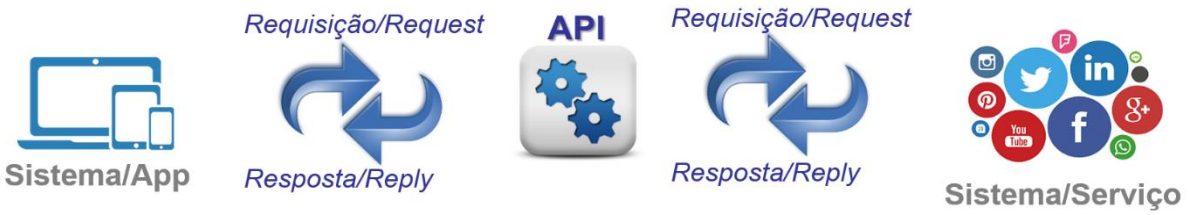
- ***Metadados:*** Algumas redes sociais, tais como o YouTube® e o Instagram®, permitem aos usuários associarem metadados, como título, descrição, georreferenciamento e hashtags, ao conteúdo compartilhado.

Existem várias abordagens e técnicas específicas para análise de redes sociais. Tais abordagens variam em relação ao tipo de conteúdo, ao tipo de dados (estruturado, não estruturado, semiestruturado) e ao objetivo de análise. Existem algumas categorias que delimitam o tipo de intercâmbio social que ocorre entre atores numa rede, por exemplo, expressão de afeto (positivo ou negativo), troca de informação, grau de influência ou recursos materiais.

1.7. O que são APIs?

O acrônimo API corresponde às palavras em inglês "*Application Programming Interface*", em português "Interface de Programação de Aplicações". Uma API é um middleware ou software intermediário que permite que dois aplicativos se comuniquem, conforme ilustrado na Figura 12 a seguir. Quando você usa um aplicativo de mídia social como o Facebook ou Twitter, ou envia uma mensagem instantânea ou verifica o clima em app no seu telefone, você está usando uma API.

Figura 12 - Esquema de funcionamento de uma API.



Fonte: Da autora.

Uma API é composta por um conjunto de rotinas (programas) que são responsáveis por realizar várias operações previamente conhecidas e divulgadas pelo fornecedor da própria API. Por meio das APIs é possível utilizar suas funcionalidades seguindo os protocolos previamente definidos.

Assim, o desenvolvedor de aplicativos ou sistemas não precisa necessariamente saber como a funcionalidade é realizada, eles só precisam saber que está disponível para uso em seu aplicativo. As APIs permitem que os

desenvolvedores economizem tempo aproveitando a implementação de uma plataforma para realizar o trabalho. Isso ajuda a reduzir a quantidade de código que os desenvolvedores precisam criar e também cria mais consistência entre aplicativos para a mesma plataforma.

A forma mais indicada para a obtenção de dados a partir das mídias sociais ou buscar conteúdo na web é por meio da utilização das APIs e das funcionalidades e serviços por elas compartilhada. Através destes serviços, é possível publicar conteúdo, coletar conteúdo e dados dos usuários, além de outras ações específicas de cada mídia social. Com isto, pode-se tanto automatizar o monitoramento das mídias sociais quanto alimentar as bases de dados para posterior análise.

APIs para mídias sociais e portais de conteúdo

Cada plataforma de mídia social possui APIs com características próprias, e que podem variar ao longo do tempo. Alguns serviços podem também estar disponíveis apenas para contas com pagamento de assinatura. Assim, ao criar uma aplicação que utilize uma API é necessário consultar as informações mais atualizadas e acompanhar a evolução da API, para que não ocorra descontinuidade nas suas funcionalidades. Além disso, devem ser respeitadas as políticas de utilização dos dados obtidos, e não se esquecer as leis de proteção de dados.

Em suma, APIs são contratos padrão que definem como os desenvolvedores se comunicam com um serviço e o tipo de saída que esses desenvolvedores esperam receber de volta. A seguir, são apresentadas os URL[14] com informações das APIs das principais plataformas de mídias sociais:

- Twitter: https://developer.twitter.com/en;
- Linkedin: https://www.linkedin.com/developers/apps;
- Facebook: https://developers.facebook.com/docs/graph-api?locale=pt_BR;

[14] Todas as URLs citadas foram acessadas em 26/05/2020. Podem sofre alterações.

- Instagram: https://www.instagram.com/developer/register/;
- Youtube: https://developers.google.com/youtube/v3/;
- Google Data API: https://developers.google.com/gdata/docs/directory.

Observação: Lembre-se que é necessário ter uma conta na mídia social para criar aplicativos e obter as chaves de acesso para utilização da API.

Além das APIs de mídias sociais, ainda existem algumas para portais de conteúdo e portais de dados abertos. A seguir são citados algumas destas APIs.

- API Jornal New York Times: https://developer.nytimes.com/.
- API do Portal Pubmed: https://www.ncbi.nlm.nih.gov/home/develop/api/.
- APIs do Portal da Transparência Brasileiro:
  - http://www.portaltransparencia.gov.br/api-de-dados.
  - http://www.transparencia.gov.br/swagger-ui.html.

O acesso aos dados através de uma API está sujeito a restrições e a alterações. Nos últimos anos, as plataformas de mídias sociais restringiram bastante o acesso a seus dados por uma aplicação de terceiros. Várias mídias inibiram o acesso a dados pessoais, por exemplo. Esta é uma tendência, devido a questões de segurança, questões legais trazidas pela LGPD e pela GDRP, e também para manter a confiança de seus usuários.

Em geral, o padrão OAuth é o padrão utilizado pelas APIs para permitir o acesso de um site ou aplicativo de terceiro aos seus serviços. É possível usar suas contas do Facebook®, Google® ou Twitter® para entrar em um novo site sem criar uma nova conta de usuário apenas para esse site, por exemplo, quando você utiliza o Medium (https://medium.com/) seu acesso por ser utilizando uma conta que você possui em outro aplicativo.

Capítulo 2. Coleta de Dados em Bancos de Dados Relacionais

Um banco de dados é “uma coleção de dados inter-relacionados, representando informações sobre um domínio específico” (SILBERSCHATZ et al., 2012). Para suporta os bancos de dados, foram criados os Sistema Gerenciadores de Banco de Dados (SGBD ou em inglês Data Base Management System - DBMS). SGBDs são sistemas ou softwares utilizados para gerir os bancos de dados, permitindo: i) criar, modificar e eliminar bases de dados; ii) realizar as operações básicas com os dados (inserir, alterar, excluir e consultar); iii) garantir a segurança de acesso aos dados; iv) garantir a integridade de dados, controle de concorrência e possibilidades de recuperação e tolerância a falhas.

2.1. Sistemas Gerenciadores de Bancos de Dados Relacionais

Os bancos de dados relacionais são os mecanismos de persistência de dados mais adotados por empresas nas últimas décadas. É um SGBD que implementa o modelo relacional de dados, que é baseado na matemática relacional de conjuntos. (ELMASRI, NAVATHE, 2005; SILBERSCHATZ et al., 2012)

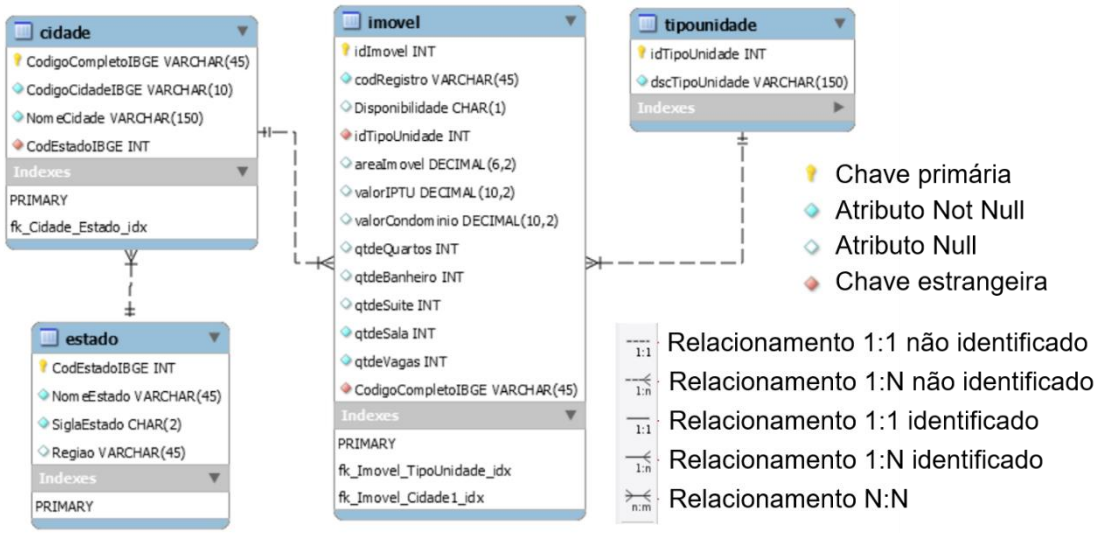
O modelo relacional é uma teoria matemática criada por Edgar Frank Codd em 1970 para descrever como as bases de dados devem funcionar. Fundamentado na teoria de conjuntos e nas possíveis relações entre os conjuntos, permitindo operações de junção, união, retorno seletivo de dados e diversas outras operações matemáticas. Até recentemente, este modelo foi considerado o mais flexível e adequado ao solucionar os vários problemas que se colocam no nível da concepção e implementação da base de dados para tratamento de dados estruturados (ELMASRI, NAVATHE, 2005; HEUSER, 2008).

Na sua estrutura fundamental, o banco de dados relacional possui a *relação* (entidade no modelo conceitual e tabela no modelo físico) e as associações ou *relacionamentos* entre as relações. Uma relação é constituída por um ou mais *atributos* (campos ou colunas no modelo físico) que traduzem o tipo de dados a

armazenar. Cada instância ou ocorrência do esquema de dados é chamada de tupla (registro ou linha no modelo físico). Um atributo possui ainda um domínio vinculado a ele, ou seja, um conjunto de valores atômicos que o atributo pode assumir.

Estes elementos são traduzidos para o modelo de dados relacional físico como tabelas, relacionamentos, atributos e chaves conforme Figura 13.

Figura 13 - Elementos de um modelo relacional.



Os bancos de dados relacionais são adequados para solucionar problemas que se colocam no nível da concepção e normalmente o uso mais comum deste tipo de banco é para implementar funcionalidades do tipo CRUD (do inglês Create, Read, Update e Delete), ou seja, criar ou inserir, ler ou selecionar, alterar e excluir um dado. As funcionalidades dos sistemas de informação fazem interface com os bancos de dados utilizando estas 4 operações básicas (CRUD) e geralmente por múltiplas aplicações em paralelo. Além disso, uma determinada funcionalidade pode envolver múltiplas operações, assim, temos o conceito de transação em um SGBD. Uma transação é uma sequência de operações executadas como uma única unidade lógica de trabalho.

Propriedades ACID

Para garantir que as transações sejam realizadas de forma que o banco de dados continue integro, os SGBDs relacionais implementam as propriedades conhecidas como ACID (Atomicidade, Consistência, Isolamento e Durabilidade):

- ***Atomicidade:*** Uma transação é composta por múltiplas operações, assim, a transação é uma unidade atómica de processamento, quando realizada, ou se faz tudo ou nada, sem meio termo. Todas as operações só serão aplicadas e persistidas se – e somente se – todo o conjunto de alterações for concluído com sucesso.

- ***Consistência:*** Os dados estarão sempre consistentes e completos, respeitando as regras de integridade, relacionamentos e restrições configuradas no banco. Tem por objetivo garantir que o banco de dados antes da transação esteja consistente e que, após a transação, o banco permaneça consistente, sem problemas de integridade.

- ***Isolamento:*** A manipulação dos dados é realizada de forma isolada, garantindo que não haja interferência externa por outra transação sendo realizada no mesmo instante. Desta forma, uma transação deve aguardar que a outra termine para poder acessar e manipular os dados.

- ***Durabilidade:*** Uma vez que o banco de dados retornou à informação de que o dado está salvo, ele não será mais perdido.

Existem vários bancos de dados relacionais no mercado, entre os mais confiáveis e robustos podemos destacar o PostgreSQL, o IBM DB2, o MySQL, o Oracle e o Microsoft SQL Server.

2.2. Linguagem Estruturada de Consultas (SQL)

A Linguagem Estruturada de Consultas mais conhecida como **linguagem SQL** (do inglês *Structured Query Language*) surgiu no início dos anos 70 com o

objetivo de fornecer uma interface mais amigável ao usuário para acesso aos bancos de dados relacionais. Foi criada para interagir com os bancos de dados relacionais, pois, assim como os próprios bancos de dados, a linguagem é criada com o conceito de teoria de conjuntos. SQL serve para criar tanto as estruturas como os dados nos bancos de dados. Esta linguagem é dividida em:

- ***Linguagem de Definição de Dados:*** DDL (Data Definition Language) é a parte da linguagem SQL utilizada para criação ou definição dos elementos ou estrutura do banco de dados, assim como a modificação e remoção das estruturas. Ou seja, permite criar (CREATE) estrutura como tabelas, chaves, índices, usuários, procedimentos, etc., excluir (DROP) as estruturas, e alterar (ALTER) as estruturas.

- ***Linguagem de Manipulação de Dados:*** Também conhecida como DML (Data Manipulation Language), é o subconjunto da linguagem SQL, utilizada para realizar as operações de manipulação de dados, como inserir (INSERT), atualizar (UPDATE) e excluir ou apagar (DELETE) dados.

- ***Linguagem de consualta a Dados:*** Também conhecida como DQL (Data Query Language), é o subconjunto da linguagem SQL, utilizada para realizar as operações de consulta aos dados armazenados no banco de dado, como selecionar (SELECT).

- ***Linguagem de Controle de Dados*** ou (DCL - *Data Control Language*): É a parte da linguagem SQL que controla os aspectos destinados a autorização de acesso aos dados para manipulação de dados dentro do BD. Alguns comandos comuns são GRANT(dá privilégios para usuários), REVOKE (revoga privilégios de usuários).

Capítulo 3. Coleta de Dados em Bancos de Dados NoSQL

Aos poucos, os bancos de dados relacionais começaram a não atender aplicações específicas relacionadas ao Big Data. Requisitos como a escalabilidade armazenamento e processamento de dados sob demanda, armazenamento de dados não padronizados e o elevado grau de disponibilidade, contribuiu para o surgimento de novos paradigmas e tecnologias. Neste contexto, surgiu uma nova categoria de Banco de Dados que trabalham com o armazenamento não relacional da informação, que foi proposta com o objetivo de atender aos requisitos de gerenciamento de grandes volumes de dados, semiestruturados ou não estruturados, que necessitam de alta disponibilidade e escalabilidade.

3.1. Sistemas gerenciadores de bancos de dados NoSQL

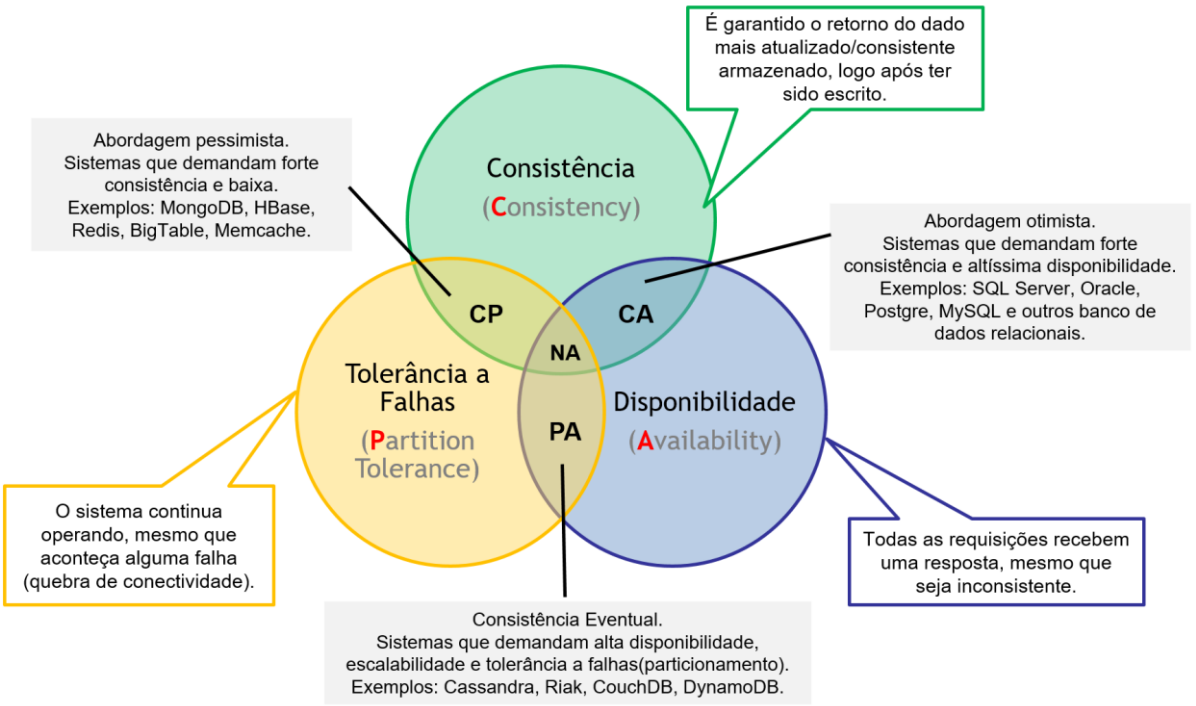
Com o passar do tempo, estes bancos de dados não relacionais acabaram sendo conhecidos pelo acrônimo ***NoSQL***, que significa *Not Only SQL* (não somente SQL, em tradução livre), em uma alusão a ideia de um movimento que prega que nem todos os cenários de dados são adequados ao uso de bancos de dados relacionais, ou mais especificamente ao uso da linguagem SQL para consulta e manipulação de dados. Desta mesma forma, o acrônimo SQL passou a se referir também em diversas esferas ao banco de dados relacional propriamente dito, de forma que passou a ser comum encontrar a designação SQL para tratar do modelo relacional e NoSQL para tratar o modelo não relacional.

Os bancos de dados NoSQL são considerados uma boa opção para aplicações fundamentadas no Big Data, uma vez que fornecem recursos eficientes para armazenamento de grandes volumes de dados estruturados e não estruturados, apresentam fácil acesso, alta escalabilidade e disponibilidade, além de baixo custo. Quando comparados com bancos de dados relacionais, as tecnologias NoSQL geralmente usam interfaces de consulta de baixo nível e não padronizadas, o que torna mais difícil a integração em aplicativos existentes que esperam uma interface SQL.

Teorema CAP

O Teorema CAP ou Teorema de Brewer, foi criado por Dr. Eric Brewer, na *Association for Computing Machinery* (ACM), para descrever o comportamento de um sistema distribuído quando acontece uma requisição de escrita de dados seguida de uma requisição de leitura (consulta). Conforme o teorema (Figura 14), dado um par de requisições, uma escrita seguida por uma leitura, é impossível que o armazenamento de dados distribuído garanta simultaneamente mais de duas das três seguintes características: Consistência, Disponibilidade e Tolerância a falhas.

Figura 14 - Esquema explicativo do teorema CAP.



Fonte: Da autora.

Em outras palavras, o teor do CAP afirma que, na presença de uma partição da rede, é preciso escolher entre consistência e disponibilidade. Observe que a consistência, conforme definida no teor de CAP, é bastante diferente da consistência garantida em transações de bases de dados ACID. Nenhum sistema distribuído está protegido contra falhas de rede, portanto, a partição geralmente deve ser tolerada. Na presença de partições, são dadas duas opções: consistência ou disponibilidade. Ao

escolher consistência em relação à disponibilidade, o sistema retornará um erro ou um tempo limite se informações específicas não puderem ser garantidamente atualizadas devido à sua partilha na rede. Ao escolher disponibilidade sobre consistência, o sistema sempre processará a consulta e tentará retornar a versão disponível mais recente da informação, mesmo que não possa garantir que ela esteja atualizada devido às partições.

Propriedades BASE

Os sistemas de gerenciamento de banco de dados NoSQL geralmente não aderem necessariamente às propriedades transacionais da atomicidade, consistência, isolamento e durabilidade (ACID). Estes sistemas promovem as propriedades conhecidas como BASE (Basically Available, Soft state, Eventual consistency), que distribui os dados em diferentes repositórios tornando-os sempre disponíveis, não se preocupa com a consistência de uma transação, delegando essa função para a aplicação, porém sempre garante a consistência dos dados em algum momento futuro à transação. Como são projetados para garantir escalabilidade e disponibilidade, muitas vezes sacrificam a consistência dos dados (STROHBACH et al., 2016). O acrônimo BASE é um acrônimo forçado para fazer uma alusão proposital à oposição do modelo ACID.

- ***Basically Available* (Basicamente Disponível):** O banco de dados não relacional é desenhado para estar sempre disponível e respondendo principalmente às operações de escrita, mesmo que nem todos os seus dados estejam disponíveis para consulta naquele momento. Esta característica garante que o grande volume de dados sendo recebido pela aplicação seja escrito, e que sempre haja uma resposta válida para uma consulta.

- ***Soft state* (em um estado flexível):** Uma informação armazenada em um banco de dados não relacional pode ter uma relevância temporal ou em relação a seu acesso. Isso faz com que uma determinada informação seja alterada ou descartada pelo próprio sistema, não garantindo que ela esteja salva da mesma forma como foi inserida pelo usuário. A localização atual de um usuário

pode ser descartada ou ter seu status alterado em função do tempo em que ela foi recebida.

- ***Eventual consistency* (Eventualmente Consistente):** Haverá um momento em que todos os dados estarão consistentes, porém haverá um momento em que parte da informação pode estar ausente ou desatualizada em relação ao todo. Isso pode ocorrer em função de um atraso na atualização dos dados ou pela queda momentânea de um dos nós da aplicação de banco de dados.

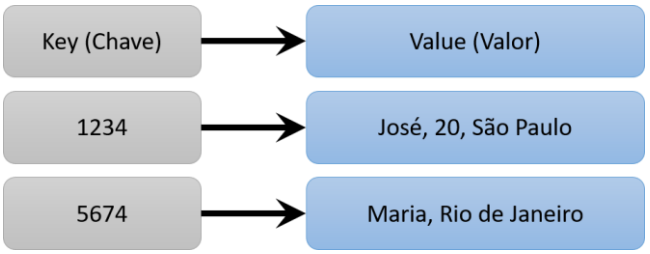
3.2. Categorias de Bancos de Dados NoSQL

Adicionalmente, devido aos diferentes cenários nos quais os bancos de dados NoSQL são necessários, diferentes tipos de soluções e abordagens foram criados e disponibilizados para o mercado. Os bancos de dados NoSQL podem ser distinguidos pelos modelos de dados que eles usam (BDW, 2014).

Armazenamento por chave-valor (key-value store)

O modelo mais simples tanto em termos de funcionamento quanto de entendimento são os bancos de dados do tipo chave-valor. Permitem o armazenamento de dados sem esquema definido. Os objetos de dados podem ser completamente não estruturados ou estruturados e são acessados por uma única chave. Neste tipo de modelo de dados NoSQL um determinado dado ou valor é acessado através de uma chave identificadora única (Figura 15). Estas chaves podem ser tratadas de forma literal ou armazenadas como hash's por parte dos bancos de dados, e os valores tipicamente são tratados como sequenciais binários, não sendo interpretados ou trados pelo banco de dados.

Figura 15 - Exemplo esquema NoSQL key-value.



O banco de dados chave-valor é livre de esquema, sendo que a aplicação-cliente fica responsável por resolver problemas de incompatibilidade, caso exista. Ao realizar a inserção de uma informação, é fornecida uma chave única que permite que esta informação seja recuperada em um momento futuro, porém não é possível realizar uma consulta baseando-se em alguma informação do valor armazenado. A recuperação, por fim, utiliza a mesma chave como acesso ao dado armazenado.

Devido a suas características de armazenamento e acesso à informação, o uso mais típico para os bancos de dados do tipo chave/valor é para armazenamento de cache, sessões, contadores de desempenho, filas de processamento e gestão de eventos dentro de uma aplicação.

Existem vários SGBDs que implementam o paradigma Chave-Valor (Key-Value), dentre eles citamos: DynamoDB[15], Redis[16], Riak[17], Voldemort[18] e Memcached[19].

[15] https://aws.amazon.com/pt/dynamodb/

[16] http://redis.io/

[17] http://basho.com/

[18] https://www.project-voldemort.com/voldemort/

[19] https://memcached.org/

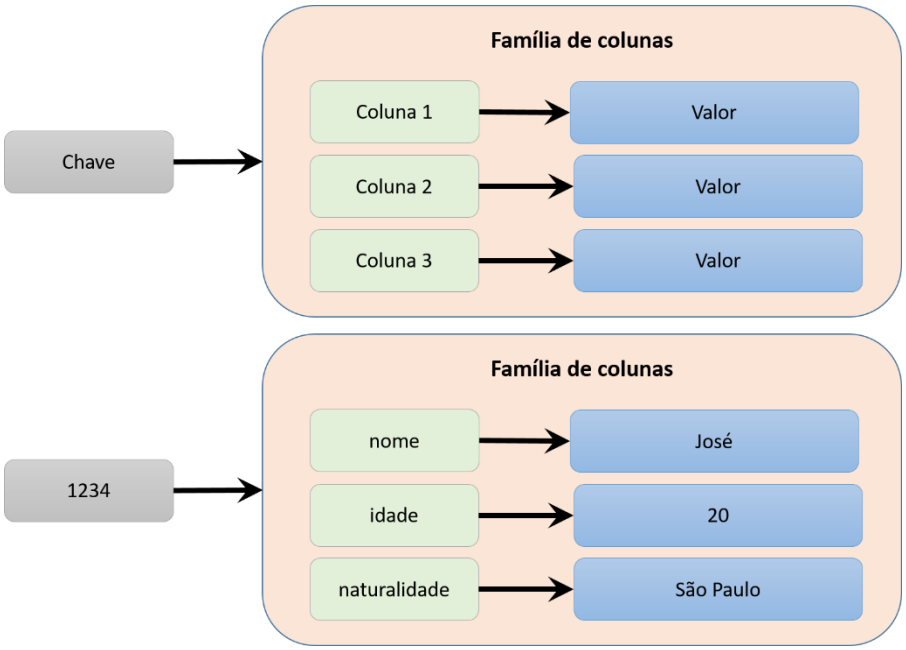
Armazenamento em coluna ou colunar (columnar stores)

O modelo de SGBD NoSQL colunar foram um dos primeiros a surgir com o objetivo de tratarem a não normalização de dados, usados especialmente para cenários em que um cenário relacional teríamos cada uma das tuplas (linhas) com apenas um determinado conjunto de dados inseridos. Estes SGBDs definem a estrutura de valores como um conjunto pré-definido de colunas, ou seja, permite o armazenamento de tabelas de dados como seções de colunas de dados e não como linhas de dados, como a maioria dos SGBD relacionais.

Este modelo foi inspirado pelo BigTable do Google, suportam várias linhas e colunas, e também permitem sub-colunas. Assim, ao invés de definir antecipadamente as colunas necessárias para armazenar um registro, o responsável pela modelagem de dados define o que é chamado de "famílias de colunas" (Figura 16).

As famílias de colunas são organizadas em grupos de itens de dados que são frequentemente usados em conjunto em uma aplicação. Nos SGBDs colunares, não é necessário que cada linha de dados possua o mesmo número de colunas, desta forma, esse modelo tem dados espaçados. Assim, existe maior flexibilidade de inserir as colunas que considerar necessárias em cada registro armazenado, sem precisar alterar a estrutura dos dados já armazenados. Desta forma, o banco de dados do tipo colunar apresenta um incontável número de linhas, porém cada uma dessas linhas utiliza apenas uma fração das colunas disponíveis na tabela.

Figura 16 - Exemplo esquema NoSQL colunar.



Esse modelo de banco de dados possui uma arquitetura especial para tratar e armazenar grandes volumes de dados distribuídos geograficamente. Além disso, ele é preparado para estar sempre disponível para escrita, uma vez que a replicação e o balanceamento dos dados entre os nós do cluster são realizados de forma transparente e automática. São recomendados e adequados para serem conectados a ferramentas de extração e análises de dados, capazes de tratar e identificar padrões e correlação de dados. Existem vários SGBDs que implementam o paradigma colunar, por exemplo: BigTable do Google[20], Cassandra[21] e HBase[22].

Armazenamento orientado a documentos (document databases)

Este tipo de modelo armazena coleções de documentos, ou seja, permitem o armazenamento de documentos estruturados, mas sem a exigência de requisito para

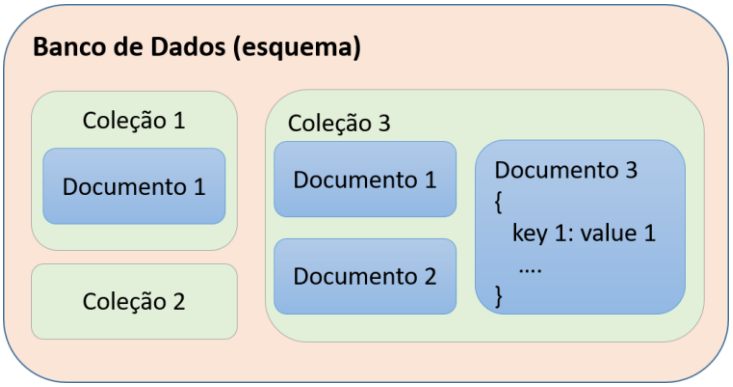
[20] https://cloud.google.com/bigtable/

[21] http://cassandra.apache.org/

[22] https://hbase.apache.org/

um esquema comum que todos os documentos devem aderir, como no caso de registros em bancos de dados relacionais. Os documentos são conjuntos de atributos e valores (semelhante ao esquema chave-valor), onde um atributo pode ser multivalorado. As chaves dentro dos documentos são únicas. Cada documento contém um identificador, que é único dentro do conjunto (Figura 17).

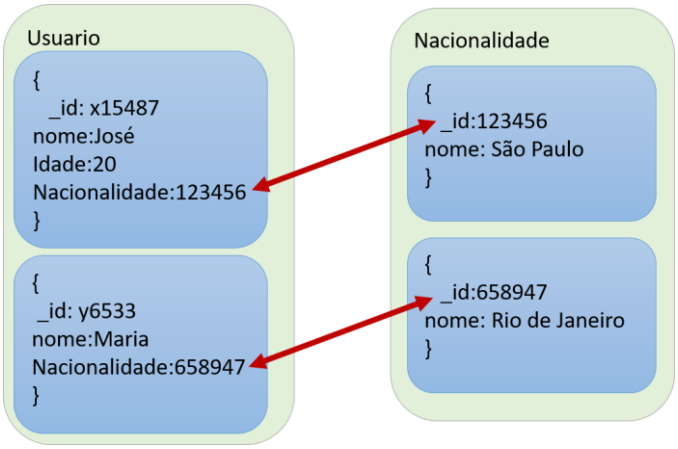
Figura 17 - Exemplo esquema NoSQL orientado a documentos.



Possibilitam a criação de índices sobre os valores dos dados armazenados, enriquecendo as possibilidades de consultas, além de permitir normalização de dados e muitos outros conceitos oriundos do banco de dados relacional, tais como a criação de joins e definição de esquemas rígidos.

Uma característica importante dos modelos NoSQL orientados a documentos a independência de um esquema rígido pré-definido, ou seja, não exige uma estrutura fixa como ocorre nos bancos relacionais. Esta flexibilidade permite a atualização na estrutura do documento, com a adição de novos campos, por exemplo, sem causar problemas ao banco de dados. Além disso, dois documentos de uma mesma coleção podem possuir um conjunto de chave-valores distinto. Cada documento possui uma chave única de identificação que pode ser usada como referência de relação entre documentos distintos (Figura 18).

Figura 18 - Exemplo de relacionamento em esquemas NoSQL orientado a documentos.



Esse modelo costuma armazenar os valores em uma estrutura como JSON (*JavaScript Object Notation*) ou XML (*Extensible Markup Language*). A possibilidade de possuir um esquema flexível na composição dos dados, permitindo que sejam aplicados filtros nos valores retornados, torna esse tipo de banco de dados ideal para cenários que reflitam uma grande granularidade de dados, como em um catálogo de produtos, conteúdo informativo ou inteligência operacional. Como é um modelo fácil manutenção são indicados por exemplo para aplicações web que precisam executar consultas dinâmicas, tais como aplicações de análise em tempo real e blogs. Dentre os diversos SGBDs orientado a documentos podemos citar: Couchbase[23], CouchDB[24], MongoDB[25], BigCouch[26].

[23] http://www.couchbase.com/

[24] http://couchdb.apache.org/

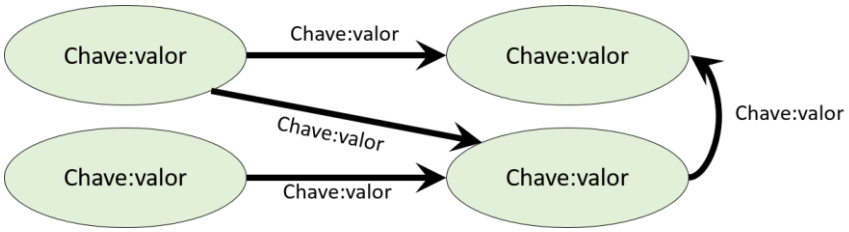
[25] https://www.mongodb.com

[26] https://bigcouch.cloudant.com/

Armazenamento orientados por grafos

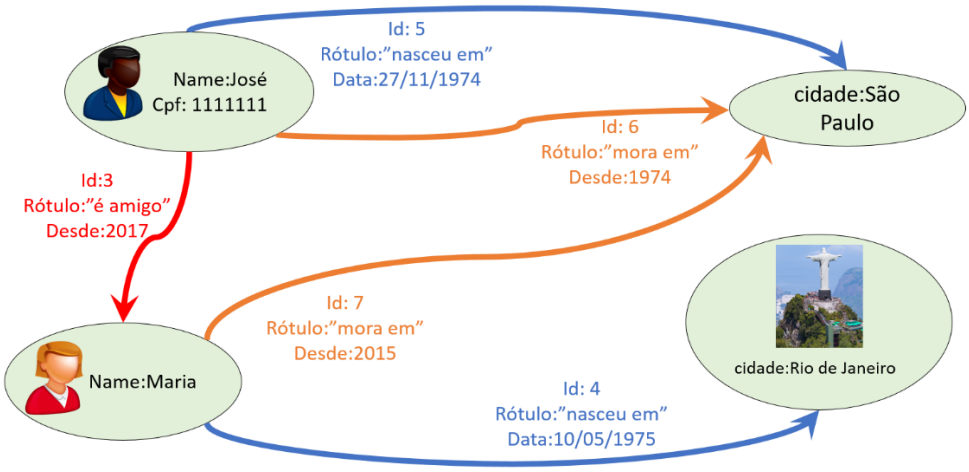
Os SGBDs NoSQL orientados por grafos armazenam os dados em estruturas definidas conforme a teoria dos grafos. Basicamente possuem três componentes básicos: os nós (são os vértices do grafo) para armazenar os dados dos itens coletados, as arestas que armazenam os relacionamentos entre os dados (vértices) e as propriedades (ou atributos) dos nós e relacionamentos. As propriedades são definidas conforme o par chave-valor (Figura 19).

Figura 19 - Exemplo esquema NoSQL orientado por grafos.



Os vértices e arestas podem ter múltiplas propriedades (Figura 20). Um conjunto de vértices conectados por meio de arestas definem um caminho no grafo. Este modelo suporta a utilização de restrições de integridade de dados, garantindo assim as relações entre elementos de forma consistente.

Figura 20 - Exemplo de dados organizados em um esquema NoSQL orientado por grafos.



Estes modelos são ideais para gerenciar relações entre diferentes objetos ou que se encontram em tipos de dados diferentes. Além disso, estes modelos são aplicáveis para modelos com muitos relacionamentos entre os dados, ou quando informações sobre a topologia ou interconectividade dos dados é importante, por exemplo, considerando que a web semântica se baseia na estrutura RDF, este modelo é bem apropriado. Dentre os diversos SGBDs orientado por grafos podemos citar: Virtuoso[27], Infinitegraph[28], AllegroGraph[29], Titan[30], ArangoDB[31] e Neo4J[32].

Por fim, deve-se ter em mente que um modelo NoSQL não deve ser considerado melhor que o outro, já que cada tipo de modelo pode ser mais adequado para determinadas aplicações. Os bancos de dados NoSQL tem sido amplamente adotado em empresas como Facebook, Amazon e Google com o intuito de atender às suas demandas de escalabilidade, alta disponibilidade e dados não estruturados.

---

27 https://virtuoso.openlinksw.com/

28 https://www.objectivity.com/products/infinitegraph/

29 http://franz.com/agraph/allegrograph/

30 https://titan.thinkaurelius.com/

31 https://www.arangodb.com/

32 https://neo4j.com/

Referências

ASLETT, M. *NoSQL, NewSQL and Beyond: The answer to sprained relational databases*. The 451 Group. On-line. 2019 2011. Disponível em: <https://blogs.the451group.com/information_management/2011/04/15/nosql-newsql-and-beyond/>. Data de accesso: 15 jun. 2020.

ATZORI, L.; IERA, A.; MORABITO, G. The internet of things: A survey. In: *Computer networks*, v. 54, n. 15, p. 2787-2805, 2010.

BARANAUSKAS, J. A. *Extração automática de conhecimento por múltiplos indutores*. 2001.

BARBIERI, C. P. *BI2 - Business Intelligence: Modelagem e Qualidade*. 1ª Edição. Rio de Janeiro: Elsevier, 2011.

BDW, B. D. W. *Introduction about NoSQL Data Models*. Big Data World. On-ine. 2019 2014. Disponível em: <http://amanbinny.blogspot.com/2014/11/introduction-about-nosql-data-models.html>. Acesso em: 15 jun. 2020.

BENEVENUTO, F.; ALMEIDA, J. M.; SILVA, A. S. Explorando redes sociais online: Da coleta e análise de grandes bases de dados às aplicações. In: *XXIX Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC)*, 2011, Campo Grande, Brasil. Sociedade Brasileira de Computação. p.63-102.

BERNERS-LEE, T. *Linked Data*. 2006. Disponível em: <https://www.w3.org/DesignIssues/LinkedData.html>. Acesso em: 15 jun. 2020.

BERNERS-LEE, T.; HENDLER, J.; LASSILA, O. The semantic web. In: *Scientific american*, v. 284, n. 5, p. 28-37, Apr 26 2001.

BINANI, S.; GUTTI, A.; UPADHYAY, S. SQL vs. NoSQL vs. NewSQL-A comparative study. In: *Database*, v. 6, n. 1, p. 1-4, 2016.

BIZER, C.; HEATH, T.; BERNERS-LEE, T. Linked data-the story so far. In: *Semantic Services, Interoperability and Web Applications: Emerging Concepts*, p. 205-227, 2009.

CANALTECH. *EMC oferece solução de armazenamento e análise de Data Lake* - Infra. On-line, 2015-04-03 2015. Disponível em: <https://canaltech.com.br/infra/EMC-oferece-solucao-de-armazenamento-e-analise-de-Data-Lake/>. Acesso em: 15 jun. 2020.

COSTA, L. H. M.; et al*. Grandes Massas de Dados na Nuvem:* Desafios e Técnicas para Inovação. 2012.

COULOURIS, G.; et al*. Sistemas Distribuídos-:* Conceitos e Projeto. Bookman Editora, 2013.

COX, M.; ELLSWORTH, D. Application-controlled demand paging for out-of-core visualization. In: YAGEL, R. and HAGEN, H., *8th conference on Visualization'97*, 1997, Phoenix, AZ, USA. IEEE Computer Society Press, October 18-24. p.235-ff.

DAMA. *The DAMA Guide to The Data Management Body of Knowledge (DAMA-DMBOK Guide).* The Data Management Association International (DAMA). 1ª ed. Bas King Ridge, New Jersey, USA: Technics Publications LLC, 2009. p.406.

_____. *DAMA-DMBOK:* Data management body of knowledge The Data Management Association International (DAMA). 2ª Ed. Bas King Ridge, New Jersey, USA: Technics Publications LLC, 2017. p.624.

DAVENPORT, T. H. *Ecologia da informação: por que só a tecnologia não basta para o sucesso na era da informação*. 2ª ed. São Paulo: Futura, 1998.

DUMBILL, E.; et al. *Big Data Now:* 2012 Edition. Sebastopol: O'Reilly Media, 2012.

DURDEN, T. *What Happens In An Internet Minute In 2019?* Zero Hedge. On-line. 2019 2019. Disponível em: <https://www.zerohedge.com/news/2019-03-13/what-happens-internet-minute-2019>. Acesso em: 15 jun. 2020.

ELMASRI, R.; NAVATHE, S. B. *Sistemas de banco de dados*. 4ª ed. São Paulo: Addison Wesley, 2005.

FAYYAD, U.; PIATETSKY-SHAPIRO, G.; SMYTH, P. The KDD process for extracting useful knowledge from volumes of data. In: *Communications of the Acm***,** v. 39, n. 11, p. 27-34, 1996a.

_____. *Knowledge Discovery and Data Mining*: Towards a Unifying Framework. KDD, 1996b. p.82-88.

FEW, S.; EDGE, P. Data visualization: past, present, and future. In: *IBM Cognos Innovation Center*, 2007.

FISHER, R. A. The use of multiple measurements in taxonomic problems. In: *Annals of eugenics,* v. 7, n. 2, p. 179-188, 1936.

FRANK, E.; HALL, M. A.; WITTEN, I. H. *The WEKA workbench*. Morgan Kaufmann, 2016.

GARRO, F. *O que é ETL e qual sua importância entre os processos de BI*. Blog IGTI. On-line: Blog IGTI. 2019 2017. Disponível em: <http://igti.com.br/blog/o-que-e-etl-bi/>. Acesso em: 15 jun. 2020.

GOLDMAN, A.; et al. Apache Hadoop: conceitos teóricos e práticos, evolução e novas possibilidades. In: *XXXI Jornadas de atualizações em informática*, p. 88-136, 2012.

GOLDSCHMIDT, R.; PASSOS, E. *Data mining: um guia prático*. Gulf Professional Publishing, 2005.

GUBBI, J.; et al. Internet of Things (IoT): A vision, architectural elements, and future directions. In: *Future Generation Computer Systems***,** v. 29, n. 7, p. 1645-1660, 2013.

HAN, J.; PEI, J.; KAMBER, M. *Data mining: concepts and techniques*. Elsevier, 2011.

HEATH, T.; BIZER, C. Linked data: Evolving the web into a global data space. In: *Synthesis lectures on the semantic web: theory and technology*, v. 1, n. 1, p. 1-136, 2011.

HEUSER, C. A. *Projeto de Banco de Dados.* 6ª ed. Porto Alegre: Bookman, 2008.

HUSSAIN, F.; HUAN, L. I. U.; TAN, C. L. *Discretization: An enabling technique*. 1999.

ISOTANI, S.; BITTENCOURT, I. I. *Dados Abertos Conectados: Em busca da Web do Conhecimento*. Novatec Editora, 2015.

KNAFLIC, C. N. *Storytelling with data: A data visualization guide for business professionals*. John Wiley & Sons, 2015.

LANEY, D. 3D data management: Controlling data volume, velocity and variety. In: *META Group Research Note*, v. 6, p. 70-73, 2001.

LAZER, D.; RADFORD, J. Data ex Machina: Introduction to Big Data. In: *Annual Review of Sociology*, v. 43, n. 1, p. 19-39, 2017/07/31 2017.

LYCETT, M. 'Datafication': making sense of (big) data in a complex world. In: *European Journal of Information Systems*, v. 22, p. 381-386, 2013.

MAYER-SCHÖNBERGER, V.; CUKIER, K. *Big data: a revolution that will transform how we live, work, and think*. Boston: Houghton Mifflin Harcourt, 2013.

MCAFEE, A.; et al. Big data: the management revolution. In: *Harvard business review*, v. 90, n. 10, p. 60-68, 2012.

MIORANDI, D.; et al. Internet of things: Vision, applications and research challenges. In: *Ad hoc networks*, v. 10, n. 7, p. 1497-1516, 2012.

OKI, O. K. I. *Guia de Dados Abertos*. On-line, 2019. Disponível em: <http://opendatahandbook.org/guide/pt_BR/>. Acesso em: 15 jun. 2020.

OLSON, M. Hadoop: Scalable, flexible data storage and analysis. In: *IQT quarterly*, v. 1, n. 3, p. 14-18, 2010.

PIMENTA, A.; et al. WEKA-G: mineração de dados paralela em grades computacionais. In: *Revista de Sistemas de Informação da FSMA*, v. 4, p. 2, 2009.

PYLE, D. *Data preparation for data mining*. Morgan Kaufmann, 1999.

QUINLAN, J. R. Induction of decision trees. In: *Machine learning*, v. 1, n. 1, p. 81-106, 1986.

RAJ, A.; D'SOUZA, R. *A Review on Hadoop Eco System for Big Data*. 2019.

SILBERSCHATZ, A.; KORTH, H. F.; SUDARSHAN, S. In: *Sistemas de banco de dados*. 6ª ed. São Paulo: Elsevier, 2012.

SLTI, S. d. L. e. T. d. I.; MPOG, M. d. P. O. e. G. *Cartilha Técnica para Publicação de Dados Abertos no Brasil*. 2019. Disponível em: <http://dados.gov.br/pagina/cartilha-publicacao-dados-abertos>. Acesso em: 15 jun. 2020.

SOUSA, F. R.; et al. Gerenciamento de dados em nuvem: Conceitos, sistemas e desafios. In: *Tópicos em sistemas colaborativos, interativos, multimídia, web e bancos de dados, Sociedade Brasileira de Computação*, p. 101-130, 2010.

SOUSA, F. R.; MOREIRA, L. O.; MACHADO, J. C. Computação em nuvem: Conceitos, tecnologias, aplicações e desafios. In: *II Escola Regional de Computação Ceará, Maranhão e Piauí (ERCEMAPI)*, p. 150-175, 2009.

STONEBRAKER, M. New opportunities for New SQL. In: *Communications of the Acm*, v. 55, n. 11, p. 10-11, 2012.

STROHBACH, M.; et al. Big Data Storage. In: CAVANILLAS, J. M.; CURRY, E. and WAHLSTER, W. (Ed.). *New Horizons for a Data-Driven Economy: A Roadmap for Usage and Exploitation of Big Data in Europe*. Cham: Springer International Publishing, 2016. p.119-141.

TAURION, C. *Cloud computing-computação em nuvem*. Brasport, 2009.

_____. *Big data*. Rio de Janeiro: Brasport, 2013.

TURCK, M. *Great Power, Great Responsibility: The 2018 Big Data & AI Landscape*. 2018. Disponível em: <http://mattturck.com/bigdata2018/>. Acesso em: 15 jun. 2020.

VAN DIJCK, J. Datafication, dataism and dataveillance: Big Data between scientific paradigm and ideology. In: *Surveillance & Society***,** v. 12, n. 2, p. 197-208, 2014.

W3C. *Resource Description Framework (RDF): Concepts and Abstract Syntax*. KLYNE, G.;CARROLL, J. J. and MCBRIDE, B.: World Wide Web Consortium, OWL Working Group 2004. Disponível em: <https://www.w3.org/TR/2004/REC-rdf-concepts-20040210/>. Acesso em: 15 jun. 2020.

_____. *RDF 1.1 Concepts and Abstract Syntax*. CYGANIAK, R.;WOOD, D. and LANTHALER, M.: World Wide Web Consortium, OWL Working Group 2014a. Disponível em: <https://www.w3.org/TR/2014/REC-rdf11-concepts-20140225/>. Acesso em: 15 jun. 2020.

_____. *RDF 1.1 Semantics*. HAYES, P. J. and PATEL-SCHNEIDER, P. F.: W3C Recommendation 2014b. Disponível em: <https://www.w3.org/TR/rdf11-mt/>. Acesso em: 15 jun. 2020.

WHITE, T. *Hadoop: The definitive guide*. O'Reilly Media, Inc., 2012.

WITTEN, I. H.; et al. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2016.

WU, X.; et al. Data mining with big data. In: *IEEE transactions on knowledge and data engineering*, v. 26, n. 1, p. 97-107, 2014.

ZHENG, Z. *Constructing new attributes for decision tree learning*. Citeseer, 1996.