

XIV
ERBD
Escola Regional de Banco de Dados
2018
Rio Grande-RS

RECUPERAÇÃO DE INFORMAÇÕES

ANAIS

Realização:



Organização:



XIV ESCOLA REGIONAL DE BANCO DE DADOS

9 a 11 de abril de 2018
Rio Grande – RS – Brasil

ANAIS

Realização

Sociedade Brasileira de Computação - SBC
Comissão Especial de Banco de Dados

Organização

Universidade Federal do Rio Grande - FURG

Comitê Diretivo da ERBD

Daniel dos Santos Kaster (UEL), Presidente
Daniel Luis Notari (UCS)
Cristiano Roberto Cervi (UPF)

Coordenação Geral

Eduardo Nunes Borges (FURG)

Coordenação do Comitê de Programa

Daniel dos Santos Kaster (UEL)
Karina dos Santos Machado (FURG)

ISSN: 2177-4226

Ficha catalográfica

E74a Escola Regional de Banco de Dados (14. : 2018 : Rio Grande/RS).
Anais [da] XIV Escola Regional de Banco de Dados - ERBD:
recuperação de informações [recurso eletrônico] / coordenação geral
Eduardo Nunes Borges ; coordenação do comitê Daniel dos Santos Kaster,
Karina dos Santos Machado. – Rio Grande : Ed. da FURG, 2018.
152 p.

Evento realizado no período de 09 a 11 de abril de 2018 na
Universidade Federal do Rio Grande - FURG.
Modo de acesso: www.sbc.org.br/erbd2018.
ISSN: 2177-4226

1. Banco de Dados - Congressos I. Borges, Eduardo Nunes
II. Kaster, Daniel dos Santos III. Machado, Karina dos Santos IV. Título.

CDU 519.68.023

Editorial

É com grande satisfação que apresentamos os artigos aceitos para a décima quarta edição da Escola Regional de Banco de Dados - ERBD e que compõem os anais do evento. Em 2018, a ERBD foi realizada de 9 a 11 de abril, na cidade de Rio Grande – RS com organização do Centro de Ciências Computacionais - C3 da Universidade Federal do Rio Grande - FURG. A ERBD é um evento anual realizado pela Sociedade Brasileira de Computação - SBC, que tem como objetivo principal integrar os participantes, oportunizando a divulgação e discussão de trabalhos em um fórum regional do sul do país sobre bancos de dados e áreas afins. Além das sessões técnicas, a programação do evento inclui oficinas, minicursos e palestras proferidas por pesquisadores de renome da comunidade brasileira.

Mantendo a tradição das edições anteriores da ERBD, foram aceitas submissões de artigos em duas categorias: Pesquisa e Aplicações/Experiências. Todos os artigos foram avaliados por pelo menos três membros do Comitê de Programa. A categoria Pesquisa recebeu 20 submissões, das quais 11 foram aceitas como artigo completo e uma como artigo curto (Aplicações/Experiências), o que representa 60% de taxa de aceitação. Cada artigo aceito nesta categoria foi apresentado em 20 minutos nas sessões técnicas. A categoria de Aplicações/Experiências recebeu cinco submissões, das quais três foram aceitas, o que representa 60% de taxa de aceitação. Os quatro artigos aceitos nesta categoria foram apresentados em 10 minutos nas sessões técnicas, bem como na forma de pôster.

Os Anais da XIV ERBD representam o resultado do esforço coletivo de um grande número de pessoas. Agradecemos ao Comitê de Organização Local da ERBD, coordenado pelo Prof. Eduardo Nunes Borges, que trabalhou arduamente para garantir o bom andamento do evento. Gostaríamos de agradecer também aos membros do Comitê de Programa que realizaram revisões de excelente qualidade. Finalmente, agradecemos aos autores que submeteram seus trabalhos para a ERBD.

Daniel dos Santos Kaster (UEL)
Coordenador do Comitê de Programa
Karina dos Santos Machado (FURG)
Coordenadora do Comitê de Programa

Carta do Coordenador Geral

Com muito orgulho realizamos a XIV edição da Escola Regional de Banco de Dados – ERBD em Rio Grande, nas instalações do Centro de Ciência Computacionais – C3 da Universidade Federal do Rio Grande – FURG. Nesta edição, a ERBD contou com aproximadamente 200 participantes, incluindo estudantes de ensino técnico, de graduação e de pós-graduação, bem como profissionais da academia e da indústria de TI de Rio Grande e da região sul do país.

O tema Recuperação de Informações foi o escolhido para articulação das palestras, minicursos, oficiais, painel e sessões técnicas com apresentação de artigos das categorias pesquisa e aplicações/experiências. Ao todo, 15 palestrantes e 15 autores de artigos colaboraram com suas experiências acadêmicas e profissionais, proporcionando um evento de altíssimo nível, contribuindo para a atualização e qualificação do público que prestigiou o evento.

O grupo de professores, técnicos administrativos e estudantes envolvidos na organização foi fundamental para que a ERBD 2018 fosse executada com sucesso. Muito obrigado pela disponibilidade e responsabilidade na execução das mais diversas tarefas, de forma voluntária e colaborativa. Agradeço ainda ao Comitê Diretivo da ERBD, à Comissão Especial de Banco de Dados e aos demais colaboradores da Sociedade Brasileira de Computação. Por fim, meu agradecimento especial à Universidade Federal do Rio Grande e ao Centro de Ciências Computacionais, por terem cedido estrutura física e recursos humanos para todo o suporte necessário à realização do evento.

Eduardo Nunes Borges (FURG)
Coordenador Geral

XIV ESCOLA REGIONAL DE BANCO DE DADOS

9 a 11 de abril de 2018
Rio Grande – RS – Brasil

Realização

Sociedade Brasileira de Computação - SBC
Comissão Especial de Banco de Dados

Organização

Universidade Federal do Rio Grande - FURG

Comitê Diretivo da ERBD

Daniel dos Santos Kaster (UEL), Presidente
Daniel Luis Notari (UCS)
Cristiano Roberto Cervi (UPF)

Coordenação Geral

Eduardo Nunes Borges (FURG)

Coordenações

Comitê de Programa: Daniel dos Santos Kaster (UEL) e Karina dos Santos Machado (FURG)

Financeira: Diana Francisca Adamatti (FURG)

Palestras: Carina Friedrich Dorneles (UFSC) e André Prisco Vargas (FURG)

Minicursos: Daniel Lichtnow (UFSM) e Ana Marilza Pernas (UFPel)

Oficinas: Denio Duarte (UFFS) e Igor Avila Pereira (IFRS)

Articulação com empresas: Ana Marilza Pernas (UFPel), Ana Trindade Winck (UFCSPA) e Luciano Maciel Ribeiro (FURG)

Premiação: Karina dos Santos Machado (FURG), Guilherme dal Bianco (UFFS) e Helena Graziottin Ribeiro (UCS)

Sessões Técnicas: Ana Trindade Winck (UFCSPA), Edimar Manica (IFRS), Helena Graziottin Ribeiro (UCS), Karina dos Santos Machado (FURG) e Luiz Celso Gomes Junior (UTFPR)

Infraestrutura e Logística: Alessandro de Lima Bicho (FURG), Bruno Todeschini de Oliveira (FURG), João Mateus Daltro de Athayde (FURG), Jônata Tyska Carvalho (FURG), Paulo Francisco Butzen (FURG)

Comitê Local

Eduardo Nunes Borges (FURG), Coordenação
Angélica Theis dos Santos (FURG), Coordenação
Anajara Arvelos Martins (FURG)
Bianca Parulla Marques (FURG)

Carlos Alberto Suzano do Nascimento da Silva Longo (FURG)
Caroline Waschburger dos Santos (FURG)
Cedenir Borges da Costa (FURG)
Everson da Silva Flores (FURG)
Fernanda Luiz Pinto (FURG)
João Mateus Datto de Athaide (FURG)
Jônata Tyska Carvalho (FURG)
Marcio Sarres Pessoa (FURG)
Maria Carmen Bitencourt Carvalho (FURG)
Matheus Baranano e Silva (FURG)
Nitchele dos Reis da Gama (FURG)
Pamela Oliveira Borges (FURG)
Prícila Karen Suzano do Nascimento da Silva Longo (FURG)
Rafael Vianna Oliveira (FURG)
Vinícius Borges Martins (FURG)
Volni Afonso Silveira (FURG)

Comitê de Programa

Alcides Calsavara (PUCPR)
Ana Marilza Pernas (UFPel)
Ana Trindade Winck (UFCSPA)
André Schwerz (UTFPR)
André Prisco Vargas (FURG)
Angelo Frozza (IFC)
Carina Friedrich Dorneles (UFSC)
Carmem S. Hara (UFPR)
Cristiano Roberto Cervi (UPF)
Daniel dos Santos Kaster (UEL)
Daniel Lichtnow (UFSM)
Daniel Luis Notari (UCS)
Deborah Carvalho (PUCPR)
Deise Saccol (UFSM)
Denio Duarte (UFFS)
Eder Pazinatto (UPF)
Edimar Manica (IFRS)
Eduardo Nunes Borges (FURG)
Eduardo Cunha de Almeida (UFPR)
Fernando José Braz (IFC)
Flávio Uber (UEM-UFPR)
Guilherme dal Bianco (UFFS)
Guillermo Nudelman Hess (FEEVALE)
Gustavo Zanini Kantorski (UFSM)
Helena Graziottin Ribeiro (UCS)
João Marynovski (PUCPR)
José Maurício Carré Maciel (UPF)
Karin Becker (UFRGS)
Karina dos Santos Machado (FURG)

Luiz Celso Gomes Junior (UTFPR)
Marcos Aurélio Carrero (UFPR)
Nádia Kozievitch (UTFPR)
Raquel Stasiu (PUCPR)
Raqueline Penteado (UEM-UFPR)
Rebeca Schroeder (UDESC)
Regina Barwaldt (FURG)
Regis Schuch (UFSM)
Renata Galante (UFRGS)
Renato Fileto (UFSC)
Ronaldo Mello (UFSC)
Sandro Camargo (UNIPAMPA)
Scheila de Ávila e Silva (UCS)
Sergio L. S. Mergen (UFSM)
Solange de Lurdes Pertile (UFSM)

Sumário

Artigos Completos de Pesquisa	10
Artigos Curtos de Aplicações/Experiências	122
Palestras convidadas	139
Minicursos	142
Oficinas	147

Artigos Completos de Pesquisa

Completos

Caracterização dos Dados Públicos de Saúde do Paraguai	12
<i>Matheus Oliveira (Universidade Tecnológica Federal do Paraná - Brasil), Nádia Kozievitch (Universidade Tecnológica Federal do Paraná - Brasil), Silvia Amelia Bim (Universidade Tecnológica Federal do Paraná - Brasil), Horacio Legal-Ayala (Universidad Nacional de Asunción - Paraguai)</i>	
Análise de evolução de emissão de alvarás próximos a dois shoppings em Curitiba	
22	
<i>Yuri Bichibichi (Universidade Tecnológica Federal Do Paraná - Brasil), Nádia Kozievitch (Universidade Tecnológica Federal do Paraná - Brasil), Renata Carvalho (Universidade Tecnológica Federal do Paraná - Brasil)</i>	
Comparação entre MySQL e Neo4J para o Acesso a Dados Complexos Usando Linguagens Declarativas	32
<i>Emerson Homrich (Universidade Federal de Santa Maria - Brasil) e Sergio Mergen (Universidade Federal de Santa Maria - Brasil)</i>	
Otimização do Mapeamento de Consultas SPARQL para SQL	42
<i>Mariana Machado Garcez Duarte (Universidade Federal do Paraná - Brasil) e Carmem Hara (Universidade Federal do Paraná - Brasil)</i>	
Comparação entre Diferentes Implementações de BK-trees para o Problema de Busca por Intervalo	52
<i>André Luciano Rakowski (Universidade Federal de Santa Maria - Brasil), Natan Luiz Paetzhold Berwaldt (Universidade Federal de Santa Maria - Brasil), Mauricio Vielmo Schmaedeck (Universidade Federal de Santa Maria - Brasil), Sergio Luis Sardi Mergen (Universidade Federal de Santa Maria - Brasil)</i>	
Predição do volume de atendimentos de saúde na cidade de Curitiba utilizando dados abertos	62
<i>Mayara Lorenzi (Universidade Tecnológica Federal do Paraná - Brasil), Cristiano da Cunha Ribas (Universidade Tecnológica Federal do Paraná - Brasil), Luiz Celso Gomes Jr (Universidade Tecnológica Federal do Paraná - Brasil)</i>	
Predição de Indicadores Zootécnicos de Carcaças Bovinas a Partir de Variáveis de Cria	72
<i>Thales Vaz Maciel (Instituto Federal de Educação, Ciência e Tecnologia Sul-rio-</i>	

Grandense - Brasil), Vinícius Lampert (Empresa Brasileira de Pesquisa Agropecuária - Brasil), Denizar Souza (Universidade da Região da Campanha - Brasil), Rodrigo Silva (Instituto Federal de Educação, Ciência e Tecnologia Sul-rio-Grandense - Brasil)

Diferenciação de Perfis de Curva de Carga para Identificação de Perdas Não-Técnicas em Redes de Distribuição Utilizando Mineração de Dados e Aprendizado de Máquina
82

Jorge Sandoval Simão (Universidade do Vale do Itajaí - Brasil) e Raimundo Teive (Universidade do Vale do Itajaí - Brasil)

Aplicações de Mineração de Dados na Pecuária de Corte: Previsão de Indicadores de Qualidade de Carcaças **92**
Rodrigo Silva (Instituto Federal de Educação, Ciência e Tecnologia Sul-rio-Grandense - Brasil), Thales Vaz Maciel (Instituto Federal de Educação, Ciência e Tecnologia Sul-rio-Grandense - Brasil), Vinícius Lampert (Empresa Brasileira de Pesquisa Agropecuária - Brasil), Denizar Souza (Universidade da Região da Campanha - Brasil)

Análise da Popularidade de Tuítes com Base em Características Extraídas de seu Conteúdo **102**
Lucas Lima de Oliveira (Universidade Federal de Santa Maria - Brasil) e Sergio Mergen (Universidade Federal de Santa Maria - Brasil)

Extração de elementos textuais em imagens capturadas por smartphones: análise da relação entre as características das imagens e a eficácia da extração **112**
Daniel Kuhn (Universidade de Passo Fundo - Brasil), Cristiano Cervi (Universidade de Passo Fundo - Brasil), Edimar Manica (Instituto Federal do Rio Grande do Sul e - Brasil)

aper:180089_1

Caracterização dos Dados Públicos de Saúde do Paraguai

**Matheus F. A. de Oliveira¹, Nádia P. Kozievitch¹, Silvia A. Bim¹,
Horacio Legal-Ayala²**

¹Universidade Tecnológica Federal do Paraná (UTFPR)
Curitiba, Paraná, Brasil.

²Universidad Nacional de Asunción (UNA)
San Lorenzo, Central, Paraguay

matheus.2016@alunos.utfpr.edu.br, nadiap@utfpr.edu.br,
sabim@utfpr.edu.br, hlegal@pol.una.py

Abstract. *The growth of population density of urban regions requires an adequate provision of basic health services and infrastructure. This increase of the population demonstrates challenges to cities, mostly in the question of public health. This paper presents an analysis of the open data of Paraguay that is related to healthcare, from its availability, their establishments, professionals and products, to a general and dynamic analysis of the information contained in this open data, finishing with a comparison with the city of Curitiba.*

Resumo. *O aumento da densidade demográfica no ambiente urbano requer uma provisão adequada de serviços básicos de saúde e infraestrutura. Esse crescimento da população apresenta desafios às cidades, principalmente no quesito dos serviços de saúde pública. Este artigo traz uma análise dos dados abertos do Paraguai que são relacionados à saúde, desde sua disponibilização, estabelecimentos, profissionais e produtos, até uma análise geral e dinâmica das informações contidas nos dados abertos, finalizando com uma comparação com a cidade de Curitiba.*

1. Introdução.

A busca por crescimento econômico e desenvolvimento sustentável é de alto valor para todas as esferas governamentais de um país, como seus estados, municípios, e suas cidades. Os países que hoje buscam por esse desenvolvimento sustentável, por tecnologias para melhorar a vida de seus cidadãos, se relacionam, de uma maneira ou de outra, com o uso e aplicação de dados abertos no dia a dia.

O uso de dados abertos vem aumentando mundialmente. Dentre as diversas categorias que se encontra em dados abertos, a saúde pública é fundamental, pois envolve toda a população, em maior ou menor intensidade. Como exemplo, temos cidades como Curitiba¹, e países como Paraguai², que disponibilizam portais de acesso aos dados abertos de diversas áreas que existem na sociedade atual, sendo eles: transporte, saúde, segurança pública, entre outros. Além disso, criam leis e realizam

¹ <http://www.curitiba.pr.gov.br/DADOSABERTOS/> Acesso em 23 de Outubro. 2017.

² <https://www.datos.gov.py/> Acesso em 23 de Outubro de 2017.

eventos como competições e *hackathons*, que promovem o conhecimento sobre dados abertos à população.

O objetivo deste trabalho é analisar os dados abertos de saúde pública do Paraguai, e usar os dados referentes à Assunção para comparar com a cidade de Curitiba, no Brasil. O resto do trabalho está organizado da seguinte forma: a Seção 2 apresenta trabalhos relacionados. A metodologia é apresentada na Seção 3 e a análise dos dados do Paraguai encontra-se na Seção 4. A análise de Assunção e Curitiba encontram-se na Seção 5, seguida pela Conclusão na Seção 6.

1.1. Características: Assunção e Curitiba.

Assunção é a capital da República do Paraguai, situada na América do Sul, e faz fronteira com o Brasil, Argentina e Bolívia. A previsão de população do Paraguai em 2017 é de 6 953 646 habitantes³, tendo Assunção com uma densidade populacional de mais de 4.499 habitantes por km² de acordo com o Anuário Estatístico⁴ de 2015. Somente em 2015, com o decreto 4064/15⁵, que o Paraguai iniciou os trabalhos com dados abertos. O decreto regulamenta o livre acesso dos cidadãos à informação pública, por meio de um portal disponibilizado pelo governo.

Curitiba é capital do Paraná, estado localizado no sul do Brasil, com uma área de 430,9 km² e com uma população de 1,8 milhões de pessoas, de acordo com o IBGE⁶. Curitiba iniciou o trabalho com dados abertos em 2011, com a Lei Federal de número 12.527⁷. O acesso aos dados é feito, principalmente, através do portal da Prefeitura.

2. Trabalhos Relacionados.

A aplicação do uso dos dados abertos para melhorias à população pode ser vista na administração das cidades inteligentes (*smart cities*). O conceito de cidade inteligente está associado ao uso de tecnologias para aumentar a eficiência dos serviços prestados. [Carvalho et al., 2016] defende a ideia de que a qualidade de vida de uma população está cada vez mais dependente dessas cidades, de sua maneira de governar e de superar desafios. Todo esse processo de desenvolvimento envolvido nas cidades inteligentes com o uso de dados abertos é interligado com crescimento científico e tecnológico. [Molloy 2011] comenta que os dados carregam evidências para este conhecimento tecnológico, que é uma base para todo o crescimento científico. Dessa maneira, a associação dos dados abertos em conjunto com o desenvolvimento de cidades e países vem crescendo por todo o mundo, o que acarretou uma necessidade maior de definição e controle dos dados abertos relacionados.

³

<http://www.dgeec.gov.py/Publicaciones/Biblioteca/proyecction%20nacional/Estimacion%20y%20proyecction%20Nacional.pdf> Acesso em 09 de Novembro de 2017.

⁴ <http://www.dgeec.gov.py/Publicaciones/Biblioteca/anuario2015/Anuario%20Estadistico%202015.pdf> Acesso em 07 de Dezembro de 2017

⁵ <http://gestordocumental.senatics.gov.py/share/s/rcDa1BG7TRyZabwuPD5xEw> Acesso em 24 de Outubro de 2017

⁶ <https://www.ibge.gov.br/> Acesso em 21 de Novembro de 2017.

⁷ http://www.planalto.gov.br/ccivil_03/_Ato2011-2014/2011/Lei/L12527.htm Acesso em 22 de Novembro de 2017.

A *Open Knowledge Foundation*⁸ define que dados abertos são as informações que mantém um livre acesso, que podem ser modificadas, usadas e compartilhadas por qualquer motivo, desde que se mantenha a integridade dos dados e que continuem abertos após o uso. A sua disponibilização, entretanto, não é o suficiente. [Aló 2009] destaca a importância das características que os dados abertos precisam ter, como qualidade e transparência das informações fornecidas. Além disso, [Freitas et al., 2005] propõe o uso de técnicas auxiliares para a descoberta de novas informações ocultas e que asseguram a originalidade e a qualidade dos dados obtidos.

A saúde e a indústria da assistência médica é historicamente a categoria de dados abertos que mais geram informações, sejam eles sobre registros médicos, cuidados médicos com o paciente, controle de doenças, etc. O registro de dados sobre a área, de acordo com [Raghupathi et al., 2014], é de grande potencial, pois reduz os custos e melhora o acesso ao serviço pela população. Uma variedade imensa de técnicas e tecnologias é desenvolvida para analisar essa grande quantidade de dados. [Manyika et al., 2011] comenta que tais técnicas e tecnologias utilizam de diversos campos da ciência, envolvendo estatísticas, matemática, economia e computação. Das diversas técnicas comentadas por [Manyika et al., 2011], a classificação dos dados é uma delas. Categorizar os dados permite reconhecer padrões e potenciais problemas na forma em como o serviço que gerou aquele dado funciona.

Um exemplo de funcionamento dessas técnicas é demonstrada por [Nakonetchnel et al., 2017], que realiza uma análise de dados abertos na perspectiva de Curitiba e New York. Nesta direção, [Flores et al., 2017] realiza um estudo de acidentes de trânsito com base em dados abertos disponibilizado pelo governo do Rio Grande do Sul. Além destes trabalhos, há também uma pesquisa realizada no Paraguai, onde [Páne et al., 2016] demonstra uma análise dos dados em relação a epidemia de dengue no país.

3. Metodologia.

Apesar do portal de dados abertos do governo do Paraguai, as pesquisas só puderam ser realizadas através do portal disponibilizado pelo *Ministerio de Salud Publica y Bienestar Social*⁹ (MSPBS), que conta com uma visualização online dos dados. Porém, as informações contidas nos arquivos utilizados continham metadados além daqueles que podem ser vistos no site.

Como a divulgação dos dados abertos do Paraguai só se iniciou em 2015, as informações presentes nos dados abertos em relação a saúde pública são divididas em anos, a partir de 2015. Não existe notificação da atualização de tais arquivos (a última vez que foram alterados ou de quanto em quanto tempo são atualizados). Para a análise dos dados nos arquivos, usou-se o PostgreSQL¹⁰ e QGIS¹¹.

4. Análise dos Dados de Saúde Pública do Paraguai.

Os dados da saúde pública são disponibilizados e separados em três categorias, sendo elas *Productos* (Produtos), *Establecimientos* (Estabelecimentos) e *Servicios* (Serviços).

⁸ <https://okfn.org/> Acesso em 21 de Novembro de 2017

⁹ <http://www.mspbs.gov.py/> Acesso em 29 de Novembro de 2017

¹⁰ <https://www.postgresql.org/>& Acesso em 7 de Novembro de 2017.

¹¹ http://www.qgis.org/pt_BR/site/ Acesso em 23 de Outubro de 2017

A seção de produtos refere-se a todos os medicamentos que circulam pelo país, assim como um registro e descrição de cada um deles, enquanto a parte de estabelecimentos envolve todos estabelecimentos de saúde pública do governo, desde hospitais até centros de saúde, seus endereços, responsáveis e horários de atendimento. Os serviços de saúde são separados por categorias, e relacionados a cada estabelecimento registrado. Todas as três categorias se relacionam entre si, de forma a ter uma informação mais dinâmica entre os dados fornecidos.

As informações disponibilizadas pelo MSPBS estão disponíveis em dois tipos de arquivos, CSV (*Comma Separated Valued*) e JSON (*JavaScript Object Notation*). As informações que estão contidas nas próximas subseções foram feitas através da análise dos arquivos em formato CSV.

Das três categorias, Estabelecimentos é a única com dados georreferenciados. Nestes portais é possível visualizar informações através do site disponibilizado pelo DGEEyC¹² (Dirección General de Estadística, Encuestas y Censos). Entretanto, esta visualização georreferenciada se encontra desatualizada, com seus últimos dados sendo do ano de 2012.

Produtos. As informações relacionadas aos produtos são disponibilizadas em três arquivos distintos: produtos, disponibilidade de produtos e histórico de produtos. No arquivo de produtos, estão relacionadas às características físicas de cada produto, como nome, forma de disponibilização, concentração do medicamento, código DNCP (*Dirección Nacional de Contrataciones Publicas*) e forma de aquisição (por lista básica ou doação). A Figura 1 mostra um exemplo do registro de alguma dessas informações.

Nome do Produto	Tipo de Produto	Estado	Forma Farmacéutica	Apresentação	Classificação do Produto	Concentração
BICARBONATO DE SODIO	MEDICAMENTO	DESHABILITADO	INYECTABLE	SACHET X 100 ML	SANGRE Y ORGANOS FORMADORES DE SANGRE	8.4 GR / 100 ML
MEBENDAZOL	MEDICAMENTO	HABILITADO	SUSPENSION	FRASCO		100 MG/5ML

Figura 1. Informações dos medicamentos no arquivo produtos.

O segundo arquivo contém informações sobre a disponibilidade de produtos por estabelecimento, relacionando a forma como este estabelecimento adquiriu o produto, com que frequência esse produto é reposto, e informações gerais do estabelecimento, como nome, bairro, distrito a qual é localizado. Essas informações podem ser vistas na Figura 2.

Nome do Produto	Tipo de Produto	Nome do Estabelecimento	Tipo de Estabelecimento	Disponibilidade
SOLUCION RINGER CON LACTATO	MEDICAMENTO	USF-LOMAI	UNIDAD DE SALUD FAMILIAR	CONSULTE EN EL ESTABLECIMIENTO
ACETATO DE MEDROXIPROGESTERONA	MEDICAMENTO	H.D.-SAN ESTANISLAO	HOSPITAL DISTRITAL	CONSULTE EN EL ESTABLECIMIENTO

Figura 2. Exemplo da relação produto e estabelecimento.

É importante ressaltar que um mesmo produto pode ter diferentes formatos de apresentação (Frasco ou Creme), ou em diferentes quantidades (500mg ou 250mg). Para

¹² <http://geo.stp.gov.py/user/dgeec/datasets?page=1> Acesso em 09 de Fevereiro de 2018.

cada versão são registrados novos produtos nos dados abertos. Na Figura 3 são apresentados os cinco produtos com mais variações registradas encontradas.

Ao todo são 1275 medicamentos listados e um total de 764 em circulação (habilitados). A forma como esses dados são armazenados conforme o tempo é relacionado com o terceiro arquivo, que envolve o histórico do produto ao longo dos anos (a partir de 2015). Ele relaciona o estabelecimento e as datas com as quais os produtos são movimentados (entrada e saída, envolvendo dia, mês, ano e os horários).

Quantidade	Nome do Produto	Tipo de Produto
13	AMOXICILINA	MEDICAMENTO
12	RISPERIDONA	MEDICAMENTO
12	METRONIDAZOL	MEDICAMENTO
11	SOLUCION DEXTROSA HIPERTONICA	MEDICAMENTO
11	PARACETAMOL (ACETAMINOFENO)	MEDICAMENTO

Figura 3. Os cinco produtos com mais variações na forma de disponibilização.

Estabelecimentos. Os estabelecimentos públicos são registrados de maneira similar a dos produtos. Possuem um arquivo exclusivo para informações com as características de cada estabelecimento público de saúde, como nome, município, distrito, rua, acesso à internet, telefone, e horários de funcionamento. Um exemplo pode ser visualizado na Figura 4.

Nome	Região	Distrito	Tipo	Municipio	Estado
H.D.- GRAL AQUINO	SAN PEDRO SUR	GRAL. AQUINO - SPS	HOSPITAL DISTRITAL	GRAL. ELIZARDO AQUINO	ACTIVO
H.R.- CAACUPE	CORDILLERA	CAACUPE	HOSPITAL REGIONAL	CAACUPE	ACTIVO

Figura 4. Algumas Informações dos Estabelecimentos de Saúde do Paraguai.

Na Figura 5 temos as divisões por tipo de estabelecimentos. A categorização prioriza determinados tipos de serviços e público-alvo. Em conjunto com as informações a respeito do estado atual dos estabelecimentos, foi possível descobrir que apenas duas estão fora de operação no momento, de um total de 174 estabelecimentos registrados.

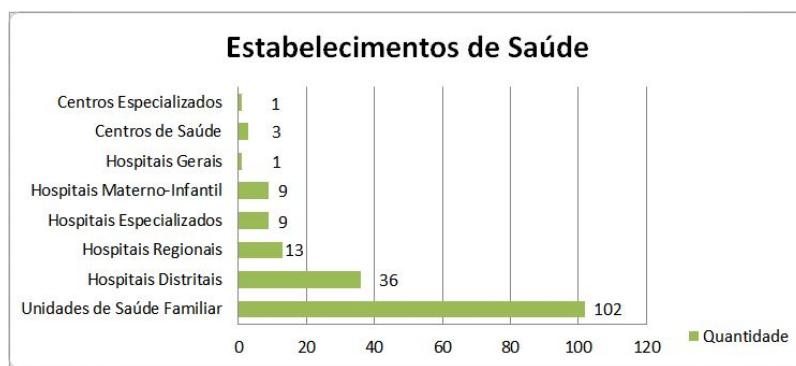


Figura 5. Gráfico dos estabelecimentos de saúde.

Serviços. Os serviços de saúde disponibilizados pelo país estão divididos entre serviços e suas características, e as relações entre os serviços e os estabelecimentos. A caracterização dos serviços é dada apenas pelo seu nome, categoria e um código auxiliar. A Figura 6 apresenta alguns exemplos de serviços baseado nessas classes.

Nome da Classe	Nome do Serviço
UNIDAD DE SALUD DE LA FAMILIA	ODONTOLOGIA
URGENCIAS	URGENCIAS NEUROCIRUGIA
CONSULTORIOS	PSICOLOGIA NIÑOS
INTERNADOS	CURGIA VASCULAR PERIFERICA
PROGRAMAS DE SALUD	DIABETES
METODOS AUXILIARES DE DIAGNOSTICO	RAYOS X
OTROS SERVICIOS	FISIOTERAPIA
SERVICIO SOCIAL	REGISTRO CIVIL

Figura 6. Exemplos de serviços para cada classe existente.

Ao todo são ofertados 277 serviços médicos separados em 8 categorias, que abrangem diversas áreas médicas. As oito categorias podem ser vistas na Figura 6. A disponibilidade destes serviços é dada através de outra tabela, contendo informações dos estabelecimentos e dos serviços ofertados. A Figura 7 mostra algumas destas informações.

Nome do Estabelecimento	Tipo de Estabelecimento	Região	Distrito	Tipo de Profissional	Dia da Semana	Serviço	Nome da Categoria
HMI - SAN LORENZO	HOSPITAL MATERNO INFANTIL	CENTRAL	SAN LORENZO	MEDICO/A CIRUJANO/A	SÁBADO	GASTROENTEROLOGIA	CONSULTORIOS
HOSPITAL GENERAL BARRIO OBRERO	HOSPITAL GENERAL	CAPITAL	ASUNCION	DOCTOR/A EM MEDICINA Y CIRUGIA	LUNES	GASTROENTEROLOGIA	CONSULTORIOS
H.R. - CONCEPCION	HOSPITAL REGIONAL	CONCEPCION	CONCEPCION	LICENCIADO/A EM MEDICINA	LUNES	CLINICA MEDICA	CONSULTORIOS

Figura 7. Algumas informações da disponibilidade dos serviços públicos de saúde.

Todos os serviços são registrados baseados no profissional de saúde que realizou este serviço, no dia da semana em que foi realizado, contendo informações dos estabelecimentos de saúde (nome, tipo, região e distrito).

Os dados também contêm outras informações, como códigos do estabelecimento e do serviço, horário em que o serviço foi realizado, e inclusive, o nome do profissional de saúde que realizou o serviço. Além disso, também contém o nome do profissional de saúde responsável por aquele serviço dentro de determinado estabelecimento.

Cerca de 1,7% dos dados não tem informações sobre o nome do profissional de saúde responsável pelo serviço no estabelecimento, e 44% não contém informações sobre o nome de um (a) médico (a) que realizou o serviço nos estabelecimentos.

Para podermos ter uma visão mais ampla de como os serviços públicos estão sendo utilizados, realizou-se uma consulta com o objetivo de obter os serviços mais registrados, que pode ser visto na Figura 8.

Nome do Serviço	Quantidade
VACUNATORIO	1299
PEDIATRIA	1206
ODONTOLOGIA	974
CURACIONES	960
FARMACIA	955

Figura 8. Os cinco serviços mais registrados.

Na Figura 8 é possível perceber certa variedade dos serviços registrados, variando entre pediatria e farmácia. Para uma maior completude dessas informações,

também buscou-se os tipos de profissionais de saúde com maior quantidade de registros. O resultado da consulta se encontra na Figura 9.

Tipo de Profissional de Saúde	Quantidade
MEDICO/A CIRUJANO/A	4501
DOCTOR/A EN MEDICINA Y CIRUGIA	3122
DR. EN MEDICINA Y CIRUGIA	2785
DOCTOR/A EN MEDICINA	952
LICENCIADO/A EN ENFERMERIA	762

Figura 9. Os cinco tipos de profissional de saúde com maior participação.

Para concluir essa pesquisa, buscou-se por fim os hospitais públicos que mais realizaram serviços, e consequentemente, sua região e categoria. O resultado da busca está presente na Figura 10.

Nome do Hospital Público	Região	Tipo de Hospital	Quantidade
HOSPITAL NACIONAL	CENTRAL	HOSPITAL ESPECIALIZADO	1177
H.G. - LUQUE	CENTRAL	HOSPITAL REGIONAL	928
HMI- SAN LORENZO	CENTRAL	HOSPITAL MATERNO INFANTIL	667

Figura 10. Os cinco hospitais públicos que mais disponibilizam serviços de saúde.

Podemos ver na Figura 10 que o quinto hospital que mais realizou serviços se encontram na capital, Assunção, enquanto os quatro primeiros se encontram na região central. A diferença dos serviços também é grande, onde um hospital especializado na região central contém próximo ao dobro de serviços realizados ao de um hospital geral na capital.

5. Dados de Assunção.

Uma análise mais específica foi realizada, com o objetivo de retornar apenas informações que são referentes a capital do Paraguai, Assunção. O motivo dessa escolha se deve não só por Assunção ser a capital do Paraguai, mas também por ser uma das cidades mais populosas¹³ e, como veremos a seguir, é uma das cidades com maior concentração de hospitais do país. Na Tabela 1 têm-se informações gerais dos dados de saúde de Assunção.

Tabela 1. Dados gerais referentes a Assunção.

Descrição	Quantidade
Estabelecimentos de Saúde	69
Serviços de Saúde Disponibilizados	105
Quantidade de Produtos Disponibilizados	201

Dos estabelecimentos públicos de saúde pública que se encontra em Assunção, só se encontram estabelecimentos presentes em sete das oito categorias registradas. Não existem instalações do tipo *Centro de Salud* no município. As distribuições entre os tipos de estabelecimentos, segundo a disponibilidade de dados abertos, podem ser verificados na Figura 11.

¹³ <http://www.mspbs.gov.py/digies/wp-content/uploads/2012/01/IBS-Paraguay-2016.pdf> Acesso em 29 de Novembro de 2017

Podemos ver que a maior demanda dos estabelecimentos do país se encontra nas Unidades de Saúde Familiar, que chega a ser maior que todos os tipos de Hospitais e dos Centros Especializados presentes na capital.

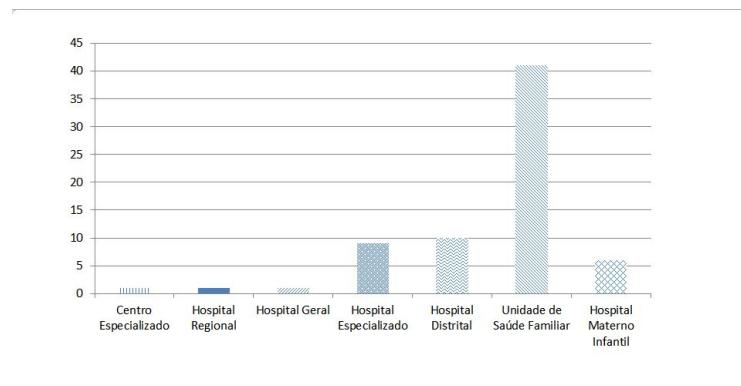


Figura 11. Distribuição dos estabelecimentos públicos de saúde em Assunção.

5.1 Análise Conjunta: Curitiba e Assunção.

Assim como o Paraguai,¹⁴ Curitiba disponibiliza os dados abertos através de portais de dados, como o E-Saúde¹⁴, e o portal da prefeitura. Os formatos disponibilizados são em formato CSV e XLSX, contendo o tempo de atualização dos dados, um histórico e um dicionário de dados para os arquivos.

Em relação aos estabelecimentos públicos de saúde, na Tabela 2 pode ser vista uma relação dos estabelecimentos públicos de saúde de Curitiba e Paraguai, relacionando a quantidade de estabelecimentos e suas classificações.

Tabela 2. Cruzamento de informações de estabelecimentos entre Curitiba e Paraguai.

Descrição	Paraguai	Curitiba
Quantidade de Estabelecimentos	174	213
Quantidade de Classificações	8	6

Podemos ver pelas informações contidas na Tabela 2 que apesar da proporção de estabelecimentos não ser próxima, a quantidade de categorias necessárias para classificar tais estabelecimentos é parecida. Na Tabela 3 podem-se visualizar uma generalização das categorias dos dois locais.

Tabela 3. Categorias de estabelecimentos de saúde de Curitiba e Paraguai.

Categorias do Paraguai	Quantidade	Categorias de Curitiba	Quantidade
Unidades de Saúde	102	Unidades de Saúde	126
Hospitais	68	Hospitais	75
Centros de Saúde	4	Unidades de Saúde - CAPS	12

¹⁴ <http://esaude.curitiba.pr.gov.br/PortalSaude/portal.do?formAction=init&v=2> Acesso em 04 de Dezembro de 2017.

Pela Tabela 3 é possível verificar que Curitiba contém uma quantidade maior de estabelecimentos de saúde. É importante verificar que em Curitiba existe uma categoria de unidade de saúde que se diferencia das demais. É a Unidade de Saúde – CAPS. O termo CAPS é dado pela abreviação de Centro de Atenção Psicossocial¹⁵. São serviços de saúde mental fornecidos pelo governo, e em Curitiba, dão atenção ao consumo de drogas e ao álcool.

A Figura 12 apresenta os mapas de calor de Curitiba e Paraguai, contendo os estabelecimentos públicos de saúde. Nesses mapas, as regiões com coloração mais forte indicam uma maior quantidade de estabelecimentos no local.

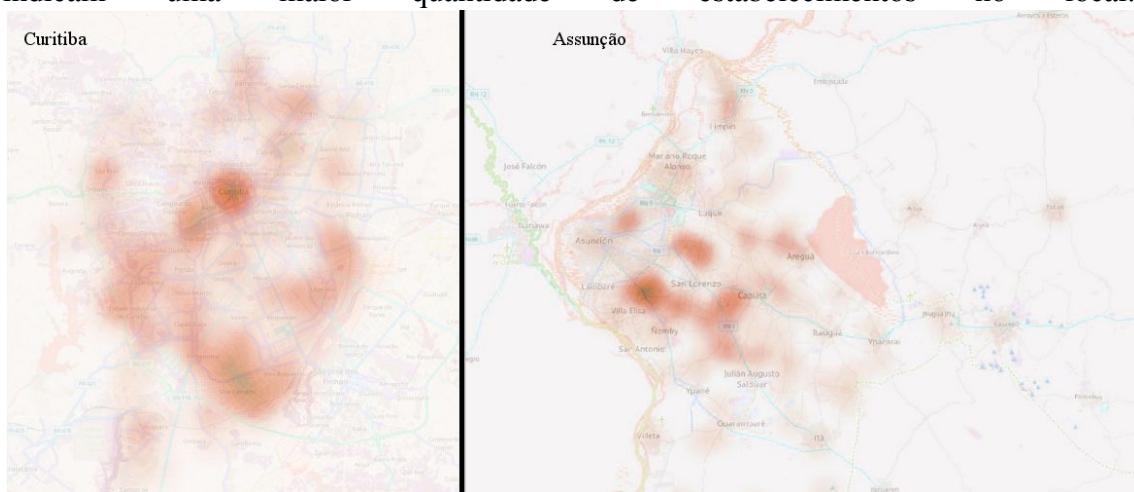


Figura 12. Mapa de calor dos estabelecimentos públicos de saúde de Curitiba e de Assunção visualizado através do QGIS.

Pode-se perceber na Figura 12 que a maior quantidade dos estabelecimentos tende a se concentrar na região central. Entretanto, os estabelecimentos de saúde de Curitiba estão mais distribuídos pela cidade em comparação com Assunção, onde algumas áreas mais distantes são desprivilegiadas, com uma menor quantidade destes estabelecimentos.

Dentre os desafios enfrentados podemos citar: 1) problemas de manutenção da publicação dos dados (diferença entre metadados e arquivos, o registro dos arquivos não é padronizado e não há informação sobre atualizações); 2) poucos arquivos contém informações geográficas (latitude e longitude); e 3) a correlação entre equipamentos de governo e os dados disponibilizados por cada entidade é diferenciada entre os países.

6. Conclusão.

Este artigo apresentou uma caracterização dos dados de saúde pública do Paraguai, e usando as informações referentes à Assunção para comparar com a cidade de Curitiba. As comparações foram realizadas para encontrar as diferenças na gestão da saúde pública e na forma como ela é, hoje, utilizada pela população.

Durante a realização desta pesquisa, encontraram-se diversos desafios em todas as etapas, desde a obtenção dos dados abertos por falta de manutenção dos portais de acesso, diferentes fontes para a disponibilização dos arquivos, e a falta de informações

¹⁵ <http://www.curitiba.pr.gov.br/servicos/cidadao/centro-de-atencao-psicossocial-alcool-e-drogas-servicos/680>
Acesso em 15 de Dezembro de 2017

nos metadados. Contudo, as informações obtidas e demonstradas neste artigo já podem ser utilizadas como embasamento para novas pesquisas mais aprofundadas, de maneira a melhorar o sistema de saúde público não só em Curitiba e no Paraguai, mas também de outras cidades.

Como trabalhos futuros podemos citar a análise de outras áreas, e a complementação e integração dos dados com outras fontes.

Agradecimentos: Prefeitura de Curitiba, IPPUC, Projeto EU-BR EUBra-BigSea (*MCTI/RNP 3rd Coordinated Call*).

Referências

- Aló, C. C. (2009) “Uma abordagem para transparência em processos organizacionais utilizando aspectos”. PhD thesis, PUC-Rio.
- Carvalho, L; Maia, C. (2016). Empreendedores cívicos e *Smart Cities*: práticas, motivações e geografias da inovação. Revista de Geografia e Ordenamento do Território (GOT), n.º 10, p. 95-112.
- Freitas, H.; Janissek-Muniz, R.; Moscarola, J. (2005). Modelo de formulário interativo para análise de dados qualitativos. Revista de Economia e Administração, São Paulo-SP, v. 4, nº 1, p. 27-48.
- Raghupathi and Raghupathi. (2014). “*Big data analytics in healthcare: promise and potential*”. Health Information Science and Systems. p. 1-9.
- Molloy, J.C. (2011). *The Open Knowledge Foundation: Open Data Means Better Science*. PLoS Biol vol 9, nº 12. p. 1-4.
- Manyika, James.; Chui, Michael.; Brown Brad.; Bughin, Jacques.; Dobbs, Richard.; Roxburgh, Charles.; Byers, H Angela.; (2011), *Big data: The next frontier for innovation, competition, and productivity*. McKinsey Global Institute. p. 1-13.
- Nakonetchnel, E. C.; Capiello, C. Kozievitch, N.P.; Vitalli, M.; Akbar, M. (2017). *Mobility Open Data: Use Case for Curitiba and New York*. XIII Escola Regional de Banco de Dados, p. 140-144.
- Pane, Juan.; Paciello, Julio.; Ojeda, Verena.; Valdez, Natalia; (2016). *Enabling dengue outbreak predictions based on open data*. Open Data Research Symposium. p. 1-16.
- Flores Junior, J. A. F.; Steffenello, L. C.; Winck, A. T. (2017). Mapeamento de Padrões de Acidentes de Trânsito com Vítimas Fatais a partir de Dados Públicos do Governo do Estado do Rio Grande do Sul. XIII Escola Regional de Banco de Dados, p. 91-94.

Análise de evolução de emissão de alvarás próximos a dois shoppings em Curitiba

Yuri Socher Bichibichi¹, Nádia P. Kozievitch¹, Renata A. M. Carvalho¹

¹Departamento Acadêmico de Informática
Universidade Tecnológica Federal do Paraná (UTFPR)
Avenida Sete de Setembro – 3165 – 80.230-901 – Curitiba – PR – Brasil

{yuribichibichi, rcarvalho}@alunos.utfpr.edu.br, nadiap@utfpr.edu.br

Abstract. In the context of smart cities, the information of businesses licenses has the potential to discriminate economics characteristics of the observed urban environment. This work performs an quantitative and entropic analysis of businesses licenses in an area around establishments considered as key venues (i.e., businesses that highly influence a neighborhood). Given the analysis is possible to notice that after the inauguration of each of the 2 studied shoppings, the number of licenses reaches a peak followed by a dramatic drop. From entropy variation is observable that in both shoppings there are a bigger dispersion compared to the rest of the neighborhood.

Resumo. No contexto de cidades inteligentes, a informação de alvarás tem potencial de discriminar as características econômicas do meio urbano observado. Neste trabalho é feita uma análise quantitativa e entrópica sobre a emissão de alvarás próximos a estabelecimentos considerados chave (isto é, estabelecimentos que afetam fortemente sua vizinhança). Dada a análise é possível notar que após a inauguração de cada um dos 2 shopping centers estudados o número de alvarás abertos alcança um pico seguido por uma queda vertiginosa. A partir da variação da entropia também é notável que há em ambos os shoppings uma maior dispersão nos tipos de alvarás que o restante do bairro.

1. Introdução

A análise de dados provenientes de cidades inteligentes possibilita a descoberta de fatores sociais, econômicos e culturais de maneira mais barata que pesquisas conduzidas tradicionalmente (ex. censos e questionários). Além disso, esses dados são mais escaláveis e, por poderem ser coletados mais rapidamente, refletem mais fidedignamente a situação atual do cenário analisado [Silva and Loureiro 2016]. Assim, o desafio é descobrir como explorar computacionalmente informações de dados urbanos na geração de conhecimento aplicado à sociedade, incluindo subsídios para definição estratégica de políticas públicas.

Os estabelecimentos comerciais de uma vizinhança possuem grande impacto econômico e social em seu contexto local, além de normalmente valorizar os terrenos inseridos dentro dos mesmos limites [Carr et al. 2003]. Dessa forma, estudar o desenvolvimento de uma área comercial oferece possíveis indicadores que ajudariam a identificar potenciais novas áreas comerciais.

Este trabalho tem como objetivo a exploração e análise da base de dados que contém a relação de alvarás para liberação de atividades comerciais e edificações dentro do município de Curitiba buscando um padrão de expansão de estabelecimento regularizados ao redor de estabelecimentos tipicamente populares (i.e., *shopping centers*). Os estabelecimentos cuja região é estudada são o Shopping Pátio Batel e o Shopping Crystal Plaza. Ambos foram escolhidos por estarem no bairro Batel, o qual está numa região central da cidade e no qual os dados são relativamente melhor documentados.

O restante do artigo está dividido da seguinte maneira: os trabalhos relacionados são abordados na Seção 2; a origem e as particularidades dos dados analisados são apresentados na Seção 3; um breve histórico com as informações relevantes do cenário para a compreensão do trabalho são descritos na Seção 4; o desenvolvimento se dá na Seção 5 e; na Seção 6, finalmente, é apresentada a conclusão.

2. Trabalhos Relacionados

A investigação foi realizada por meio de pesquisa bibliográfica, revisão bibliográfica e conceitualização dos diversos temas que envolvem o assunto principal.

2.1. Dados urbanos

Metade da população mundial já vive em cidades grandes e previsões indicam que em 2050 esta fatia será de 70% [Rassia et al. 2014]. Este é um cenário pior que o atual onde já há problemas envolvendo consumo de energia, engarrafamentos, tratamento de resíduos, emprego, inclusão social, saúde, educação ambiental e proteção ambiental. Por exemplo, segundo *World Energy Outlook*¹, 76% das emissões de CO₂ são geradas nas cidades e engarrafamentos representam um prejuízo de 1% do PIB Europeu [Cardin et al. 2015]. Esse crescimento proporciona a possibilidade de melhoria de vida da população, embora sua concretização dependa também de medidas governamentais, mercado e de investimentos na infraestrutura [Turok and McGranahan 2013].

Dados gerados na localidade urbana podem ser utilizados para auxiliar a tomada de decisões de governos públicos e corporações privadas. Em 2012, a cidade de Chicago passou por uma reforma que reduziu o número de categorias de licenças de estabelecimentos para agilizar e diminuir o peso da burocracia nos pequenos negócios após a análise de dados empíricos da cidade (ex., quantidade de tipos de licenças, número de inspeções anuais em um local, porcentagem de restaurantes que passaram a primeira inspeção sanitária) [Team 2015].

2.2. Cidades Inteligentes e Computação Urbana

Conceitos como cidades geo-inteligentes, localidades urbanas que permitem a exploração de dados sensoriais para sua melhoria, foram definidos em conjunto a uma arquitetura de sistema que visa aumentar a facilidade no tratamento de tarefas de geoprocessamento por [Morales and Garcia 2015]. O sistema proposto diminui a complexidade do geoprocessamento ao focar na detecção e execução de eventos, diminuindo o volume de dados.

Segundo a Cisco existem 3 etapas² para implantação de um modelo de cidade inteligente:

¹Disponível em: <http://www.worldenergyoutlook.org/>. Acessado em: 26/11/2017.

²Disponível em: https://www.cisco.com/c/dam/en_us/solutions/industries/docs/scc/ie_citizen_svcs_white_paper_idc_2013.pdf. Acessado em: 26/11/2017.

- Primeiramente expandir o acesso à banda larga em toda cidade.
- Em segundo, construir serviços sobre esta estrutura, por exemplo educação, saúde, turismo, etc (varia de acordo com a necessidade da cidade).
- Finalmente a cidade deve fortalecer a interação dos cidadãos nestas plataformas.

A participação dos cidadãos em plataformas online tais quais as Redes Sociais Baseadas em Localização oferece a oportunidade de análise de uma imensa quantidade de dados quase em tempo real da qual é possível extrair informações econômicas, sociais, culturais, dinâmicas de movimentação, etc [Silva and Loureiro 2016].

2.3. Geoprocessamento: aplicações

O estudo da dinâmica dos meios de transportes da cidade de Belo Horizonte foi realizado com o auxílio de um Sistema de Informação Geográfica (SIG) que incluía informações como malha de eixos, arruamento, nós de cruzamento, entre outras [Zuppo et al. 1996]. O estudo foca na modelagem dos dados como os agentes e pontos espaciais do transporte coletivo e a forma que ocorre a circulação viária e sinalização da cidade.

Em um estudo realizado usando dados do estado de São Paulo foi verificada uma correlação entre problemas respiratórios e queima de cana-de-açúcar [Lopes and Ribeiro 2006]. O trabalho foi feito correlacionando dados de casos hospitalares no SUS e dados geoespaciais disponibilizados pelo INPE. Os dados foram manipulados na ferramenta MS-Access 2000³ e depois foram visualizados na ferramenta MapInfo⁴. Apesar de antiga, a ferramenta MapInfo, semelhante ao QGIS, ainda está em manutenção. No entanto é uma ferramenta de código fechado não-gratuita.

Em um trabalho realizado sobre os dados de transporte de Curitiba foi identificado que o número de veículos cresceu mais rápido que a infraestrutura da cidade [Vila et al. 2016]. No artigo foram usados os dados do Instituto de Planejamento de Curitiba (IPPUC⁵). Para a visualização foram usados os dados do Google Maps, OpenStreetMaps e PostGIS.

Em outro estudo utilizando dados de Curitiba são apresentados os desafios relacionados aos dados de redutores de velocidade no transporte público [Costa et al. 2017]. Foram utilizados dados do IPPUC, Prefeitura Municipal de Curitiba (PMC), Secretaria Municipal de Trânsito (SETRAN) e Companhia de Urbanização de Curitiba (URBS). Para a visualização dos mesmos foram usadas as ferramentas Google Map e QGIS.

Em um estudo também realizado em Curitiba a atividade econômica foi analisada através da entropia de Shannon ([Shannon 1948]) sobre os bairros Centro, Batel e Tatuquara [Rosa et al. 2016], concluindo que ela tende a diminuir quando há um grande número de alvarás, i.e., os alvarás tendem a dispersar. O trabalho foi realizado utilizando os dados abertos da Prefeitura de Curitiba e PostGIS. Para a visualização foram usados o QGIS, Google Maps e OpenStreetMaps. Com exceção dos do Google Maps, estas ferramentas são as mesmas utilizadas neste artigo.

Diferentemente dos trabalhos apontados, neste artigo é realizada uma análise dos alvarás ao redor de shoppings. Uma comparação dos trabalhos apresentados pode ser visualizada na Tabela 1.

³Disponível em: <http://office.microsoft.com/access> . Acessado em: 26/11/2017.

⁴Disponível em: <http://www.mapinfo.com/> . Acessado em: 26/11/2017

⁵Disponível em: <http://www.ippuc.org.br/> . Acessado em: 08/12/2017.

Table 1. Comparação entre o artigo atual e trabalhos relacionados.

Artigo	Local	Ferramentas	Trabalho feito
[Vu et al. 2013]	Haiphong - Vietna	VISUM traffic model	identificação de correlação entre tráfego e poluição atmosférica
[Lopes and Ribeiro 2006]	São Paulo (estado)	MS-Access 2000, MapInfo	relação entre problemas respiratórios e queima de cana-de-açúcar
[Vila et al. 2016]	Curitiba	Google Maps, Open-StreetMaps, PostGIS	identificação de que o número de veículos cresceu mais rápido que a estrutura da cidade
[Costa et al. 2017]	Curitiba	Google Maps, QGIS	descrição dos desafios relacionados aos dados de redutores de velocidade no transporte público
[Rosa et al. 2016]	bairros Centro, Batel e Tatuquara de Curitiba	Google Maps, Open-StreetMaps, PostGIS	análise da entropia de shannon sobre a atividade econômica dos alvarás
(artigo atual)	bairro Batel em Curitiba	Google Maps, Open-StreetMaps, PostGIS	análise do impacto de um shopping nos alvarás ao redor (análise exploratória e entropia de shannon)

3. Dados e ferramentas

Os dados utilizados na análise são a relação de alvarás para liberação de atividades comercias e edificações dentro do município de Curitiba. Estes dados foram obtidos através do portal da Prefeitura de Curitiba⁶. Os atributos disponibilizados estão representados na Tabela 2.

Dados geográficos, apesar de complexos [Bezerra and Kaster 2017], têm sua análise viabilizada por ferramentas das quais a escolhida para visualização dos dados de maneira gráfica e auxílio na interpretação das informações obtidas foi o QGIS 2.14⁷. Este foi utilizado neste trabalho por tratar-se de um software gratuito compatível com PostgreSQL 9.1⁸

4. Cenário

Nesta seção são apresentados os cenários usados na análise: dois shoppings localizados no bairro Batel de Curitiba.

⁶Disponível em: <http://www.curitiba.pr.gov.br/dadosabertos/consulta/?grupo=2>. Acessado em: 26/11/2017.

⁷Disponível em: <http://qgis.org/ja/site/>. Acessado em: 26/11/2017.

⁸Disponível em: <https://www.postgresql.org/> e PostGIS 2.1⁹, o qual adiciona funções espaciais pelas quais é possível manipular geometrias e determinar relações espaciais [Schneider et al. 2017].

Para a renderização dos *boxplots* foi utilizada a linguagem R¹⁰.

Table 2. Atributos disponíveis na tabela de alvarás.

Atributo	Descrição	Atributo	Descrição
NOME_EMPRESARIAL	nome da empresa	CEP	CEP do endereço
NUMERO_DO_ALVARA	número da licença do alvará	DATA_EMISSAO	data de emissão da licença do alvará
DATA_EXPIRACAO	data de expiração da licença do alvará	UNIDADE	identificação da unidade
ATIVIDADE_SECUNDARIA1	descrição da atividade secundaria 01	ENDERECO	rua de endereço
ATIVIDADE_SECUNDARIA2	descrição da atividade secundaria 02	NUMERO	número predial
ATIVIDADE_PRINCIPAL	descrição da atividade principal	ANDAR	identificação do andar
COMPLEMENTO	complemento do endereço	BAIRRO	bairro do endereço
INICIO_ATIVIDADE	data de início da atividade		

4.1. Shopping Pátio Batel

Começando a construção em 2008 e tendo o lançamento comercial em 2011, o Shopping Pátio Batel tem como público alvo as classes A e B [Gazeta do Povo (2008)] e está localizado na Avenida do Batel. Sua inauguração ocorreu em setembro de 2013 com cerca de 200 lojas e expectativa mensal de 1 milhão de visitantes [Bem Paraná (2013)].

4.2. Shopping Crystal Plaza

Inaugurado em 1996, o Shopping Crystal Plaza é uns dos shopping mais antigos de Curitiba e o primeiro voltado para o público de classe A [Gazeta do Povo (2016)]. Atualmente é formado por mais de 150 lojas e duas vezes por ano sedia o evento de moda Crystal Fashion [Hotel Nikko]. Está localizado à 1 km do Shopping Pátio Batel, para o qual faz concorrência direta.

5. Desenvolvimento

Neste estudo é realizada uma análise sobre a quantidade de alvarás e dispersão da atividade principal. As regiões analisadas estão indicadas na Figura 1. Os círculos maiores em azul e vermelho demarcam regiões em torno dos shoppings de, respectivamente, 100 e 20 metros de raio. Os demais círculos representam alvarás que surgiram nestas regiões.

O número de alvarás que surgiu anualmente nas regiões analisadas pode ser visto nas Figuras 2 e 3 . Em laranja estão contabilizados todos os alvarás que surgiram em torno do local estudado (100 metros de raio) enquanto que em azul é retirado a região “do próprio local estudado” (20 metros de raio) - esta região tem forma de donut/roquinha na Figura 1. Observando a data de inauguração (respectivamente, em 2013 e 1996) é possível notar o impacto que ambos os shoppings causaram.

A Figura 4 indica novos alvarás expedidos anualmente no bairro Batel como um todo. Em laranja estão todos os alvarás enquanto que em azul são retirados os alvarás



Figure 1. Regiões analisadas.

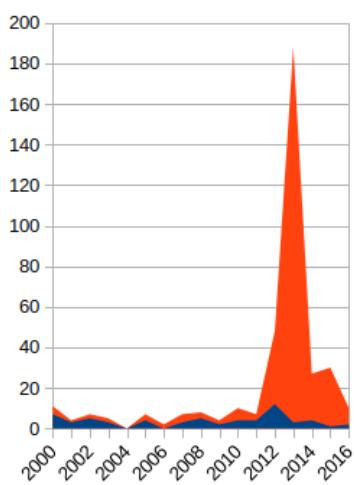


Figure 2. Novos alvarás por ano na região do Shopping Pátio Batel.

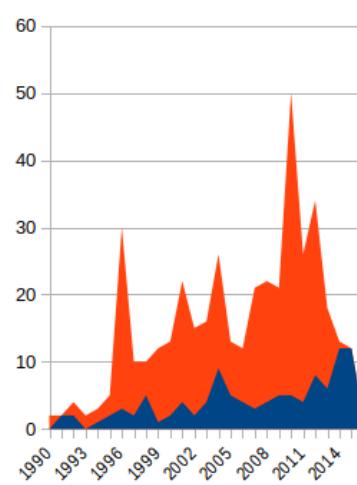


Figure 3. Novos alvarás por ano na região do Shopping Crystal Plaza.

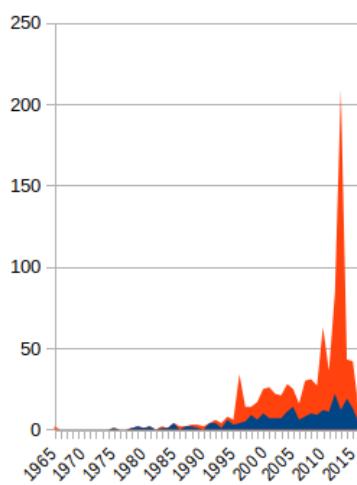


Figure 4. Novos alvarás por ano no bairro Batel.

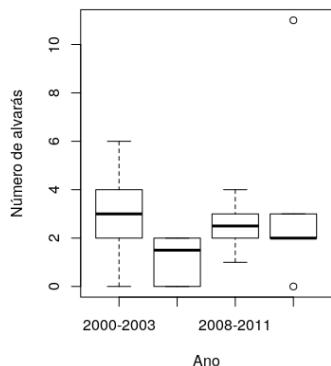


Figure 5. Análise bivariável de “novos alvarás” × “tempo” na região do Shopping Pátio Batel.

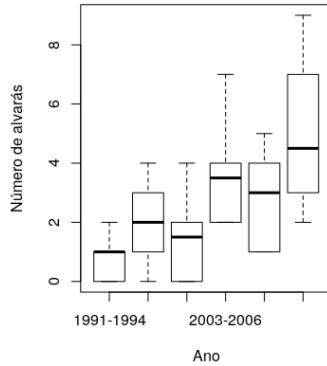


Figure 6. Análise bivariável de “novos alvarás” × “tempo” na região do Shopping Crystal Plaza.

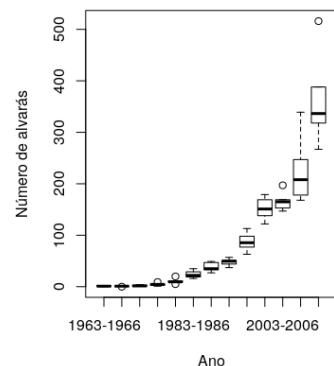


Figure 7. Análise bivariável de “novos alvarás” × “tempo” no bairro Batel.

das regiões próximas dos lugares estudados (20 metros de raio). Através destes dados é possível notar o tamanho do impacto de ambos os shoppings.

O impacto de ambos os shoppings em novos alvarás é discriminado nas Figuras 5 e 6. Para detectar *outliers* (valores fora da curva) é usado análise bivariável com *boxplots*. A região analisada está de 20 a 100 metros dos locais estudados (região em forma de rosquinha). Os dados estão organizados em grupos de 8 meses. Cada *boxplot* recebe estas informações referentes a um período de 4 anos (6 valores de 8 meses cada).

Na região do Shopping Pátio Batel, no período 2012-2015, aparecem dois *outliers*, indicando que houve um crescimento e decrescimento anômalo (Figura 5). Os dados referentes ao Shopping Crystal Plaza expressam picos nos períodos 1995-1998, 2003-2006 e 2011-2014, mas nenhum *outlier*. Na Figura 7 pode-se observar o mesmo *boxplot* das Figuras 5 e 6 aplicado sobre todo o bairro Batel. Nota-se que no período de 2011-2014 houve um crescimento anômalo.

Neste artigo o termo dispersão é definido como variação, por exemplo, o cenário em que há um alvará de cada tipo representa dispersão máxima. Concentração é definida no sentido contrário da dispersão, então, por exemplo, o cenário em que todos os alvarás são do mesmo tipo representa concentração máxima. Para a dispersão dos tipos de alvarás é utilizada a entropia de Shannon [Shannon 1948], calculada conforme a equação:

$$H_S = - \sum_{i=1}^N p_i \log_2 p_i \quad (1)$$

no qual $P = \{p_1, p_2, \dots, p_N\}$ (probabilidades), $0 \leq p_i \leq 1$ e $\sum_{i=1}^N p_i = 1$. A entropia mínima $H_S^{min} = 0$ indica concentração máxima enquanto $H_S^{max} = \log_2 N$ indica dispersão máxima. Para que seja possível comparar entropias com diferentes valores para N é usada $H_{norm} = \frac{H_S}{H_S^{max}}$, onde $0 \leq H_{norm} \leq 1$.

Para o cálculo da dispersão da atividade principal, $p_i = \frac{C_i}{C}$, no qual C_i representa o número de ocorrências para cada atividade principal e C , o número total de alvarás. Os resultados destes cálculos estão apresentados na Tabela 3. Observa-se que os alvarás na região de cada shopping são aqueles distantes no máximo 20 metros dos mesmos.

Table 3. Entropia em diferentes locais e períodos.

Local	Período	Entropia
Shopping Crystal Plaza	1996 - 2006	0,757
Shopping Pátio Batel	2012 - 2017	0,706
Batel	1990 - 1995	0,822
Batel (longe dos shoppings)	1996 - 2005	0,849
Batel (longe dos shoppings)	2006 - 2011	0,865
Batel (longe dos shoppings)	2012 - 2015	0,853

Table 4. Top 10 Atividades mais comuns no bairro Batel.

Atividade (2000-2002)	Qnt.	Atividade (2010-2012)	Qnt.
Restaurantes e similares	36	Atividades de estética e outros serviços de cuidados com a beleza	91
Atividades de consultoria em gestão empresarial, exceto consultoria técnica específica	35	Cabeleireiros	40
Atividade médica ambulatorial restrita a consultas	27	Atividades de consultoria em gestão empresarial, exceto consultoria técnica específica	37
Comércio varejista de artigos do vestuário e acessórios	21	Incorporação de empreendimentos imobiliários	34
Serviços advocatícios	17	Serviços combinados de escritório e apoio administrativo	32
Atividade médica ambulatorial com recursos para realização de procedimentos cirúrgicos	14	Holdings de instituições não-financeiras	32
Corretores e agentes de seguros, de planos de previdência complementar e de saúde	13	Comércio varejista de artigos do vestuário e acessórios	31
Sedes de empresas e unidades administrativas locais	13	Atividade médica ambulatorial restrita a consultas	30
Comércio varejista especializado de equipamentos e suprimentos de informática	12	Estacionamento de veículos	28
Representantes comerciais e agentes do comércio de mercadorias em geral não especializado	12	Restaurantes e similares	26

Analizando os resultados é possível notar que os alvarás dos shoppings tendem a ter uma diversidade maior que no restante do bairro, no qual os tipos de alvarás estão tornando-se mais concentrados. Na Tabela 4 estão as atividades mais populares em diferentes períodos.

6. Conclusão

Neste trabalho são exploradas informações geográficas dos alvarás do bairro Batel na cidade de Curitiba com o intuito de analisar a relação entre a abertura de certos estabelecimentos chaves e o crescimento de alvarás em uma vizinhança. Assim, é possível constatar a ocorrência de picos de aumento de estabelecimentos na área logo após a inauguração dos shoppings e um contraste entre a entropia dos alvarás dos shoppings com o restante do bairro.

A partir deste trabalho existem diversas outras possíveis análises, por exemplo: um estudo que envolva a base de dados "Disque Economia", a qual pode ser usada para estimar a inflação dos preços de produtos básicos na cidade de Curitiba; uma pesquisa sobre a criminalidade nos locais citados (principalmente na periferia dos shoppings, onde uma possível queda no número de bares possa aumentar a criminalidade no período noturno, por exemplo); uma análise que estude o impacto que shoppings causam na entropia geoespacial (provavelmente uma grande concentração de alvarás cause congestionamento, por exemplo); etc.

Agradecimentos

Os autores agradecem a Prefeitura Municipal de Curitiba, IPPUC e ao projeto EU-BR EUBra-BigSea (*MCTI/RNP 3rd Coordinated Call*).

References

- [Bem Paraná (2013)] Bem Paraná (2013). Prefeitura autoriza inauguração do Shopping Pátio Batel. <https://www.bemparana.com.br/noticia/276092/prefeitura-autoriza-inauguracao-do-shopping-patio-batel>. Acesso em: 25 Out. 2017.
- [Bezerra and Kaster 2017] Bezerra, V. H. and Kaster, D. d. S. (2017). Inclusão de técnicas de interpolação de pontos em algoritmos de descoberta on-line do padrão floc. In *Anais da XIII Escola Regional de Banco de Dados*, pages 57–66.
- [Cardin et al. 2015] Cardin, M.-A., Krob, D., Lui, P. C., Tan, Y. H., and Wood, K. (2015). *Complex systems design & management Asia*. Springer.
- [Carr et al. 2003] Carr, D., Education, D. R. E., Lawson, J., Lawson, J., and Schultz, J. (2003). *Mastering Real Estate Appraisal*. Kaplan Financial Series. Kaplan.
- [Costa et al. 2017] Costa, G., Kozievitch, N. P., Fonseca1, K., Gadda, T., and Berardi, R. (2017). Integração de dados de redutores de velocidade no transporte público de curitiba. In *Anais da XIII Escola Regional de Banco de Dados*, pages 152–156.
- [Gazeta do Povo (2008)] Gazeta do Povo (2008). Pátio Batel ficará pronto em 2012. <http://www.gazetadopovo.com.br/economia/patio-batel-ficara-pronto-em-2012-b6wp0wlnzv343ph3n0k5lzivi>. Acesso em: 25 Out. 2017.
- [Gazeta do Povo (2016)] Gazeta do Povo (2016). Shopping Crystal abandona ‘DNA de luxo’ e terá serviços mais populares. <http://www.gazetadopovo.com.br/economia/shopping-crystal-abandona-dna-de-luxo-e-tera-servicos-mais-populares-54dajr0han0eluv28e1a4vo40>. Acesso em: 29 Dez. 2018.

- [Hotel Nikko] Hotel Nikko. Shopping Crystal Plaza. <http://www.hotelnikko.com.br/dicas/shopping-crystal-plaza-em-curitiba>. Acesso em: 29 Dez. 2018.
- [Lopes and Ribeiro 2006] Lopes, F. S. and Ribeiro, H. (2006). Mapeamento de internações hospitalares por problemas respiratórios e possíveis associações à exposição humana aos produtos da queima da palha de cana-de-açúcar no estado de são paulo. *Rev Bras Epidemiol*, 9(2):215–25.
- [Morales and Garcia 2015] Morales, J. and Garcia, M. (2015). Geosmart cities: Event-driven geoprocessing as enabler of smart cities. In *2015 IEEE First International Smart Cities Conference (ISC2)*, pages 1–6.
- [Rassia et al. 2014] Rassia, S. T., Pardalos, P. M., et al. (2014). Cities for smart environmental and energy futures. *Springer-Verlag Berlin Heidelberg. doi*, 10:978–3.
- [Rosa et al. 2016] Rosa, J., Silva, T. H., Kozievitch, N. P., and Ziviani, A. (2016). Ciência de dados: Explorando três décadas de evolução da atividade econômica em curitiba. In *Anais da XII Escola Regional de Banco de Dados*, pages 139–142.
- [Schneider et al. 2017] Schneider, V. E., Graciolli, O. D., Graziottin, R. H., Spiandorello, R. C., Hoffmann, G. V., and Giordani, M. A. P. (2017). Consulta de dados espaciais em um sistema de informacoes de uma bacia hidrografica. In *Anais da XIII Escola Regional de Banco de Dados*, pages 87–90.
- [Shannon 1948] Shannon, C. E. (1948). A mathematical theory of communication, part i, part ii. *Bell Syst. Tech. J.*, 27:623–656.
- [Silva and Loureiro 2016] Silva, T. H. and Loureiro, A. A. (2016). Users in the urban sensing process: Challenges and research opportunities. In *Pervasive Computing: Next Generation Platforms for Intelligent Data Collection*, pages 45–95. Academic Press.
- [Team 2015] Team, R. R. (2015). Case study: Chicago licensing and permitting reform. *Data-smart city solutions*.
- [Turok and McGranahan 2013] Turok, I. and McGranahan, G. (2013). Urbanization and economic growth: the arguments and evidence for africa and asia. *Environment and Urbanization*, 25(2):465–482.
- [Vila et al. 2016] Vila, J. J. R., Kozievitch, N. P., Gadda, T. M., Fonseca, K., Rosa, M. O., Gomes-Jr, L. C., and Akbar, M. (2016). Urban mobility challenges—an exploratory analysis of public transportation data in curitiba. *Revista de Informática Aplicada*, 12(1).
- [Vu et al. 2013] Vu, V.-H., Le, X.-Q., Pham, N.-H., and Hens, L. (2013). Application of gis and modelling in health risk assessment for urban road mobility. *Environmental Science and Pollution Research*, 20(8):5138–5149.
- [Zuppo et al. 1996] Zuppo, C. A., Davis Jr, C. A., Meirelles, A. A., and do Município, P.-P. d. D. (1996). Geoprocessamento no sistema de transporte e trânsito de belo horizonte. *Anais II GIS Brasil*, pages 376–387.

Comparação entre MySQL e Neo4J para o Acesso a Dados Complexos Usando Linguagens Declarativas

Émerson P. Homrich¹, Sergio L. S. Mergen¹

¹Universidade Federal de Santa Maria (UFSM)
Santa Maria, RS – Brasil

{ehomrich, mergen}@inf.ufsm.br

Abstract. *Graph databases are a type of NoSQL databases focused on highly connected data and dynamic relationships, characteristics that make them plausible for applications such as social networks and preferences systems. With data becoming increasingly sparse and semi-structured, it's questioned whether graph databases already have maturity to be more advantageous than relational databases to access complex data using queries with a declarative syntax. The purpose of this work is to compare two database leaders in graph and relational technology (Neo4J and MySQL, respectively) with respect to their performance in accessing complex data, using only the resources of their declarative query languages.*

Resumo. *Os bancos de dados de grafo são um tipo de bancos de dados NoSQL voltados a dados altamente conectados e com relacionamentos dinâmicos, características plausíveis para aplicações como redes sociais e sistemas de preferências. Com os dados se tornando cada vez mais esparsos e semi-estruturados, questiona-se se bancos de dados de grafo já possuem maturidade para serem mais vantajosos do que bancos de dados relacionais para acessar dados complexos usando consultas com uma sintaxe declarativa. O objetivo deste trabalho é comparar dois bancos de dados líderes nas tecnologias de grafo e relacional (Neo4J e MySQL, respectivamente) em relação ao seu desempenho no acesso a dados complexos, usando apenas recursos de suas linguagens declarativas.*

1. Introdução

Bancos de dados de grafo são um tipo de bancos de dados NoSQL voltado para trabalho com dados altamente conectados e com relacionamentos dinâmicos em grandes volumes. Tais características os tornam opções plausíveis para aplicações como redes sociais, sistemas de recomendações e outros tipos de dados complexos [Sadalage and Fowler 2012].

Geralmente construídos para uso com sistemas transacionais, são otimizados para garantir integridade transacional e disponibilidade de operação, assim como os bancos de dados relacionais [Robinson et al. 2013]. Outra característica em comum é o uso de linguagens de consulta declarativas (ex. SQL e Cypher). Muito embora existam interfaces de acesso mais procedurais, as linguagens declarativas aumentam a legibilidade do código, levando a maior manutenibilidade e produtividade.

O crescimento de dados guiados pelo usuário aumentou o volume e o tipo de dados gerados. Em paralelo a esse crescimento, os dados também estão se tornando cada vez

mais esparsos e semi-estruturados [Tiwari 2011]. Com isso, questiona-se se bancos de dados de grafo já possuem maturidade suficiente para serem mais vantajosos em relação à tecnologia dos bancos de dados relacionais para trabalhar com dados complexos usando apenas comandos da linguagem de consulta declarativa.

Nesse sentido, este trabalho tem o intuito de comparar bancos de dados relacionais e bancos de dados não relacionais orientados a grafo, com foco no MySQL e no Neo4j, analisando características e o desempenho desses SGBDs no acesso a dados complexos. A escolha pelos representantes de cada tipo de SGBD foi tomada considerando o ranking de bancos de dados do site *DB-Engines*¹. O Neo4j ocupa a 21^a posição no ranking geral, envolvendo todos os SGBDs, e a 1^a posição no ranking de bancos de dados de grafo. O MySQL, por sua vez, é o segundo mais popular tanto entre sua categoria como nos bancos de dados em geral.

Este trabalho está organizado da seguinte forma: a seção 2 apresenta trabalhos correlatos. A seção 3 apresenta uma comparação entre características dos modelos relacional e de grafos. Na seção 4 são apresentados os testes realizados e os resultados obtidos. Por fim, a seção 5 trata das conclusões e considerações finais.

2. Trabalhos Relacionados

Considera-se como trabalhos relacionados aqueles que analisam o desempenho de operações nos SGBDs MySQL e Neo4j com abordagens semelhantes à proposta deste trabalho. Neste contexto, são discutidos alguns dos trabalhos já realizados.

Em [Batra and Tyagi 2012], é realizado um experimento que consiste em um conjunto de consultas explorando relacionamentos. O domínio de dados é composto por usuários com relações de amizade, e cada usuário pode ter filmes favoritos estrelados por atores. Um conjunto de consultas complexas foi definido, como a busca pelos protagonista dos filmes favoritos dos amigos de um usuário. As consultas foram realizadas sobre bases com 100 e 500 usuários, para verificar o comportamento dos SGBDs com o crescimento do volume de dados. Os resultados mostram a prevalência do Neo4j nos testes, com tempos de execução consideravelmente menores que os do MySQL e pouca variação com o aumento da base de dados.

De modo semelhante, [Medhi and Baruah 2017] apresenta um experimento de desempenho. O domínio aborda jogadores, times e partidas de críquete. As consultas foram realizadas recuperando 100, 300 e 400 objetos. O desempenho do Neo4j prevalece nos testes, sendo duas vezes mais rápido nos bancos de dados com 100 objetos e até dez vezes mais rápido quando o número de objetos sobe para 400.

Em [Vicknair et al. 2010], são utilizados grafos acíclicos direcionados gerados aleatoriamente. No experimento, utilizou 4 grafos, de 1.000, 5.000, 10.000 e 100.000 nós. Os nós de cada grupo guardam um atributo inteiro aleatório ou cadeia de caracteres de 8KB ou 32KB. Constatou-se que o tamanho dos grafos no Neo4j, em MB, pode ser até duas vezes maior que os grafos no MySQL. Os tempos de execução das consultas estruturais foram menores no MySQL com os grafos menores e/ou de nós com atributos inteiros, mas consideravelmente mais rápidos no Neo4j quando realizadas sobre os grafos maiores ou de nós com atributos de caracteres. Nas consultas de dados, o MySQL apre-

¹<https://db-engines.com>

sentou desempenho muito superior ao Neo4j utilizando campos numéricos. O contrário ocorreu em consultas com parâmetros de texto. O ambiente de testes foi uma máquina com 4GB RAM e CPU Intel(R) Core(TM) 2 Duo com 3GHz de frequência de operação.

Com exceção de [Vicknair et al. 2010], os demais trabalhos utilizaram bases de dados bastante reduzidas durante os experimentos. Os trabalhos também não realizaram as consultas de forma padronizada, isto é, não utilizaram um mesmo meio para realizá-las. [Batra and Tyagi 2012] e [Medhi and Baruah 2017] realizam as consultas do MySQL através de *scripts* em linguagem PHP e as consultas do Neo4j diretamente em linguagem Cypher. A falta de padrão no método pode ter exposto as consultas ao MySQL à influência da linguagem.

3. O Modelo Relacional e o Modelo de Grafos

A eficiência no acesso aos dados pode sofrer forte influência da forma como os dados são modelados, bem como das interfaces de acesso disponibilizadas. Essa seção resume as principais diferenças entre o modelo relacional e o modelo orientado a grafos nesses dois aspectos.

No modelo relacional, o banco de dados é um conjunto de relações [Ramakrishnan and Gehrke 2008]. Já nos bancos de dados de grafo, o modelo baseia-se na teoria dos grafos. Os dados são representados como um conjunto de vértices (nós) conectados por arestas (relacionamentos), sendo que ambos podem conter propriedades (atributos) no formato de chave-valor. Os nós representam objetos e as arestas representam a forma como esses objetos se relacionam, formando caminhos [Robinson et al. 2013]. Nós podem ser agrupados em rótulos e relacionamentos podem ter nomes e devem possuir uma direção.

Diferente do modelo relacional que é regrado pelo esquema (e que por vezes é considerado rígido por essa característica), o modelo de grafos possui formato livre. Nós de um mesmo rótulo podem ter estruturas diferentes. Outra diferença é o formato dos relacionamentos. No modelo relacional, os relacionamentos são feitos através de campos de chave estrangeira, e podem exigir a criação de tabelas intermediárias. No modelo de grafos, os relacionamentos são tão importantes quanto os nós em si e são armazenados como relacionamentos de fato [Sadlage and Fowler 2012]. Os nomes dos relacionamentos adicionam clareza semântica à estruturação dos nós [Robinson et al. 2013].

No que tange à interface de acesso, ambos os SGBDs adotam linguagens de consultas declarativas. Os bancos de dados relacionais utilizam a linguagem *Structured Query Language* (SQL) [Silberschatz et al. 2010]. O Neo4j utiliza a linguagem Cypher², inicialmente desenvolvida para uso exclusivo e posteriormente adotada por outros bancos de dados de grafo através do projeto openCypher. É uma linguagem compacta e expressiva, projetada para ser de fácil leitura, e seu uso é de acordo com a forma intuitiva na qual os grafos geralmente são feitos em diagramas [Robinson et al. 2013].

Tanto nos bancos de dados relacionais como no Neo4j, a execução de consultas constitui-se por etapas. As consultas SQL passam por verificação e validação, que analisa a sintaxe da consulta e identifica as tabelas envolvidas, bem como os atributos solicitados

²Existem outras linguagens de consulta amplamente suportadas pelos bancos de dados de grafo (inclusive pelo Neo4j), como a *SPARQL Protocol and RDF Query Language* (SPARQL) e a Gremlin.

ou utilizados para filtragens. O SGBD traduz a consulta para uma estrutura como uma árvore de consulta, e define uma estratégia de execução, que é decomposta em blocos. A escolha da estratégia de execução é a complexa tarefa de otimização de consulta, que nem sempre será a melhor, e sim a mais razoável [Elmasri and Navathe 2011].

Na linguagem Cypher, as consultas também são executadas através de planos de execução. Cada consulta é dividida em porções menores chamadas de operadores³. O conteúdo da consulta é tokenizado e avaliado semanticamente, formando uma árvore sintática abstrata após ser otimizado e normalizado. Um grafo de consulta é criado, e então são definidos planos lógicos. A seletividade dos rótulos e índices, bem como a cardinalidade de registros são definidos baseados em estatísticas. A partir dos planos lógicos, um algoritmo seleciona o plano menos custoso e otimiza-o para gerar o plano de execução⁴.

Consultas a dados conectados são realizadas de forma diferente nos dois SGBDs. No MySQL (e nos demais bancos de dados relacionais), os relacionamentos são buscados através de junções, combinando registros em tempo de recuperação através de regras, como igualdade de campos em registros. No Neo4j, é feito o processo de travessia, que consiste em seguir os caminhos existentes no grafo levando em consideração a direção dos relacionamentos e suas propriedades [Robinson et al. 2013].

No Neo4j, também é possível realizar travessias em nível de comprimento. Esse tipo de travessia consiste em seguir um padrão de caminho no grafo, contendo muitos nós e relacionamentos em sequência. A linguagem Cypher disponibiliza recursos sintáticos chamados de comprimento variável para realizar a travessia através dos relacionamentos⁵. Considera-se cada relacionamento acessado como um novo nível, e esse processo é realizado de forma recursiva.

A Figura 1 apresenta exemplos de consultas em Cypher. A filtragem é realizada através da cláusula *WHERE*, comum à SQL. Nas consultas de travessia, o recurso de comprimento variável pode ser utilizado adicionando as cardinalidades após o relacionamento a ser explorado, no formato *<min>..<max>*. Na Figura, a segunda consulta explora o relacionamento *CONNECTED_TO* em três níveis, declarando *[:CONNECTED_TO*..3]*. O nível mínimo foi omitido, indicando que a travessia deve considerar desde a origem.

FILTRAGEM	COMPRIMENTO VARIÁVEL
<pre>MATCH (a:Intersection) WHERE a.num_hotels = 5 RETURN a.intersection_id</pre>	<pre>MATCH (a:Intersection)-[:CONNECTED_TO*1..3]->(b:Intersection) WHERE a.intersection_id = 213 RETURN DISTINCT b.intersection_id</pre>

Figura 1. Exemplos de consultas com filtragem e travessia em Cypher

A linguagem Cypher pode ser limitada para alguns cenários de consultas que exigem muito processamento durante a expansão do subgrafo, sendo necessária a utilização de bibliotecas de *procedures* como a APOC ou a implementação de *procedures* através da Traversal API em Java do Neo4j. A segunda opção oferece maior liberdade de escolha

³<https://neo4j.com/docs/developer-manual/current/cypher/execution-plans/>

⁴<https://neo4j.com/blog/tuning-cypher-queries/>

⁵<https://neo4j.com/docs/developer-manual/current/cypher/syntax/patterns/#cypher-pattern-varlength>

nas decisões do processamento, mas a implementação de *procedures* é consideravelmente mais complexa do que as consultas escritas de forma puramente declarativa.

4. Testes e Resultados

Esta seção apresenta experimentos realizados para comparar o desempenho dos SGBDs MySQL e Neo4j em tarefas como a carga de dados e alguns cenários de consulta, com foco na utilização de apenas recursos padrão das linguagens de acesso dos SGBDs. O objetivo principal dos experimentos é verificar se há vantagem em utilizar bancos de dados de grafo em detrimento de bancos de dados relacionais para o acesso a dados complexos.

4.1. Ambiente de testes

Os testes foram realizados através um *framework* desenvolvido em linguagem Python, versão 3.6.4. O uso do *framework* visa padronizar o acesso aos dois SGBDs e tornar os testes simétricos.

Os experimentos foram realizados sobre uma máquina dispondo de processador Intel(R) Core(TM) i5-3230M, de terceira geração. O processador dispõe de 4 núcleos (2 núcleos reais e 2 simulados, resultando em 4 *threads*), frequência de operação de 2,6GHz, 3MB de memória cache e placa de vídeo compartilhada HD Graphics 4000. A máquina dispõe de 6GB de memória RAM DDR3 com frequência de 1600 MHz.

O sistema operacional utilizado na máquina foi o Linux Mint 18.3, edição Cinnamon 64 bit. O sistema operacional e os processos nativos ocupam aproximadamente 1,2GB de memória RAM, deixando em torno de 4,5GB de memória disponíveis para a realização dos testes. Não foram utilizadas máquinas virtuais, pois a quantidade de memória disponível seria consideravelmente reduzida. Além disso, a máquina virtual sofreria influência do sistema hospedeiro, e a execução dos testes também sofreria influência do sistema operacional instalado na mesma.

4.2. Bancos de dados e ferramentas utilizadas

As funcionalidades disponíveis de cada SGBD foram analisadas para decidir quais seriam utilizadas e se alguma configuração de ambiente precisava ser alterada. No MySQL, optou-se por utilizar duas das *engines* disponíveis: InnoDB e MyISAM. A *engine* InnoDB foi escolhida pelo fato de implementar todas as propriedades ACID e por ser o padrão do MySQL. Já MyISAM foi escolhida por ser simplificada e geralmente recomendada para dados mais utilizados para leitura do que escrita.

Duas versões do MySQL foram utilizadas: 5.7.21 e 8.0.4-rc. O Neo4j foi utilizado em sua versão a 3.3.2. A conexão aos SGBDs foi realizada por conectores para linguagem Python mencionados em seus guias e recursos para desenvolvedores⁶⁷. Para o MySQL, utilizou-se o conector oficial, MySQL Connector/Python, e para o Neo4j, o conector Py2neo.

4.3. Definição do domínio de dados

Para a realização do estudo, foi utilizado o grafo da rede rodoviária do estado da Califórnia, proveniente do *Stanford Network Analysis Project* (SNAP)⁸. O grafo foi escolhido

⁶<https://www.mysql.com/products/connector/>

⁷<https://neo4j.com/developer/python/>

⁸<https://snap.stanford.edu>

por ser um domínio de dados real, com quantidade significativa de nós e arestas e com amplo uso de relacionamentos.

Neste grafo, interseções e pontos de extremidade são representados por nós, enquanto as ruas conectando essas interseções ou pontos de extremidade da estrada são representadas por arestas (relacionamentos) não direcionadas [Leskovec and Krevl 2014]. A base de dados possui 1.965.206 nós e 2.766.607 relacionamentos. Os nós e relacionamentos não possuem nenhum atributo.

4.4. Pré-processamento

A base de dados foi processada para que os nós fossem extraídos. Dados fictícios com valores pseudoaleatórios foram adicionados aos nós para permitir consultas com filtragem. Os nós receberam três novos atributos: *num_hotels*, que representa número de hotéis, *num_restaurants*, como o número de restaurantes e *num_gas_stations*, que representa número de postos. Os valores desses atributos foram gerados através de sorteio ponderado.

A Tabela 1 apresenta os possíveis valores dos novos atributos, com seus respectivos pesos. Os pesos totalizam 100% e foram divididos de modo que alguns valores permitissem alta seletividade quando utilizados ou combinados em filtros.

Tabela 1. Valores possíveis e pesos para sorteio nos atributos fictícios adicionados aos nós

Valor	0	1	2	3	4	5
Peso	30%	30%	20%	14%	5%	1%

4.5. Modelagem

No MySQL, o grafo foi representado como um relacionamento muitos para muitos, fazendo uso de duas tabelas, considerando que múltiplas arestas podem partir de um nó ou chegar até o mesmo simultaneamente. Os nós foram representados por uma tabela *Intersection*, contendo os três atributos adicionados no pré-processamento e o identificador do nó, *intersection_id*, como chave primária. As ruas foram representadas por uma tabela *Road*, que contém referências aos identificadores dos nós de partida e chegada. Essa estrutura pode ser observada na Figura 2.a).

No Neo4j, apesar da ausência de esquema, os nós seguem a mesma estrutura e nome da tabela *Intersection* do MySQL, enquanto os relacionamentos foram nomeados *CONNECTED_TO*. O Neo4j adiciona, por padrão, um atributo *id* imutável em todos os nós criados. Para evitar quaisquer problemas de identificação, foi criado o *intersection_id* nos nós. Uma prévia da estrutura do Neo4j é representada na Figura 2.b).

4.6. Metodologia dos testes

Para garantir a homogeneidade dos testes, algumas medidas foram tomadas para reduzir o impacto do sistema operacional e para restringir o uso de cache apenas no contexto de cada banco de dados. Uma das medidas foi o encerramento dos processos não nativos da máquina de modo a garantir a maior quantidade de memória livre.

Os testes de carga de dados foram realizados uma vez cada, com apenas um SGBD ativo por vez. Para os testes onde uma única operação exige múltiplas transações, o

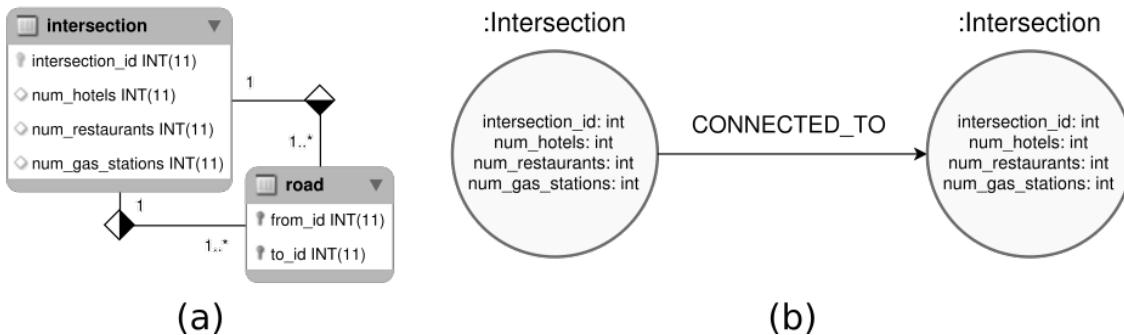


Figura 2. Representação da estrutura do grafo no MySQL e no Neo4j

tempo total de execução é o somatório do tempo decorrido em cada transação. Nas de dados, cada teste foi executado trinta vezes no MySQL e no Neo4j, de forma intercalada. Em cada iteração, todas as consultas são executadas nas *engines* MyISAM e InnoDB e no Neo4j, nessa ordem. O resultado de cada consulta é a média aritmética dos tempos coletado nas trinta execuções, excluindo os 10% melhores e os 10% piores resultados.

A versão 5.7.21 do MySQL foi utilizada em todos os testes, exceto os testes de consultas por caminhos onde nós satisfazem uma condição. Neste cenário de teste, foi utilizado o MySQL 8.0.4-rc, que possui a implementação de funções recursivas (*WITH RECURSIVE*). Estes testes foram executados separados dos demais, mas seguindo a organização de número de execuções, cômputo de resultados e execuções intercaladas.

4.7. Carga de dados

A carga é importante em cenários onde se deseja armazenar os dados complexos temporariamente em um SGBD para resolver problemas pontuais que demandem maior processamento. A carga foi realizada de duas formas: a inserção tradicional e a importação de dados de arquivos.

A inserção tradicional foi feita através dos comandos *INSERT INTO* e *CREATE*, comandos padrão de inserção de registros das linguagens SQL e Cypher, respectivamente. A importação por arquivos foi realizada através de recursos das linguagens SQL e Cypher que permitem que o conteúdo de arquivos sejam processados e inseridos nos bancos de dados. Em ambos os casos, foram tomadas medidas de otimização para acelerar as cargas.

No MySQL, foram desativadas as verificações de singularidade e de chave estrangeira, além da desativação do *commit* automático. No Neo4j, foram criados índices no atributo identificador de cada nó anterior à inserção dos dados, pois o SGBD usa propagação em segundo plano para concluir a operação. A criação dos índices é de suma importância uma vez que no Neo4j, é necessário buscar os nós envolvidos em um relacionamento para poder criá-lo.

A inserção tradicional nos dois SGBDs foi realizada em partes, com transações de 40.000 objetos cada. A quantidade de objetos por transação foi escolhida de forma empírica. A medida se fez necessária por limitações de memória para o processamento do Neo4j, e foi replicada nas inserções no MySQL para garantir homogeneidade nos testes. Para a importação, a mesma quantidade de objetos por transação foi utilizada no Neo4j. Já a SQL não permite esse tipo de quebra transacional.

A Tabela 2 apresenta os resultados dos testes, em segundos. A *engine* MyISAM no MySQL apresentou o melhor desempenho em todos os casos. O desempenho do Neo4j foi inferior aos do MySQL em qualquer sentido, com inserções consideravelmente mais dispendiosas mesmo na importação, com comandos otimizados para inserção massiva.

Tabela 2. Desempenho das cargas de dados no MySQL e no Neo4j

	Importação		Inserção	
	Nós	Relacionamentos	Nós	Relacionamentos
MySQL MyISAM	3,73 s	7,05 s	442,41 s	576,25 s
MySQL InnoDB	19,04 s	60,95 s	451,94 s	629,54 s
Neo4j	73,79 s	103,09 s	756,61 s	860,89 s

4.8. Consultas com filtragem de dados

A segunda modalidade de testes foi a de consultas com filtragens de dados. Os testes desta modalidade foram divididos em aplicações de filtros de igualdade sobre um e dois atributos indexados. Para as consultas com um filtro, buscou-se por nós com valor 5 no campo *num_hotels*, que correspondem a 1% da base de dados. Nas consultas com dois atributos, foram utilizados *num_restaurants*, com valor igual a 2, e *num_gas_stations*, com valor 4, combinados com o operador lógico *AND*. Os valores escolhidos representam 20% e 5% dos nós ou registros, respectivamente, e combinados também recuperaram em torno de 1% da base de dados.

Os resultados são apresentados pela Tabela 3. O Neo4j apresentou desempenhos inferiores aos obtidos pelas *engines* do MySQL, com tempo médio de execução próximo de um segundo. A *engine* InnoDB apresentou o melhor desempenho para o MySQL, com execução aproximadamente três vezes mais rápida que a de MyISAM na aplicação de um filtro. Os resultados utilizam seis casas decimais em sua representação, devido às diferenças insignificantes entre os resultados das *engines* do MySQL.

Tabela 3. Desempenho das consultas com filtragem de dados com índice

	Um filtro	Dois filtros
MySQL MyISAM	0,002980 s	0,011908 s
MySQL InnoDB	0,001191 s	0,016102 s
Neo4j	0,885031 s	0,936510 s

4.9. Consultas com cruzamento de dados

A terceira modalidade de testes explora os relacionamentos no grafo. Para isso, foram executadas consultas que fazem uso da cláusula *JOIN* no SGBD MySQL e a travessia de caminhos no Neo4j. Foram testados dois cenários de consulta explorando a vizinhança de um nó específico, buscando por todos os nós em até três níveis de relacionamentos a partir do mesmo, e todos os nós a exatos três níveis de relacionamento.

No Neo4j, foi utilizado o recurso de comprimento variável. No MySQL, o comportamento foi simulado utilizando diversas junções na tabela de relacionamentos (*Road*). Os resultados podem ser observados na Tabela 4. A *engine* InnoDB e o Neo4j apresentaram os melhores desempenhos, em ordem, com tempos de execução foram na faixa dos

milissegundos. A *engine* InnoDB obteve um desempenho mais satisfatório que o Neo4j, sendo aproximadamente sete vezes mais rápida, ainda que esperava-se vantagem do Neo4j nesses cenários. O tempo de execução da *engine* MyISAM manteve-se constante.

Tabela 4. Desempenhos das consultas com cruzamento de dados

	Consulta de nós em até três níveis	Consulta de nós a três níveis
MySQL MyISAM	11,540125 s	11,522095 s
MySQL InnoDB	0,001099 s	0,000870 s
Neo4j	0,007750 s	0,006132 s

4.10. Consultas por caminhos cujos nós satisfazem uma condição

O último teste proposto consiste em uma busca de comprimento variável com condições baseadas em propriedades dos nós. Diferentemente dos testes de travessia anteriores, onde retornava-se todos os nós envolvidos no níveis compreendidos pela consulta, este cenário busca apenas pelos caminhos onde todos os nós obedecem uma determinada condição sobre um dos seus atributos, dado um nó de partida.

As consultas construídas em SQL utilizam funções recursivas (*WITH RECURSIVE*), disponíveis na versão 8.0.4-rc do MySQL, pois a recursão é necessária para que o comprimento do caminho possa ser parametrizável. As consultas realizam junções entre as tabelas *Intersection* e *Road*, e utilizam uma coluna pivô *n*, incrementada a cada chamada recursiva, como critério de parada. As consultas em Cypher, por sua vez, continuam a usar comprimento variável, agora aplicando um filtro sobre os nós de cada caminho.

Foi aplicado um filtro de igualdade sobre o atributo *num_hotels*, verificando quais caminhos existem com nós cujo valor do atributo é menor ou igual a 2. Além disso, a consulta foi executada múltiplas vezes fazendo uso de três comprimentos de caminho: 4, 6 e 8 relacionamentos.

A Tabela 5 apresenta os resultados obtidos. O MySQL apresentou tempos de execução superiores ao Neo4j, com a *engine* InnoDB obtendo o melhor desempenho no geral. Os tempos de execução do Neo4j foram de três a cinco vezes maiores que os tempos da *engine* InnoDB, dependendo do comprimento do caminho analisado. Os tempos de execução dos dois SGBDs apresentaram variação conforme o aumento do comprimento.

Tabela 5. Desempenho das consultas por caminhos com condição

	Comprimento do caminho		
	4	6	8
MySQL MyISAM	0,001931 s	0,002123 s	0,002559 s
MySQL InnoDB	0,001743 s	0,001999 s	0,002400 s
Neo4j	0,006079 s	0,008911 s	0,012885 s

5. Conclusões

Este trabalho teve a finalidade de fornecer uma visão geral a respeito do modelo relacional e do modelo de grafos, com foco no MySQL e no Neo4j, e uma comparação de desempenho em operações comuns, como carga e alguns tipos de consulta de dados.

A constatação que se faz a partir dos resultados obtidos é que os bancos de dados de grafo ainda não atingiram maturidade de recursos suficiente para superarem a eficiência dos bancos de dados relacionais ao trabalhar com dados de alta complexidade utilizando apenas suas linguagens declarativas.

O Neo4j oferece uma linguagem de acesso simples e intuitiva para grafos, com recursos predefinidos para manipulação dos dados durante a expansão dos subgrafos nas consultas. É possível que essa simplicidade seja um dos motivos para sua popularidade em detrimento das demais opções de sua categoria. Entretanto, os bancos de dados relacionais também permitem o acesso a dados complexos de forma relativamente simples. As funções recursivas das expressões de tabela comum em SQL são um exemplo de recursos tão ou até mais sofisticados que o comprimento variável da linguagem Cypher.

Mesmo em casos em que seja aceitável o uso de recursos procedurais de linguagens orientadas a grafos, pode ser mais vantajoso recorrer a outra abordagem, tanto em termos de eficiência quanto de familiaridade com a linguagem. Uma possibilidade é o uso de critérios de seleção mais abrangentes no SGBD e a transferência da complexidade do tratamento para a aplicação. Este trabalho serve como ponto de partida para que questões como essa venham a ser discutidas.

Referências

- Batra, S. and Tyagi, C. (2012). Comparative analysis of relational and graph databases. *International Journal of Soft Computing and Engineering (IJSCE)*, 2(2):509–512.
- Elmasri, R. and Navathe, S. B. (2011). *Sistemas de Banco de Dados*. Pearson Addison Wesley, São Paulo, 6 edition.
- Leskovec, J. and Krevl, A. (2014). SNAP Datasets: Stanford large network dataset collection.
- Medhi, S. and Baruah, H. K. (2017). Relational database and graph database: A comparative analysis. *Journal of Process Management. New Technologies*, 5(2):1–9.
- Ramakrishnan, R. and Gehrke, J. (2008). *Sistemas de Gerenciamento de Banco de Dados*. McGraw-Hill, São Paulo, 3 edition.
- Robinson, I., Webber, J., and Eifrem, E. (2013). *Graph Databases*. O'Reilly Media, Inc.
- Sadalage, P. J. and Fowler, M. (2012). *NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence*. Addison-Wesley Professional, 1 edition.
- Silberschatz, A., Korth, H. F., and Sudarshan, S. (2010). *Database System Concepts*. McGraw-Hill, 6 edition.
- Tiwari, S. (2011). *Professional NoSQL*. Wrox Press Ltd., Birmingham, UK, UK.
- Vicknair, C., Macias, M., Zhao, Z., Nan, X., Chen, Y., and Wilkins, D. (2010). A comparison of a graph database and a relational database: A data provenance perspective. In *Proceedings of the 48th Annual Southeast Regional Conference*, ACM SE '10, pages 42:1–42:6. ACM.

Otimização do Mapeamento de Consultas SPARQL para SQL

Mariana Machado Garcez Duarte e Carmem S. Hara

¹Departamento de Informática – Universidade Federal do Paraná (UFPR)

marijanamgd@gmail.com, carmem@inf.ufpr.br

Resumo. A Web Semântica tem como princípio a organização e publicação das informações com conteúdo semântico. Ela adota o modelo RDF como padrão de armazenamento e a linguagem SPARQL para consultas. A grande quantidade de dados de RDF existente requer que as consultas SPARQL sejam processadas de forma eficiente. Os trabalhos de [de Lima Prado 2018] e [G. Pauluk and Hara 2016] buscaram uma solução para este problema através do armazenamento de dados RDF em um Sistema Gerenciador de Banco de Dados Relacional (SGBDR) e da tradução de consultas SPARQL para SQL. Este artigo apresenta uma investigação que deu continuidade a esses trabalhos, com o objetivo de otimizar o mapeamento de consultas SPARQL para SQL, utilizando índices, visões e filtros. Os experimentos realizados determinaram o impacto desses recursos no desempenho das consultas. Constatou-se que a implementação de visões é altamente recomendável, com uma redução do tempo de processamento da consulta de até 54,4%. A utilização de filtros que desconsideram tuplas contendo valores nulos resultou em uma redução do tempo de processamento da consulta de até 58,8%.

1. Introdução

A Web Semântica é uma extensão da *World Wide Web*, que organiza informações, de tal forma que permite que computadores e humanos trabalhem em cooperação, através da organização e publicação de dados no formato RDF. Uma base de dados RDF é composta por um conjunto de triplas (*sujeito, predicado, objeto*). SPARQL é a linguagem utilizada para consultas sobre uma base de dados RDF, na qual padrões de triplas são combinados para a geração de um resultado. A grande quantidade de dados de RDF existente na Web requer que as consultas SPARQL sejam processadas de forma eficiente. Há várias formas de obter esse objetivo, sendo uma delas mapear os dados RDF para o modelo relacional e consultas SPARQL para SQL. Assim, é possível aproveitar as otimizações que um SGBDR oferece sobre a linguagem de consulta SQL.

Buscando essa solução, o trabalho de [de Lima Prado 2018] propõe o Sistema de Armazenamento Otimizado de Dados RDF em SGBDR (AORR). A estratégia adotada pelo AORR difere da abordagem direta de armazenamento RDF no modelo relacional, na qual triplas RDF são armazenadas em uma única relação com 3 atributos (uma tabela SPO - sujeito, predicado, objeto). Este mapeamento direto resulta em uma relação com cardinalidade igual ao número de triplas. No AORR, predicados de um mesmo sujeito são agrupados em uma única tupla, gerando tabelas *estruturadas*.

Dada a natureza semi-estruturada das bases RDF, nem todas as informações podem ser mapeadas para as tabelas estruturadas. Assim, o trabalho de [de Lima Prado 2018],

propõe a criação de tabelas de *Overflow* Específico, a tabela de *Overflow* Geral e as tabelas de metadados. As tabelas de *Overflow* correspondem à parte *não estruturada* da base relacional. É criada uma tabela de *Overflow* Específico para cada tabela estruturada para armazenar informações dos sujeitos pertencentes à tabela estruturada, mas que não se adequam ao seu esquema. Assim, se existe uma tabela estrutura para *Pessoa*, é criada também uma tabela *SPO Overflow_Pessoa*. Já a tabela *Overflow* Geral possui os sujeitos que não pertencem a nenhuma tabela estruturada. As tabelas de metadados fornecem informações que permitem a tradução de consultas SPARQL para SQL, sem que o usuário tenha conhecimento do esquema de mapeamento da base RDF para a base relacional. O trabalho de [G. Pauluk and Hara 2016], utiliza essas tabelas de metadados para gerar traduções do SPARQL para o SQL.

O trabalho apresentado neste artigo teve como objetivo dar continuidade a esses trabalhos com a otimização do mapeamento de consultas SPARQL para SQL de [G. Pauluk and Hara 2016], explorando a utilização de índices, filtros e visões. O objetivo dos experimentos realizados foi determinar o impacto destes recursos no desempenho das consultas. Para atingir este objetivo, um plano de atividades foi traçado, o qual possuiu três etapas e testes. Como primeira etapa, foram criados índices nas tabelas estruturadas e seus resultados foram comparados com a situação original. Como etapa seguinte, foram investigadas diferentes estratégias de consultar as tabelas de *Overflow* Específico, como através da criação de visões. Como última etapa, foram utilizadas consultas com filtros que desconsideraram tuplas contendo valores nulos.

O restante do artigo está estruturado da seguinte forma. A Seção 2 apresenta trabalhos relacionados. A Seção 3 detalha o método para transformar o RDF em relacional, proposto em [de Lima Prado 2018] bem como exemplifica a tradução de consultas proposta em [G. Pauluk and Hara 2016]. A Seção 4 descreve os experimentos realizados e seus resultados. A Seção 5 finaliza o artigo enumerando alguns trabalhos futuros.

2. Trabalhos Relacionados

Diversas abordagens para o armazenamento de dados RDF em um SGBDR foram propostas na literatura nos últimos anos. O trabalho de [Abadi et al. 2007] utiliza a abordagem baseada em propriedades, com a criação de uma tabela para cada propriedade distinta. A proposta pode resultar em uma grande quantidade de tabelas, dependendo da base RDF sendo tratada. Já o trabalho de [Bornea et al. 2013] é orientado a entidades(*entity-oriented*) e cria quatro tabelas, duas para tratar atributos multivalorados, uma para sujeitos e a última para objetos. O trabalho de [Scabora et al. 2017] mapeia todas as triplas RDF para uma única tabela, contendo múltiplas colunas. Cada vértice corresponde a uma ou mais linhas da tabela, que é preenchida com suas propriedades, até atingir uma quantidade k de colunas. Após esta quantidade ser atingida, uma nova linha é inserida na tabela. Já o AORR foi inspirado na proposta de [Pham et al. 2015], que tem por objetivo criar tabelas que agrupam sujeitos com estruturas semelhantes ou que sejam semanticamente relacionados. No entanto, a tradução de consultas SPARQL para SQL não é tratada, já que a base relacional é criada para ser diretamente consultada através do SGBDR.

3. Armazenamento de RDF em um SGBDR no Sistema AORR

Para explorar a flexibilidade do RDF com a otimização de um SGBDR, foi proposto o Sistema de Armazenamento Otimizado de Dados RDF em SGBDR (AORR), que utiliza um

SGBDR como *backend* de armazenamento RDF e processamento de consulta SPARQL. O AORR possui um módulo de extração de estrutura de armazenamento inspirado na proposta de [Pham et al. 2015], o qual é baseado no conceito de *characteristics sets* e que tem por objetivo identificar estruturas comuns dos sujeitos da base RDF. Um CS é definido como um conjunto de predicados. Um sujeito na base RDF *pertence* a um determinado *characteristics set* cs_1 se ele possui o conjunto de predicados cs_1 . Desta forma, após um processo de agrupamento, remoção e limpeza de CSs, uma tabela *estruturada* é gerada para cada CS resultante. Elas agrupam em uma tabela predicados que são comumente encontrados para um mesmo sujeito. Os dados inseridos nestas tabelas formam a parte *estruturada* da base. Como resultado, além da base relacional, são mantidas informações sobre as relações entre componentes da base original RDF e a base relacional criada. Elas são armazenadas em uma tabela denominada TB_DatabaseSchema.

Para dar suporte à heterogeneidade do RDF e permitir atualizações com triplas que não se adequam ao esquema relacional extraído, o AORR mantém um conjunto de tabelas SPO (chamadas de tabelas de *Overflow*), que correspondem à parte *não estruturada* da base relacional. Existem dois tipos de tabelas de *Overflow*: específica e geral. Existe uma tabela de *Overflow* específica para cada tabela estruturada. Ela armazena, por exemplo, predicados que são incomuns aos sujeitos armazenados na tabela estruturada. A tabela de *Overflow* geral armazena informações sobre sujeitos que não se adequam ao esquema das tabelas estruturadas ou que foram inseridos posteriormente à geração da base relacional.

sujeito	predicado	objeto
<http://dbtune.org/..60678 >	Name	Andy Halstead
<http://dbtune.org/..60678>	Type	<http://xlmns.com/foaf/0.1/Person>
<http://dbtune.org/..60678>	Label	AndyHalstead
<http://dbtune.org/..14002>	Name	Cat Stevens
<http://dbtune.org/..14002>	Name	Yusuf
<http://dbtune.org/..14002>	Type	<http://purl.org/ontology/mo/MusicArtist>
<http://dbtune.org/..25789>	Type	<http://purl.org/ontology/mo/Performance>
<http://dbtune.org/..25789>	Fk_Performer	<http://dbtune.org/..60678>
<http://dbtune.org/..25789>	Fk_Recorded_as	<http://dbtune.org/..2591>
<http://dbtune.org/..76229>	Time	1998-07-05

Tabela 1. Tabela SPO

Considere, por exemplo, uma parte de uma tabela SPO apresentada na Figura 1. O resultado da base relacional resultante do processo AORR, está ilustrado na Figura 1. Esta base possui 2 tabelas estruturadas (MusicArtistRDF) e (PerformanceRDF). Cada uma delas contém uma coluna para cada predicho comumente encontrados nos sujeitos que pertencem ao CS. Predicados infrequentes, multivvalorados ou com tipos distintos são armazenados na tabela de *Overflow* Específico de cada tabela estruturada. Assim, no exemplo, são ilustradas duas tabelas de *Overflow* específico (Overflow_MusicArtistRDF) e (Overflow_PerformanceRDF). Além das tabelas de *Overflow* Específico, há a tabela geral chamada de *Overflow*. Ela comporta os sujeitos que não se adequam a nenhuma tabela estruturada.

O AORR gera diversas tabelas de metadados, que contem as informações sobre o mapeamento da base RDF para o esquema relacional. As principais são : TB_DatabaseSchema e TB_Subj_OID. A tabela TB_DatabaseSchema relaciona os predicados de cada CS às tabelas e atributos nos quais eles são armazenados, além do tipo de cada atributo. A Figura 2 ilustra um TB_DatabaseSchema.

Music Artist RDF			Overflow_MusicArtistRDF		
OID	Name	Type	Subj	Pred	Obj
60678	Andy Halstead	< http://xmlns.com/foaf/0.1/ Person>	60678	Label	AndyHalstead
14002	Cat Stevens	< http://purl.org/ontology/mo/ MusicArtist>	14002	Name	Yusuf

PerformanceRDF				Overflow_PerformanceRDF		
OID	fk_performer	fk_performance_of	Type	Subj	Pred	Obj
25789	60678	NULL	< http://purl.org/ontology/mo/ Performance>	25789	fk_recorded_as	25791

Overflow		
OID	Pred	Obj
76229	Time	1988-07-05

Figura 1. Estrutura gerado pelo AORR

cs_identifier	PropertyName	ValueType	TableName	TableAttribute
CS1	OID	Literal	MusicArtistRDF	OID
CS1	Name	Literal	MusicArtistRDF	name
CS1	Type	Literal	MusicArtistRDF	type
CS2	OID	Literal	PerformanceRDF	OID
CS2	Type	Literal	PerformanceRDF	type
CS2	fk_performer	CS1	PerformanceRDF	fk_performer
CS2	fk_performance_of	CS5	PerformanceRDF	fk_performance_of
CS1	Name	Literal	Overflow_MusicArtistRDF	pred
CS1	Label	Literal	Overflow_MusicArtistRDF	pred
CS2	fk_recorded_as	CS3	Overflow_PerformanceRDF	pred
...
...
CSover	Label	Literal	Overflow	pred
CSover	Time	Literal	Overflow	pred
CSover	Type	Literal	Overflow	pred

Figura 2. Exemplo de um TB_DatabaseSchema

Na tabela TB_Subj_OID encontram-se informações sobre a relação da IRI do sujeito para o OID que a representa, e também, a tabela na qual esse sujeito está armazenado. Essa tabela também é utilizada para identificar se a IRI de um determinado sujeito, ou de um objeto de um relacionamento, já existem na base. A Figura 3 ilustra um exemplo desta tabela. As tabelas TB_DatabaseSchema e TB_Subj_OID possibilitam a tradução de uma consulta SPARQL para SQL. A Figura 4 apresenta um exemplo de consulta SPARQL e a consulta SQL resultante da tradução proposta em [G. Pauluk and Hara 2016].

subj	OID	tableName
< http://dbtune.org/bbc/peel/artist/000449859d55f41aad74fb36f9fd7f46 >	1	MusicArtistRDF
< http://dbtune.org/bbc/peel/perf_ins/000449859d55f41aad74fb36f9fd7f46 >	2	PerformanceRDF
< http://dbtune.org/bbc/peel/artist/0004ca7431d195cd64459fc8e784daec >	3	MusicArtistRDF
.	.	.
.	.	.
< http://dbtune.org/bbc/peel/signal/119/e1b54f76aa6d53d03fd585de690bce5f >	69116	Overflow

Figura 3. Exemplo de um TB_Subj_OID, como visto em [de Lima Prado 2018]

4. Propostas para otimização do Sistema AORR

Esta seção apresenta os experimentos realizados para determinar o impacto de algumas estratégias de otimização sobre as propostas de armazenamento e tradução de consultas

```
SELECT b.OID AS b,
       a.typeYLEIFC AS t,
       a.nameEY9TOM AS n,
       a.OID AS a
  FROM
    (SELECT MusicArtistRDF.OID,
            MusicArtistRDF.name AS nameEY9TOM,
            MusicArtistRDF.type AS typeYLEIFC
   FROM MusicArtistRDF
  UNION ALL
    SELECT t1.subj AS a, t1.obj AS t, t2.obj AS n
   FROM Overflow_MusicArtistRDF AS t1
    FULL OUTER JOIN Overflow_MusicArtistRDF AS t2
      ON t1.subj=t2.subj
   WHERE t1.pred='name' AND t2.pred='type') AS a,
  (SELECT PerformanceRDF.OID,
            PerformanceRDF.fk_performer AS fk_performerEKRX08
   FROM PerformanceRDF) AS b
  WHERE a.OID = fk_performerEKRX08
```

(a) Consulta SPARQL

```
SELECT b.OID AS b,
       a.typeYLEIFC AS t,
       a.nameEY9TOM AS n,
       a.OID AS a
  FROM
    (SELECT MusicArtistRDF.OID,
            MusicArtistRDF.name AS nameEY9TOM,
            MusicArtistRDF.type AS typeYLEIFC
   FROM MusicArtistRDF
  UNION ALL
    SELECT t1.subj AS a, t1.obj AS t, t2.obj AS n
   FROM Overflow_MusicArtistRDF AS t1
    FULL OUTER JOIN Overflow_MusicArtistRDF AS t2
      ON t1.subj=t2.subj
   WHERE t1.pred='name' AND t2.pred='type') AS a,
  (SELECT PerformanceRDF.OID,
            PerformanceRDF.fk_performer AS fk_performerEKRX08
   FROM PerformanceRDF) AS b
  WHERE a.OID = fk_performerEKRX08
```

(b) Consulta SQL

Figura 4. Tradução SPARQL para SQL

apresentados na Seção 3.

Com o objetivo de otimizar o mapeamento de consultas SPARQL para SQL, resultante dos trabalhos de [de Lima Prado 2018] e [G. Pauluk and Hara 2016], um plano de implementação foi traçado, o qual possuiu três etapas e testes. Como primeira etapa, foram criados índices nas tabelas estruturadas e seus resultados foram comparados com a situação original. Como etapa seguinte, foram investigadas diferentes estratégias de consulta à tabelas de *Overflow* Específico, como através da criação de visões. Como última etapa, foram utilizadas consultas com filtros que desconsideraram tuplas contendo valores nulos.

4.1. Experimentos

O computador utilizado para executar os experimentos foi um MacOSX Intel Core m3 1.1 GHz e com 8 GB de memória RAM. O SGBDR utilizado foi o MySQL com o mecanismo de armazenamento InnoDB. Suas especificações estão na Figura 5a.

O banco de dados utilizado possui 26 tabelas, que foram geradas a partir do processo de [de Lima Prado 2018], com o sistema AORR a partir da base de dados Peel, disponível em <http://dbtune.org/bbc/peel/>. A Figura 5b apresenta as tabelas geradas e seus tamanhos.

As tabelas estruturadas utilizadas nos experimentos são: MusicArtistRDF, PerformanceRDF, RecordingRDF, SignalRDF, TrackRDF. As tabelas multivaloradas, que correspondem a predicados multivalorados da base são: chart_positionMultivalueRDF, createdMultivalueRDF, fk_engineerMultivalueRDF, fk_engineeredMultivalueRDF, fk_performedMultivalueRDF, fk_producedMultivalueRDF, fk_sameAsMultivalueRDF, fk_sub_eventMultivalueRDF, instrumentMultivalueRDF, isrcMultivalueRDF, labelMultivalueRDF.

Como a base de dados Peel não disponibiliza um conjunto de consultas, foram definidas 6 consultas para executar os experimentos. As primeiras 5 consultas variam a complexidade pela quantidade de tabelas, enquanto a consulta 6 inclui uma tabela multivalorada. Seguem as descrições das consultas.

Table Name	Rows Count	Table Size (MB)
MusicArtistRDF	18544	1.52
Overflow	16319	1.52
Overflow_MusicArtistRDF	2842	0.27
Overflow_PerformanceRDF	12958	1.52
Overflow_RecordingsRDF	6	0.02
Overflow_SignalRDF	152	0.02
Overflow_TrackRDF	383	0.06
PerformanceRDF	28600	2.52
RDF_Triple	268590	44.58
RecordingRDF	3924	0.52
SignalRDF	5658	0.42
TB_DatabaseSchema	60	0.02
TB_FullPredicate	38	0.02
TB_Subj_OID	76455	7.52
TrackRDF	19335	2.52
chart_positionMultivalueRDF	1502	0.08
createdMultivalueRDF	1522	0.08
fk_engineerMultivalueRDF	3801	0.16
fk_engineeredMultivalueRDF	3801	0.16
fk_performedMultivalueRDF	11924	0.44
fk_producedMultivalueRDF	3640	0.16
fk_sameAsMultivalueRDF	126	0.02
fk_sub_eventMultivalueRDF	29590	1.52
instrumentMultivalueRDF	9098	0.41
isrcMultivalueRDF	689	0.06
labelMultivalueRDF	5787	0.28

Variable_name	Value
innodb_version	5.7.18
protocol_version	10
slave_type_conversions	
tls_version	TLSv1, TLSv1.1, TLSv1.2
version	5.7.18
version_comment	Homebrew
version_compile_machine	x86_64
version_compile_os	osx10.12

(a) Detalhamento da Configuração do MySQL.

(b) Configuração das entradas de dados por tabela.

Figura 5. Ambiente Experimental

Consulta 1: faz uma busca na tabela MusicArtistRDF, extraiendo o nome e tipo dos artistas.

Consulta 2: faz uma busca nas tabelas MusicArtistRDF e PerformanceRDF, retornando o nome e tipo do artista, além do local no qual o artista realizou uma performance. Esta é a consulta ilustrada na Figura 4.

Consulta 3: faz uma busca nas tabelas RecordingRDF, SignalRDF e TrackRDF. Ela procura para cada gravação, o tipo de sinal gravado signal e como ele foi publicado.

Consulta 4: faz uma busca nas tabelas RecordingRDF, SignalRDF, TrackRDF e MusicArtistRDF. Ela retorna as mesmas informações que a consulta 3 e adiciona-mente quem produziu e qual tipo de sinal.

Consulta 5: faz uma busca nas tabelas RecordingRDF, SignalRDF, TrackRDF e MusicArtistRDF. Ela retorna todas as informações da Consulta 4, mais informações sobre o Track que corresponde à publicação do sinal.

Consulta 6: faz uma busca nas tabelas SignalRDF, TrackRDF e na tabela multivalorada chart_positionMultivalueRDF. Ela retorna o sinal que foi publicado e seu tipo. Busca também o título, seu label e tipo do track e quais posições ele esteve, na tabela multivalorada.

4.2. Impacto da Criação de índices

Nesta subseção é determinado o impacto da criação de índices sobre todas as tabelas no atributo OID. As consultas foram executadas com duas configurações. Na primeira, é considerada a existência de índice B+ somente nas tabelas de metadados TB_DatabaseSchema e TB_Subj_OID. Estes índices são importantes para o processo de tradução das consultas. Na segunda configuração, são gerados índices chamados de id_OID, sobre o atributo OID nas tabelas estruturadas MusicArtistRDF, PerformanceRDF, RecordingRDF, SignalRDF, TrackRDF. As tabelas utilizam o OID como chave primária e podem impactar o tempo de execução da consulta SQL.

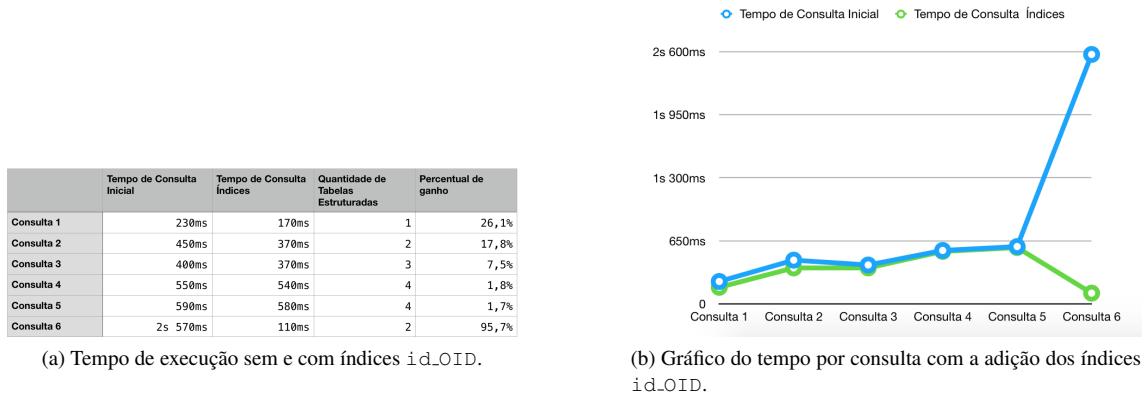


Figura 6. Impacto dos índices sobre a atributo OID

Nas Figuras 6a e 6b, é possível observar os tempos de consulta na configuração inicial e após a adição dos índices `id_OID`.

Foi possível concluir que houve um ganho médio de 11% no tempo de consulta nas tabelas que não são multivvaloradas com a criação dos índices. A Consulta 6 utilizou uma tabela multivvalorada e obteve um ganho significativo de 96%.

As consultas 1-5 não tiveram um ganho significativo em razão de serem consultas que não filtram o resultado por valores específicos, mas apenas executam junções sobre múltiplas tabelas. Foi também observado que com a junção de mais tabelas, o ganho proporcionado pelo índice é menor. É possível verificar essa constatação na Figura 6a. Portanto, o incremento do ganho da utilização de índices foi pouco significativo. Foi constatado pelo `explain` da consulta SQL, que para as consultas 3, 4 e 5, os índices não são utilizados. É realizado um *Block Nested Loop* e não *Index-Nested Loop*, que utiliza índices.

Para uma melhor análise da melhoria da consulta 6, foi executado o `explain` da consulta na situação inicial e o `explain` da consulta com índices. Como a consulta 6 possui busca em uma tabela multivvalorada, o MySQL realiza *join* entre a tabela TrackRDF e a multivvalorada chart_positionMultivalueRDF. O MySQL segundo [Oracle 2017], utiliza índices para extrair linhas de outras tabelas quando realiza junção entre tabelas. Ou seja, retira a necessidade de escanear a tabela estruturada TrackRDF múltiplas vezes. Na inexistência do índice, a consulta 6 utiliza *Block Nested Loop*, o que justifica a grande diferença no tempo de execução.

4.3. Criação das visões

Com o objetivo de otimizar as buscas nas tabelas de *Overflow* Específico, foram criadas visões com a mesma estrutura da tabela estruturada a qual ela está associada. Desta forma, consultas que envolvem predicados que pertencem à tabela estruturada podem ser processadas utilizando a visão, facilitando o processo de tradução.

Um exemplo da criação da visão sobre a tabela estruturada MusicArtistRDF está ilustrado na Figura 7(a). A consulta da Figura 4(b) utilizando a visão é apresentada na Figura 7(b).

As 6 consultas foram executadas com as duas formas de tradução: utilizando di-

```

CREATE VIEW MusicArtistRDF_View AS
    SELECT t1.subj as OID,
        t1.obj as name,
        t2.obj as type
    FROM Overflow_MusicArtistRDF as t1
    FULL OUTER JOIN
        Overflow_MusicArtistRDF as t2
    ON t1.subj=t2.subj
    WHERE t1.pre='name' and t2.pred='type'

```

(a) Criação da visão

```

SELECT b.OID as b,
    a.typeYLEIFC as t,
    a.nameEY9TOM as n,
    a.OID as a
    FROM (SELECT MusicArtistRDF.OID,
        MusicArtistRDF.name as nameEY9TOM,
        MusicArtistRDF.type as typeYLEIFC
    FROM MusicArtistRDF
    UNION ALL
    SELECT MusicArtistRDF_View.OID,
        MusicArtistRDF_View.name as nameEY9TOM,
        MusicArtistRDF_View.type as typeYLEIFC
    FROM MusicArtistRDF_View) AS a,
    (SELECT PerformanceRDF.OID,
        PerformanceRDF.fk performer
    AS fk_performerEKRX08
    FROM PerformanceRDF) AS b
    WHERE a.OID = fk_performerEKRX08

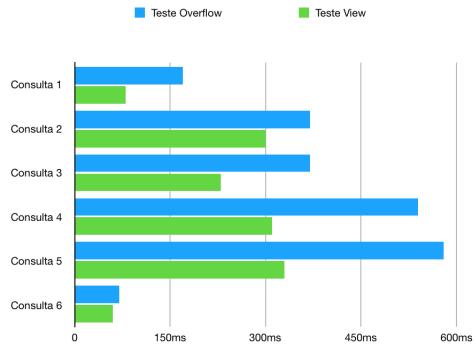
```

(b) Consulta SQL utilizando a visão

Figura 7. Tradução SQL utilizando uma visão

	Tempo de Consulta Inicial	Teste Overflow	Teste View
Consulta 1	230ms	170ms	80ms
Consulta 2	450ms	370ms	300ms
Consulta 3	400ms	370ms	230ms
Consulta 4	550ms	540ms	310ms
Consulta 5	590ms	580ms	330ms
Consulta 6	2s 570ms	70ms	60ms

(a) Tempo por consulta da situação inicial, com índices e com visões



(b) Gráfico do tempo por consulta da situação inicial, com índices e com visões

Figura 8. Impacto das Visões

retamente o *Overflow* e utilizando as visões. É importante destacar que os índices criados nos experimentos descritos na SubSeção 4.2 continuaram implementados. Os tempos de execução das consultas utilizando os dois tipos de tradução são apresentados nas Figuras 8a e 8b. Eles mostram um ganho de desempenho considerável utilizando a estratégia visões, que varia de 7% a 54,4%.

4.4. Impacto na Utilização de filtros

Neste experimento foram consideradas consultas que introduzem filtros para desconsiderar atributos com valores nulos `IS NOT NULL`. A consulta da Figura 7(b) com a adição de filtros é apresentada na Figura 9 e os resultados obtidos estão nas Figuras 10a e 10b.

A redução do tempo de processamento com o filtro pode ser explicada pela redução das tabelas intermediárias geradas com a aplicação do filtro. Foi executado o *explain* da Consulta 2, que fez busca direta no *Overflow*. Constatou-se uma redução de 31.190 linhas para 28.069 linhas com a aplicação do filtro já no inicio da busca. Após a união na operação, observa-se que as entradas de dados filtradas reduziram na mesma proporção.

Analogamente, o *explain* da mesma consulta, utilizando visões, reduz de 43.513.880

```

SELECT b.OID as b,
       a.typeYLEIFC as t,
       a.nameEY9TOM as n,
       a.OID as a
  FROM (SELECT MusicArtistRDF_OID,
               MusicArtistRDF.name as nameEY9TOM,
               MusicArtistRDF.type as typeYLEIFC
          FROM MusicArtistRDF
         WHERE MusicArtistRDF.name IS NOT NULL and
               MusicArtistRDF.type IS NOT NULL
UNION ALL
SELECT MusicArtistRDF_View.OID,
       MusicArtistRDF_View.name as nameEY9TOM,
       MusicArtistRDF_View.type as typeYLEIFC
  FROM MusicArtistRDF_View
 WHERE MusicArtistRDF_View.name IS NOT NULL and
       MusicArtistRDF_View.type IS NOT NULL) AS a,
(SELECT PerformanceRDF_OID,
       PerformanceRDF_fk_performer
      AS fk_performerEKRX08
  FROM PerformanceRDF
 WHERE PerformanceRDF_fk_performer IS NOT NULL) AS b
 WHERE a.OID = fk_performerEKRX08
    
```

Figura 9. Tradução SQL utilizando filtros

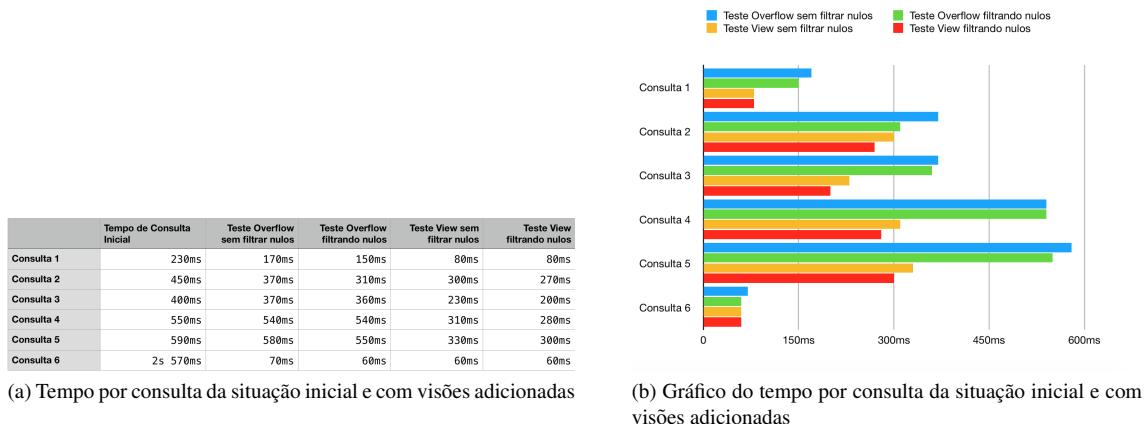


Figura 10. Impacto da utilização de filtros e comparação geral

linhas para 39.162.489, já na primeira operação de busca. Após a união na décima primeira operação, observa-se que as entradas de dados filtradas reduziram na mesma proporção.

Os resultados dos experimentos realizados mostraram que o mapeamento utilizando visões foi bastante vantajoso com relação ao mapeamento processando consultas diretamente no *Overflow Específico*. Além disso, a redução de tabelas intermediárias com a aplicação do filtro `IS NOT NULL` apresentou uma melhoria no tempo de processamento de 7% a 58,5%.

5. Conclusão

Este artigo apresentou a pesquisa que explorou possíveis otimizações de consultas do sistema AORR, que é um modelo de armazenamento de uma base de dados RDF utilizando um SGBDR como *backend* de armazenamento. O AORR é composto de dois módulos: o primeiro realiza a geração de um esquema relacional a partir de uma base RDF e o outro,

converte de uma consulta SPARQL para SQL, compatível com a estrutura de dados criada. O objetivo deste trabalho foi explorar recursos resultantes dos módulos anteriores, visando otimizar o desempenho das consultas.

As técnicas exploradas foram a criação de índices, a utilização de visões no mapeamento de consultas e aplicação de filtros. Os experimentos mostraram que a criação de índices sobre os identificadores das tabelas estruturadas resultam em um ganho médio de 11%, em razão de serem consultas que não filtram o resultado por valores específicos, mas apenas executam juntas sobre múltiplas tabelas. O ganho com a criação de visões mostrou-se bastante significativo, apresentando uma melhoria média de 54,4%, sem filtro de valores nulos e de 58,5% com o filtro. Pode-se concluir que a implementação de visões no ambiente AORR foi altamente recomendável para se obter otimização, assim como a utilização filtros de *IS NOT NULL*.

Uma otimização que possivelmente teria um grande impacto no sistema AORR é o armazenamento da tabela de metadados TB_DatabaseSchema em uma estrutura em memória, dado que o tempo de tradução utilizando o procedimento de [G. Pauluk and Hara 2016] é alto e utiliza bastante essa estrutura em sua tradução. Outra possível otimização poderia ser a análise de carga das seleções mais frequentes, para a geração de índices clusterizados. A pesquisa não explorou a otimização de buscas no *Overflow Geral*. A criação de índices adicionais sobre o predicado e objeto desta tabela poderia ser também explorada. Por fim, poderiam ser realizados experimentos adicionais com a execução de consultas em outros SGBDRs e com outras estruturas de indexação.

Referências

- Abadi, D. J., Marcus, A., Madden, S. R., and Hollenbach, K. (2007). Scalable semantic web data management using vertical partitioning. In *Proceedings of the International Conference on Very Large Data Bases*.
- Bornea, M. A., Dolby, J., Kementsietsidis, A., Srinivas, K., Dantressangle, P., Udrea, O., and Bhattacharjee, B. (2013). Building an efficient rdf store over a relational database. In *Proceedings of the ACM SIGMOD International Conference on Management of Data Conference*.
- de Lima Prado, R. (2018). Armazenamento otimizado de dados RDF em um SGBD relacional. Master's thesis, UFPR, Programa de Pós-graduação em Informática.
- G. Pauluk, J. and Hara, C. S. (2016). Processamento de consultas SPARQL em um SGBD relacional. Trabalho de Graduação, UFPR, Bacharelado em Ciência da Computação.
- Oracle (2017). Mysql 5.6 reference manual.
- Pham, M.-D., Passing, L., Erling, O., and Boncz, P. (2015). Deriving an emergent relational schema from RDF data. In *Proceedings of the International World Wide Web Conferences*.
- Scabora, L. C., Oliuvera, P. H., Kaster, D. S., Traina, A. J. M., and Junior, C. T. (2017). Relational graph data management on the edge: Grouping vertices' neighborhood with edge-k. In *Proceedings of the Brazilian Symposium on Databases*.

Comparação entre Diferentes Implementações de BK-trees para o Problema de Busca por Intervalo

Andre Luciano Rakowski¹, Natan Luiz Paetzhold Berwaldt^{1,2}, Mauricio Vielmo Schmaedeck¹, Sergio Luis Sardi Mergen¹

¹Universidade Federal de Santa Maria
Santa Maria – RS – Brasil

²Programa de Educação Tutorial - Ciência da Computação

{alrakowski, nlberwaldt, mvschmaedeck, mergen}@inf.ufsm.br

Resumo. No contexto de recuperação de informação, o problema de busca por similaridade para consultas por intervalo é caracterizado pela busca de palavras cuja distância de edição até a consulta seja menor do que um limite estipulado. BK-tree é um método de indexação que traz bons resultados para esse tipo de consulta. Experimentos mostram que, apesar da sua simplicidade, na busca textual sua eficiência é superior à outras abordagens. No entanto, nenhuma informação foi dada a respeito da estrutura da árvore. Neste artigo são analisadas formas diferentes de implementação desta estrutura de indexação. Os experimentos elaborados comparam as variações investigadas em termos de custo em espaço, tempo de indexação e tempo de busca.

Abstract. In the context of information retrieval, similarity search for range queries is the problem of finding words whose edit distance to a query are smaller than a determined distance. BK-trees is a pivot based indexing mechanism that achieves good results for this kind of query. Experiments show that, despite its simplicity, the efficiency is superior to other approaches when it comes to text searching. However, no information is provided concerning the underlying structure of the tree. In this paper, we analyze different ways of implementing this indexing structure. Our experiments compare the investigated variations in terms of space cost, indexing time and search time.

1. Introdução

A busca por similaridade de palavras tem por objetivo encontrar, dentro de um dicionário, palavras que sejam parecidas a alguma palavra chave usada como consulta [Chen et al. 2017]. A área da recuperação de informação pode usar esse tipo de busca em diversos cenários, como por exemplo, sugerindo melhores termos a serem usados em consultas por palavra chave. O problema pode ser reformulado a partir do conceito de distância entre as palavras. Nesse sentido, existe a busca por intervalo, cujo objetivo é procurar por todas as palavras que estejam a uma distância máxima r da palavra de consulta.

Ao longo dos anos, diversas técnicas foram propostas com a finalidade de averiguar a distância entre duas palavras. Uma que ganhou notoriedade é chamada de distância de edição de Levenshtein [Navarro 2001]. A distância mede o número de operações de

edição necessárias para transformar uma palavra na outra. A importância dessa técnica é o fato de ela respeitar a inigualdade triangular, o que permite que sejam inseridas em uma estrutura de índice em espaço métrico. Valer-se de uma estrutura de índice agiliza a busca por intervalo, evitando que todo o dicionário precise ser consultado [Zezula et al. 2006].

Uma das técnicas de indexação em espaço métrico é uma árvore chamada BK-tree [Burkhard and Keller 1973]. Sua construção é baseada no conceito de pivôs, e seu algoritmo é razoavelmente simples. Apesar da simplicidade, tem boa capacidade de filtragem, ou seja, consegue podar um número considerável de ramos da árvore que não levam ao resultado desejado.

Recentemente, o trabalho de [Chen et al. 2017] mediou o desempenho dessa estrutura em termos de espaço e tempo de resposta. No entanto, o trabalho não detalha a forma como que a árvore foi implementada. É importante que esse aspecto seja analisado, ainda mais levando em consideração que existem possibilidades de implementação mais dispendiosas em termos de memória.

Dessa forma, este artigo visa criar diferentes implementações da estrutura de dados BK-tree, e usá-las em testes de desempenho para o problema de busca por intervalo. O objetivo principal é verificar quais delas se sobressaem em termos de espaço e tempo de busca e tempo de indexação. Além disso, o artigo analisa se o uso adicional de memória é compensado por um ganho significativo no tempo de resposta.

O artigo está estruturado da seguinte forma: A seção 2 apresenta as principais técnicas de indexação para busca por similaridade baseadas em pivôs. A seção 3 as diferentes formas de implementação consideradas para BK-trees. A seção 4 é reservada aos experimentos. As conclusões são apresentadas na seção 5.

2. Busca em Espaço Métrico baseada em Pivôs

Os algoritmos de busca em espaço métrico são aqueles que calculam a distância d entre objetos a e b (a partir de uma função $d(a, b)$) de modo que quatro propriedades sejam satisfeitas: 1) simetria: $d(a, b) = d(b, a)$; 2) não negatividade: $d(a, b) \geq 0$, 3) identidade: $d(a, b) = 0$, se $a = b$; 4) inigualdade triangular: $d(a, b) \leq d(a, c) + d(b, c)$ [Zezula et al. 2006].

Diversas funções satisfazem essas propriedades. Para o caso específico em que se mede a distância entre palavras, uma das funções mais conhecidas e usadas é a distância de edição de Levenshtein [Navarro 2001]. Essa distância mede o número de operações de remoção, inserção e substituição de caracteres que transformam uma palavra em outra. Por exemplo, a palavra 'erbd' está a uma distância dois de 'errc', pois uma pode ser transformada na outra com duas operações de substituição.

As distâncias podem ser usadas em mecanismos de filtragem baseada em pivôs [Chen et al. 2015]. Nesse contexto, os pivôs são objetos para os quais são guardadas distâncias até outros objetos indexados. Sabendo-se a distância de um pivô p até um objeto o , e a distância de um objeto de consulta q até p , pode-se determinar, através da inigualdade triangular, se o objeto o está ou não a uma distância máxima de q . Esse conceito é empregado em estruturas de índice em forma de árvore. As principais estruturas são BK-trees [Burkhard and Keller 1973], FQ-tree [Baeza-Yates et al. 1994] e VP-Trees [Yianilos 1993].

Cada nó de uma BK-tree possui uma palavra indexada. Uma aresta entre um nó pai e um nó filho leva à palavras que estejam a uma mesma distância do nó pai. Dado um objeto de busca q e uma distância máxima permitida r , e sabendo-se a distância d do nó atual n até um ramo específico, o algoritmo de busca visita o ramo somente se $d(q, n) - r \leq d \leq d(q, n) + r$. Ao ser visitado, a palavra armazenada no nó é adicionada ao resultado somente se ela estiver à uma distância máxima r do objeto q .

Por exemplo, a Figura 1 mostra a visão abstruída de uma árvore que contém as palavras 'sbbd', 'sbes', 'sbie', 'erad', 'erbd', 'errc' e 'wei'. Como pode-se ver, o nó raiz leva a ramos que contenham palavras que estejam à distâncias 2, 3 e 4 da palavra 'sbbd'. Supondo que o objeto de consulta seja a palavra 'sbia', e que $r = 1$, os nós marcados na figura são aqueles que satisfizeram essa condição (os demais sequer foram visitados). Dentre os nós que foram efetivamente visitados, apenas 'sbie' foi adicionado à resposta.

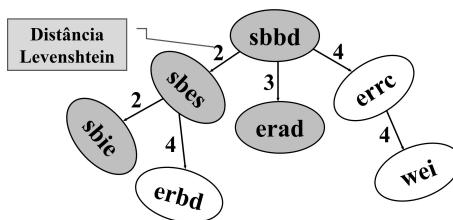


Figura 1. Exemplo de BK-tree utilizada para verificação ortográfica.

A estrutura de dados FQ-tree (Fixed Queries trees) utiliza a mesma abordagem das BK-trees. A diferença reside no fato de o mesmo pivô ser utilizado para todos os nós que estejam no mesmo nível da árvore. Como pode-se ver pelo segundo nível da árvore da Figura 1, as BK-trees não possuem esse requisito, já que três pivôs diferentes são utilizados.

Por fim, a estrutura de dados VP-tree (Vantage Point tree) caracteriza-se por procurar escolher pivôs que dividam o espaço métrico de forma balanceada, de modo que metade dos objetos estejam a uma distância até um pivô menor do que um raio definido, enquanto a outra metade esteja a uma distância maior. Durante a construção da árvore, essa divisão ocorre recursivamente, até que não seja mais possível realizar divisões (ou seja, quando restar apenas um objeto no ramo criado). Uma extensão, chamada MVP-Tree (Multiple Vantage Point tree), permite que cada nó possua múltiplos pivôs, em vez de um só [Bozkaya and Ozsoyoglu 1997].

De todas as abordagens mencionadas, a BK-tree é a que possui a menor complexidade de implementação. Isso se deve ao fato de ela ser a única onde os ramos da árvore são gerados sem preocupação com balanceamento (os ramos são criados conforme a ordem com que as palavras são indexadas). Essa característica pode levar à árvores desbalanceadas, aumentando o número de nós acessados a cada busca. No entanto, essa afirmação se aplica a objetos com poucas dimensões (como distâncias geográficas). Quando os objetos são palavras, muitas dimensões estão envolvidas (os caracteres de cada palavra). Isso faz com que os objetos se encontrem todos igualmente afastados entre si, o que dificulta a obtenção de um bom fator de balanceamento [Chávez and Navarro 2003].

De fato, experimentos recentes mostram que, quando usada para busca de pala-

vras, as BK-trees apresentam um tempo de processamento inferior às demais estruturas [Chen et al. 2017]. Apesar de ser um resultado relevante, nenhuma informação foi fornecida referente à como essa estrutura foi implementada.

3. Proposta

A seção 2 apresentou um exemplo abstrato de uma BK-tree. Já esta seção discute diferentes estratégias para a representação concreta dos nós dessa árvore. Duas estratégias básicas foram trabalhadas: baseada em vetores e baseada em listas encadeadas. Para cada uma delas, duas implementações foram desenvolvidas. A escolha da implementação pode levar a diferenças no desempenho, considerando custo de memória, de indexação, e principalmente, custo de busca. As próximas seções apresentam as vantagens e desvantagens de cada variação.

3.1. Processamento Geral de uma Busca

Antes de apresentar as variações implementadas, é importar mencionar o que está envolvido durante o processo de busca por intervalo em uma BK-tree. Esse processo está descrito em pseudo-código no Algoritmo 3.1. A função recebe quatro parâmetros: o nó pai sendo processado no momento, a palavra usada como consulta, a distância máxima permitida e a lista de palavras a serem retornadas.

O algoritmo é dividido em duas partes. Na primeira parte ocorre a aplicação da função de distância, responsável por calcular a distância de edição entre a palavra do nó e a palavra da consulta. Caso a distância encontrada d satisfaça o critério de busca (for menor ou igual a $dist$), a palavra é adicionada nos resultados. A segunda parte se encarrega de visitar recursivamente todos os filhos do nó pai que estejam à uma distância mínima de $d - dist$ e uma distância máxima de $d + dist$. Devido à propriedade da inigualdade triangular, esses são os únicos ramos que podem conter respostas para a consulta. Uma checagem é necessária para certificar-se de que a distância mínima não seja negativa.

```
BUSCA_INTERVALO(no_pai, consulta, dist, palavras_retornadas)
1: if no_pai.palavra == Ø then
2:   return
3: end if
4: d ← distância(no_pai.palavra, consulta)
5: if d <= dist then
6:   adicionar no_pai.palavra em palavras_retornadas
7: end if
8: min_dist ← d - dist
9: if min_dist < 0 then
10:   min_dist = 0
11: end if
12: max_dist ← distância + dist
13: busca_filhos(no_pai, consulta, min_dist, max_dist, palavras_retornadas)
```

Algorithm 3.1: ESTRUTURA GERAL DO ALGORITMO DE BUSCA POR INTERVALO.

A forma com que os filhos são acessados depende da forma como a árvore foi implementada. As próximas seções apresentam diferentes possibilidades de estruturação dos nós, salientando a forma como o acesso é realizado.

3.2. Estratégia Baseada em Vetores

Na estratégia baseada em vetores, cada nó possui um vetor que armazena os seus filhos. A distância de edição entre um pai e um filho é representada pela posição correspondente

nesse vetor. Por exemplo, um nó na posição dois indica que esse nó está a uma distância de edição igual a dois em relação ao seu pai.

As variações dessa abordagem devem-se ao uso de alocação estática ou dinâmica. Em ambos os casos, a principal vantagem é o acesso direto. Ou seja, todos os nós que estejam dentro intervalo de busca podem ser acessados pelos seus índices. Já a desvantagem se refere ao excessivo consumo de memória. As próximas seções discorrem a respeito dessas questões.

3.2.1. Uso de Alocação Estática

Com vetores estáticos, o tamanho do vetor em cada nó é predefinido. Para garantir que a capacidade do vetor seja suficiente, o tamanho deve suportar a maior distância de edição possível entre uma palavra de busca q e uma palavra indexada p . Considerando $D = \{p_1, \dots, p_n\}$ como sendo o dicionário de palavras indexadas, e $tam(p)$ como a função que retorna o número de caracteres de p , a maior distância possível pode ser definida como $H \leftarrow \arg \max_{p \in A} tam(p)$. Ou seja, ela corresponde ao tamanho da maior palavra indexada.

A Figura 2 (a) ilustra o uso de vetores estáticos para uma configuração qualquer de nós. Observe que todos os vetores possuem o mesmo tamanho (cinco). Nesse exemplo hipotético, a maior palavra que pode ser indexada possui quatro caracteres. Ou seja, o vetor tem espaço para distâncias que variem de zero ao máximo possível(quatro).

O Algoritmo 3.2 descreve como a busca nos filhos ocorre quando os vetores são alocados estaticamente. O processo é baseado em um laço, onde em cada iteração é acessado um dos filhos do intervalo permitido.

```
BUSCA_FILHOS(no_pai, consulta, min_dist, max_dist, palavras_retornadas)
1: while min_dist <= max_dist do
2:   busca.Intervalo(no_pai.vetor.filhos[min_dist], consulta, palavras_retornadas)
3:   min_dist += 1
4: end while
```

Algorithm 3.2: BUSCAS DOS FILHOS DE UMA BK-TREE USANDO VETORES ESTÁTICOS.

Um aspecto negativo dessa estratégia refere-se ao problema da sub-utilização do espaço. Mesmo que a distância máxima usada por um nó pai seja inferior a H , todas as posições são alocadas. O desperdício de espaço é relativo ao grau médio de um nó. Quanto menor o grau, maior o desperdício. Além disso, é necessário descobrir o maior tamanho de palavra antes da construção do índice. Atualizações são permitidas, desde que as novas palavras não ultrapassem o tamanho definido.

3.2.2. Uso de Alocação Dinâmica

O uso de alocação dinâmica implica que o tamanho de cada vetor seja definido em tempo de execução, podendo ser realocados quando surgir a necessidade. Ao contrário da versão anterior, os vetores possuem tamanhos distintos, que coincidem com a maior distância necessária.

A Figura 2 (b) ilustra o uso de vetores dinâmicos para a mesma configuração de nós usada na Figura 2 (a). Observe que vetores dinâmicos acarretam em um menor consumo de memória, em comparação com os vetores estáticos. A diferença entre as duas abordagens reside na parte final dos vetores. Com alocação dinâmica, os espaços que sucedem a última posição preenchida não são sequer criados.

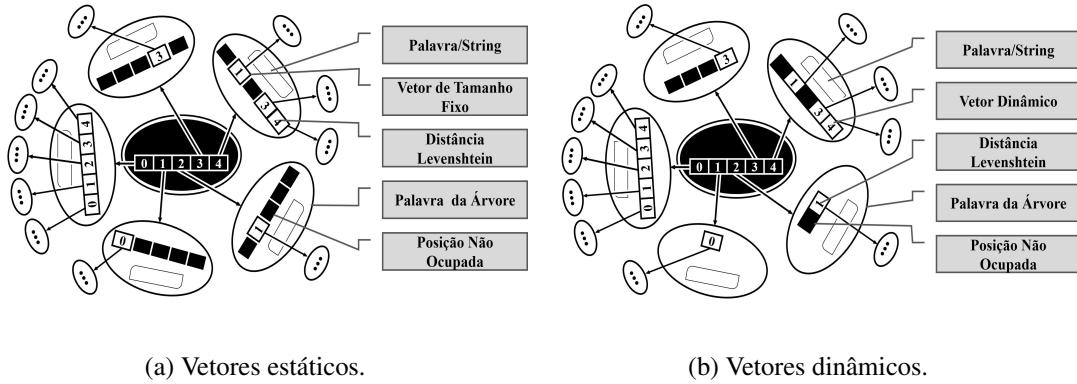


Figura 2. BK-tree implementada com vetores.

A busca nos filhos pode ser realizada pelo mesmo algoritmo empregado com vetores dinâmicos, desde que o valor máximo não ultrapasse o tamanho do vetor. Esse limite superior pode ser garantido pela inclusão de um comando condicional no código que altera *max_dist* caso haja necessidade.

Um aspecto negativo dos vetores dinâmicos é o maior custo para criação do índice, pois surge a sobrecarga relativa às realocações. Por outro lado, o custo é amortizado pela redução de espaço necessário, o que simplifica o gerenciamento de memória por parte do sistema operacional. Apesar da redução de espaço, ainda existe um desperdício associado às posições intermediárias do vetor não preenchidas. Não é possível abrir mão desses espaços intermediários não usados sem perder o acesso direto, razão principal para o uso de vetores.

3.3. Estratégia Baseada em Listas

Na estratégia baseada em listas, cada nó tem acesso ao primeiro filho. Além disso, nós são encadeados para que os irmãos sejam acessíveis. Um nó pai consegue acessar todos os filhos passando pelo primeiro filho e seguindo o encadeamento dos irmãos. Cada nó conta também com um valor numérico que simboliza a sua distância para o respectivo pai.

As variações dessa abordagem devem-se ao uso de encadeamento ordenado ou não. Em ambos os casos, a principal vantagem no uso de listas encadeadas é o menor consumo de memória. Já a desvantagem se refere a necessidade de caminhamento na lista para localizar um nó específico. As próximas seções discorrem a respeito dessas questões.

3.3.1. Uso de Listas Desordenadas

Com listas desordenadas, um nó filho é inserido sempre no começo da lista, independente do seu valor de distância. A Figura 3 (a) ilustra o uso de listas desordenadas. Nesse caso, o nó raiz possui filhos distantes dele por uma, três e quatro posições. Por sua vez, o filho cuja distância é um também possui uma descendência composta por quatro filhos. Como não se pode determinar a ordem com que os nós são inseridos na árvore, é natural que os valores de distância dos nós irmãos para o nó pai estejam dispostos aleatoriamente.

O Algoritmo 3.3 descreve como a busca ocorre usando listas desordenadas. A lista é percorrida do início ao fim para que os filhos sejam acessados. A função de busca por intervalo é chamada para todo nó acessado cuja distância satisfaça o critério. Como não existe ordenação, a pesquisa necessariamente visita todos nós da lista.

```
BUSCA_FILHOS(no_pai, consulta, min_dist, max_dist, palavras_retornadas)
1: no_filho ← no_pai.prim_filho
2: while no_filho ≠ Ø do
3:   if no_filho.d ∈ [min_dist, max_dist] then
4:     busca.Intervalo(no_filho, consulta, palavras_retornadas)
5:   end if
6:   no_filho ← no_filho.irmao
7: end while
```

Algorithm 3.3: BUSCAS DOS FILHOS DE UMA BK-TREE USANDO LISTAS DESORDENADAS.

Além do baixo consumo de memória, destaca-se como ponto positivo a eficiência na atualização. Como o nó a ser inserido substituirá o primeiro nó filho, a atualização da lista ocorre em tempo constante, sendo necessário apenas ajuste dos nós envolvidos. Outro ponto que merece destaque é o baixo consumo de memória. O aspecto negativo refere-se ao custo do percorrimento da lista. Como não existe ordenação, a pesquisa necessariamente visita todos nós de forma exaustiva. Quanto maior for o grau médio do nó, mais custosa se torna a busca.

3.3.2. Uso de Listas Ordenadas

Com listas ordenadas, um nó filho é inserido na lista na posição que mantenha os nós ordenados pela distância. A Figura 3 (b) ilustra o uso de listas ordenadas, usando os mesmos valores apresentados na Figura 3 (a). Independente da ordem com que os nós sejam inseridos na árvore, existe a garantia de que eles permanecem ordenados.

O algoritmo 3.4 descreve como a busca ocorre usando listas ordenadas. Assim como na versão anterior, a lista é percorrida para que os filhos sejam acessados, e a função de busca por intervalo é chamada para todo nó acessado cuja distância satisfaça o critério. No entanto, como a lista está ordenada, a busca pode ser encerrada quando se visitar algum nó que tenha ultrapassado a distância máxima permitida.

Uma vantagem desta versão é a redução no custo do percorrimento da lista. Enquanto na versão desordenada todos os n filhos precisam ser acessados, a ordenação reduz o número de acessos para $\frac{n}{2}$, na média. Em contrapartida, a atualização é mais cara. Em média $\frac{n}{2}$ nós precisam ser acessados até que o lugar correto seja encontrado, enquanto que na lista desordenada nenhum acesso extra é necessário.

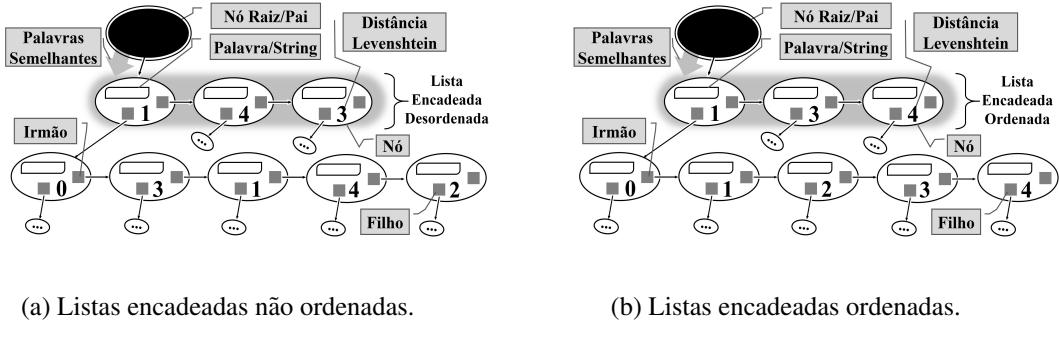


Figura 3. BK-tree implementada com listas encadeadas.

```
BUSCA_FILHOS(no_pai, consulta, min_dist, max_dist, palavras_retornadas)
1: no_filho ← no_pai.prim_filho
2: while no_filho <> Ø do
3:   if no_filho.d > [max_dist] then
4:     return
5:   end if
6:   if no_filho.d >= [min_dist] then
7:     busca_Intervalo(no_filho, consulta, palavras_retornadas)
8:   end if
9:   no_filho ← no_filho.irmao
10: end while
```

Algorithm 3.4: BUSCAS DOS FILHOS DE UMA BK-TREE USANDO LISTAS ENCADEADAS ORDENADAS.

4. Experimentos

O objetivo desta seção é avaliar as quatro variações de BK-trees apresentadas no decorrer do artigo. Os critérios de avaliação incluem o tempo de indexação, espaço ocupado e tempo de busca. Os tempos relatados correspondem a média de 30 execuções, descartando os 10% piores e 10% melhores resultados.

Por permitir um gerenciamento de memória mais ajustado, os algoritmos foram implementados em C. As execuções foram realizadas em uma máquina Core i5, com 6 GigaBytes de memória utilizando o Sistema Operacional aberto Lubuntu de 64 bits. O dicionário usado contém 994.703 palavras da língua portuguesa¹. A maior palavra tem 29 caracteres, e na média cada palavra tem 12 caracteres.

4.1. Tempo de Indexação

Não houve diferença significativa no tempo de indexação para as quatro variações. Em qualquer um dos casos, o tempo para construir as árvores ficou em torno de 28 segundos. O resultado demonstra que o cálculo da distância de edição, necessário para fazer o roteamento da palavra a ser inserida, tem um custo predominante.

O tempo necessário para localização dos nós (nas listas ordenadas) tem pouca importância devido ao baixo grau de saída de cada nó (média de 1 filho por nó). Como as listas tem poucos elementos, o custo para localização acaba sendo baixo.

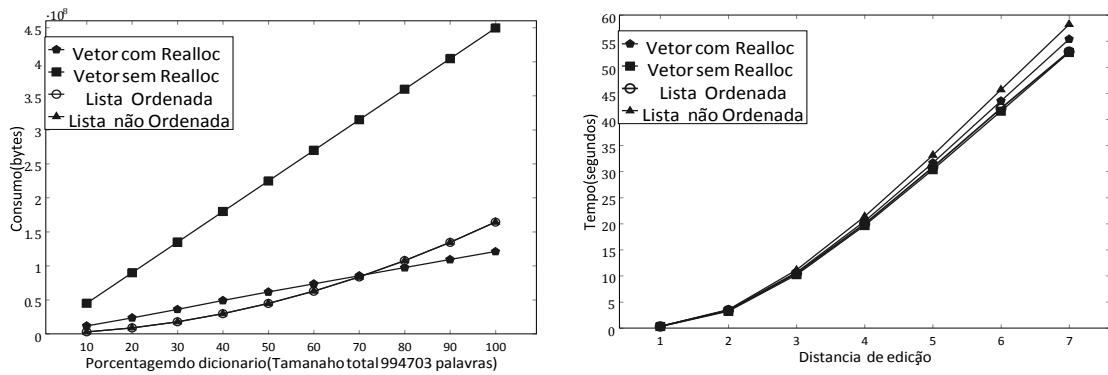
O custo da realocação (nos vetores dinâmicos) também se mostrou insignificante. No pior cenário, o sistema operacional precisaria encontrar espaço contíguo para apenas

¹Disponível em <http://natura.di.uminho.pt/download/sources/Dictionaries/wordlists/wordlist-ao-20170814.txt.xz>

29 ponteiros (que é a maior distância de edição possível), o que poderia ser feito sem grande esforço.

4.2. Espaço Ocupado

A Figura 4 (a) apresenta o espaço ocupado na memória em bytes que cada estrutura de árvore precisou para a indexação das 994.703 palavras. O gráfico mostra como o consumo é afetado quando partes maiores do dicionário são consideradas.



(a) Custo de Memória Variando o Tamanho do Dicionário.

(b) Tempo de Busca Variando a Distância Máxima Permitida.

Figura 4. Resultados.

Como era esperado, o uso de vetores estáticos leva ao pior resultado. A diferença que existe para as demais abordagens é um efeito direto da diferença entre o tamanho da maior palavra e o tamanho médio. Na verdade, as outras três estratégias alocam menos espaços do que esse tamanho médio de palavra. Por exemplo, os vetores dinâmicos alocam espaço suficiente para uma distância média igual a 4. Como a indexação aproxima palavras parecidas, a distância de um filho para o pai acaba sendo na maioria dos casos menor do que o tamanho médio de palavra.

Também é interessante notar que os vetores dinâmicos passam a ocupar menos memória que os concorrentes conforme o dicionário cresce. O motivo está relacionado aos buracos dos vetores dinâmicos, criados para os espaços intermediários não preenchidos. Para dicionários suficientemente grandes, menos espaços vazios são deixados, o que leva a um melhor aproveitamento da memória.

4.3. Tempo de Busca

A Figura 4 (b) apresenta o tempo em segundos necessário para responder 25 consultas, com distâncias máximas de edição variando de um a sete. As 25 consultas são palavras escolhidas aleatoriamente do dicionário.

Os resultados mostram que há pouco diferença entre os algoritmos, especialmente quando se usa distâncias pequenas. O motivo mais uma vez está atrelado ao peso que o cálculo da distância tem frente às demais operações necessárias. Apesar da diferença irrisória, os vetores estáticos e as listas ordenadas apresentaram os melhores resultados em todos os casos. O ganho se torna mais evidente conforme a distância

aumenta. De qualquer forma, a busca por intervalo geralmente encontra resultados satisfatórios usando distâncias menores do que 3. Por exemplo, com distância de edição igual a um já foi possível trazer uma média de 4 resultados para as 25 consultas usadas no experimento.

5. Considerações Finais

Este artigo mostrou as diferenças estruturais entre quatro variações do algoritmo de busca por similaridade BK-tree. Através de exemplos, mostrou-se as vantagens e desvantagens de cada abordagem, levando em consideração tempo de indexação, espaço ocupado em memória e tempo de busca.

Os experimentos realizados mostraram que o desempenho na carga e na busca(para distâncias pequenas) é muito parecido. Sendo assim, a escolha por uma abordagem pode levar em consideração o consumo de memória. Nesse quesito, vetores dinâmicos mostraram-se superiores. Os resultados também serviram para constatar que o principal custo tanto na busca quanto na indexação é o cálculo da distância de edição. O algoritmo baseado em programação dinâmica que encontra a distância para as palavras m e n tem complexidade em tempo proporcional a $O(m.n)$. Uma tema interessante para trabalhos futuros é a definição de outro tipo de distância de edição, tão relevante quanto à distância de Levenshtein, mas cuja complexidade em tempo seja menor.

Referências

- Baeza-Yates, R., Cunto, W., Manber, U., and Wu, S. (1994). Proximity matching using fixed-queries trees. In *Annual Symposium on Combinatorial Pattern Matching*, pages 198–212. Springer.
- Bozkaya, T. and Ozsoyoglu, M. (1997). Distance-based indexing for high-dimensional metric spaces. In *ACM SIGMOD Record*, volume 26, pages 357–368. ACM.
- Burkhard, W. A. and Keller, R. M. (1973). Some approaches to best-match file searching. *Communications of the ACM*, 16(4):230–236.
- Chávez, E. and Navarro, G. (2003). Probabilistic proximity search: Fighting the curse of dimensionality in metric spaces. *Information Processing Letters*, 85(1):39–46.
- Chen, L., Gao, Y., Li, X., Jensen, C. S., and Chen, G. (2015). Efficient metric indexing for similarity search. In *Data Engineering (ICDE), 2015 IEEE 31st International Conference on*, pages 591–602. IEEE.
- Chen, L., Gao, Y., Zheng, B., Jensen, C. S., Yang, H., and Yang, K. (2017). Pivot-based metric indexing. *Proceedings of the VLDB Endowment*, 10(10):1058–1069.
- Navarro, G. (2001). A guided tour to approximate string matching. *ACM computing surveys (CSUR)*, 33(1):31–88.
- Yianilos, P. N. (1993). Data structures and algorithms for nearest neighbor search in general metric spaces. In *SODA*, volume 93, pages 311–321.
- Zezula, P., Amato, G., Dohnal, V., and Batko, M. (2006). *Similarity search: the metric space approach*, volume 32. Springer Science & Business Media.

Predição do volume de atendimentos de saúde na cidade de Curitiba utilizando dados abertos

Mayara Regina Lorenzi¹, Cristiano da Cunha Ribas¹, Luiz Gomes Jr.¹

¹Departamento Acadêmico de Informática – Universidade Tecnológica Federal do Paraná (UTFPR) – Curitiba, PR - Brasil

may.lorenzi@gmail.com, cristianorbs@gmail.com,
gomesjr@dainf.ct.utfpr.edu.br

Abstract. This paper explores the prediction of total demand of nursing service on public hospitals and walkin clinics of the city of Curitiba. The analysis is based on data mining techniques applied to open data provided by the municipality and weather data by Weather Underground. We present here an exploratory analysis and the implementation of a preliminary model of linear regression to estimate the variation of the demand for healthcare service related to respiratory ailments.

Resumo. Este trabalho explora a predição de volume de atendimentos de doenças respiratórias na rede pública da cidade de Curitiba. A análise é baseada em métodos de mineração de dados aplicados em dados abertos de atendimento fornecidos pelo município e dados climáticos fornecidos pelo portal Weather Underground. Apresentamos aqui uma análise exploratória dos dados e a implementação de um modelo preliminar de regressão linear para estimativa de variação no volume de atendimentos referentes a doenças respiratórias.

1. Introdução

As variações metereológicas e climáticas podem impactar diretamente na saúde da população. Para as doenças do trato respiratório, especula-se que a condição climática do local pode interferir no volume de pessoas afetadas. Em períodos de chuva ou com mudanças bruscas de temperatura, por exemplo, pode ocorrer maior propensão a doenças virais, como a gripe [Viveiros, 2014].

Além disso, a gestão pública de saúde não possui insumos necessários para o planejamento a médio e longo prazo de escala de médicos e enfermeiros, compra de itens medicamentosos e de uso clínico.

Isso se deve ao fato de que o volume de pacientes que necessitam de consultas e de internamentos depende de diversos fatores, como período do ano, condições meteorológicas, qualidade de vida da população, entre outros.

Nos anos 2000, foi lançado o conceito de Cidades Inteligentes e está em crescente debate político e acadêmico. O tema é abordado para se referir a cidades que fazem uso da tecnologia para aprimorar o processo de planejamento, a fim de melhorar a sustentabilidade do local e encontrar soluções para a sociedade e para o Estado.

Nesse contexto, a Tecnologia da Informação, através de aprendizado de máquina e análise exploratória de dados, visa maximizar a obtenção de informações ocultas em um grande volume de dados, descobrir variáveis importantes nas tendências e detectar comportamentos anômalos.

Esse projeto tem como objetivo criar, a partir de dados reais fornecidos pelos órgãos responsáveis, artefatos computacionais para apoio à previsão de volume de pacientes com sintomas de doenças respiratórias que necessitarão de atendimento médico na cidade de Curitiba, a fim de resolver parte do problema de escala de médicos e enfermeiros e a compra de medicamentos.

Foram utilizados dados relacionados aos atendimentos médicos e de enfermagem disponibilizados pela Secretaria de Saúde da Prefeitura de Curitiba. Aplicando técnicas de análise exploratória e algoritmos de aprendizado de máquina nos dados fornecidos, juntamente com dados de informações climáticas, pôde-se obter variáveis importantes e identificar tendências de comportamento.

2. Trabalhos Correlatos

Nos anos 2000, um estudo realizado pelo Departamento de Saúde da Prefeitura de Curitiba mostrou que as doenças do aparelho respiratório constituíram o motivo mais frequente de consulta (19,6%) e quarta causa de morte em todas as faixas etárias da população do município. Na faixa etária pediátrica a proporção se torna mais extrema, representando 50% das consultas ambulatoriais e aproximadamente 25% dos internamentos em menores de 14 anos [Prefeitura de Curitiba, 2010].

Pesquisas realizadas na Filadélfia [Hollemand, 1996] estudaram as associações entre variáveis climáticas e volume de consultas agendadas para o período em clínicas de emergência e pronto atendimento. A análise levou em consideração, além dos fatores mencionados, a estação do ano, o dia da semana e as vésperas de feriado.

Foi identificado um alto volume de pacientes durante os meses de inverno, com exceção de dezembro. Esse fato pode ser devido à estação ser facilitadora para aumentar as doenças pulmonares, dada a exposição ao frio. E, se tratando de agendamento de consultas, o número pode ser reduzido em dezembro devido aos feriados e férias.

Um estudo realizado pela Universidade Federal de Santa Maria [Gonçalves, 2010] analisou a variação da morbidade de doenças respiratórias em função da variação da temperatura entre os meses de abril e maio em São Paulo. A pesquisa demonstrou que pode haver relação entre a temperatura mínima e doenças respiratórias na população infantil. Há um pico de morbidade por doenças respiratórias das vias superiores no mês de maio, devido ao problema de termo-regulação em indivíduos adaptados ao clima mais ameno. Esta tendência de aumento da diferença pode aumentar a ida aos hospitais no mês de maio, gerando impacto em hospitais e em políticas públicas.

3. Descrição dos dados utilizados

A. Dados de atendimentos hospitalares

Os dados de atendimento hospitalar foram disponibilizados pela Prefeitura de Curitiba através da plataforma de dados abertos da Universidade Federal do Paraná¹.

A base de dados é oriunda do sistema “E-saúde”, que mantém as informações de atendimentos de enfermagem e médicos prestados pela Secretaria Municipal de Saúde de Curitiba em sua rede de atenção, que é composta por Unidades Básicas de Saúde, Unidades de Pronto Atendimento e Centros de Especialidades Médicas e Odontológicas.

A base inclui diversas informações sobre os atendimentos e pacientes. As utilizadas para esse projeto são:

- Código do CID-10, que é o código do diagnóstico do paciente. A CID (Classificação Estatística Internacional de Doenças e Problemas Relacionados com a Saúde) é publicada pela Organização Mundial de Saúde (OMS) e fornece códigos relativos à classificação de doenças utilizados globalmente para estatística de morbidade e mortalidade.
- Data de nascimento e sexo do paciente, utilizados nesse projeto para aprofundamento dos resultados.
- Desencadeou Internamento, que indica se o paciente foi encaminhado para internamento após a consulta.

A área de interesse desse projeto se refere às doenças catalogadas no CID-10 como Doenças do Aparelho Respiratório (J00-J99), que podem ser desdobradas de acordo com os grupos de doenças destinado para cada código do CID-10 abordado.

O total de atendimentos de doenças do aparelho respiratório representaram, no período de junho a agosto de 2016, o principal motivo de consultas médicas (17,9%) e o segundo cenário em número de internamentos (15,6%), para todas as faixas etárias. Para a faixa etária pediátrica (até 14 anos), o número é ainda maior, sendo 37,6% no número de consultas e 46,9% em internamentos.

B. Dados climáticos

Os dados metereológicos utilizados foram disponibilizados pela The Weather Company, através da Weather Underground, que fornece uma API gratuita para desenvolvedores.

O ponto de referência para esse estudo é o bairro Centro da cidade de Curitiba. As variáveis climáticas utilizadas nessa pesquisa são a temperatura média, umidade relativa do ar média do dia e a amplitude térmica, as demais nove características climáticas, como velocidade do vento, foram desprezadas.

C. Remoção de anomalias

Esse passo consistiu na eliminação de 31 colunas, que representam variáveis que não foram utilizadas nesse estudo, como localização da unidade de atendimento e dados pessoais dos pacientes.

Após a eliminação das colunas desnecessárias, foram excluídos os registros duplicados e com preenchimento incompleto – sem data de atendimento e sem código CID-10. Além disso, foram calculadas as idades dos pacientes, gerando uma nova coluna à tabela, para que a análise fosse aprofundada nesse nível.

Os arquivos de entrada dessa etapa são divididos por período trimestral e totalizam 2,81GB de dados, com 7.757.527 registros. A manipulação desses dados, que durou aproximadamente 3,5 minutos, resultou em 301.915 registros, o que representa 3,9% da amostra inicial. Dessa forma, a execução das próximas etapas do processo de análise exploratória foi otimizada.

D. Agregação dos dados

Os dados metereológicos foram agrupados aos registros de atendimento, utilizando-se a data de atendimento como parâmetro de junção e de agregação.

Além das variáveis mencionadas acima, também foram calculadas as médias de temperatura, umidade e amplitude térmica dos últimos sete dias e armazenadas em um novo campo. Com essas variáveis, pretendeu-se afirmar a hipótese de que os atendimentos não variam conforme as características do dia em questão ou do dia anterior, e sim, com a influência das características dos últimos dias.

Após a agregação dos valores, os dados foram separados pelos códigos CID-10 que se iniciam com J, o que indica que o paciente foi diagnosticado com uma doença respiratória. Obteve-se, assim, duas bases de dados para fins comparativos: total de atendimentos e somente atendimentos de doenças respiratórias.

E. Normalização dos dados

O processo utilizado foi o de normalização por desvio padrão, que constitui-se em calcular a média do volume de atendimento de cada dia da semana, subtrair do total de cada dia e dividir o resultado dessa operação pelo desvio, como mostra a equação (1).

$$x' = \frac{x - x_m}{\sigma} \quad (1)$$

Onde, x é o volume de atendimento, x_m é a média de atendimento do dia da semana em questão, σ é o desvio padrão e x' é o valor resultante da normalização, que será considerado para aquele dia. Dessa forma, ao subtrair a média e dividir pelo desvio padrão, manteve-se apenas a variação de cada dia.

Com a normalização, pretendeu-se eliminar a possibilidade dos resultados serem influenciados pelos dias da semana e, sendo assim, a escala foi alterada.

4. Metodologia

A. Pré-processamento de dados

O pré-processamento de dados consistiu na execução de três etapas:

1) Remoção de anomalias

Foi observado um fenômeno em que nas segundas e terças-feiras o volume de atendimentos aumentava significativamente. Após análise, identificou-se que grande parte desses atendimentos eram de consultas que não possuíam CID-10, ou seja, eram de atendimentos a pessoas que não estavam doentes.

Além disso, foram removidos dados duplicados, visto que muitas vezes eram cadastrados dados sobrepostos, duplicando ou até triplicando o mesmo registro.

Também foram removidos registros com formato de data inválido. Esse efeito pode ser visto pelo fato de os dados serem cadastrados manualmente e, não necessariamente, seguindo um padrão.

2) Integração dos dados

Foram agrupados os dados climáticos aos dados de atendimento a partir da coluna de data, ou seja, foi feita a junção dos registros climáticos e de atendimento quando possuíam a mesma data de ocorrência.

3) Transformação de dados

Foi feita a alteração em alguns formatos de datas, pois nem todos seguiam o mesmo padrão. Além disso foram adicionadas algumas novas colunas calculadas, como idade, temperatura média, umidade média e amplitude térmica média da última semana.

B. Regressão Linear

A partir das características selecionadas (temperatura média, umidade média e amplitude térmica média da última semana), decidiu-se criar um modelo de regressão linear com o objetivo de tentar estimar o volume de consultas de um ou mais dias posteriores baseado nas variáveis climáticas dos dias anteriores.

Primeiramente foi feita a divisão de 80% dos dados para treinamento e 20% para a validação. Esses dados foram selecionados de maneira aleatória.

O modelo foi ajustado a partir dos dados de treinamento e validado utilizando os dados de teste.

Por fim foi calculado o erro quadrado mínimo dos valores previstos, bem como o coeficiente de determinação e o valor p de cada característica para avaliar quão bem o modelo escolhido pode ou não prever o total de atendimentos futuro.

5. Resultados

Para a criação do modelo foi, primeiramente, calculada a similaridade do cosseno centralizado, ou coeficiente de correlação de Pearson, entre as características e o alvo (total de atendimentos normalizado). Tais coeficientes são descritos na tabela 1 e são utilizados para dizer se há uma possibilidade de existir uma relação linear ou não entre duas variáveis.

TABELA 1. COEFICIENTES DE CORRELAÇÃO DE PEARSON ENTRE AS CARACTERÍSTICAS E O ALVO

	Total Normalizado
Temperatura mínima do dia	-0,176277
Média de temperatura dos últimos 7 dias	-0,289704
Média de umidade dos últimos 7 dias	-0,058195
Média da amplitude térmica dos últimos 7 dias	0,038121

Os coeficientes negativos indicam que as variáveis tendem a descrecer. Nesse caso seria a indicação, por exemplo, de que quando a temperatura mínima diminui, o total normalizado aumenta.

Segundo [Evans, 1996], a correlação do valor mais significativo encontrado (-0,289704) é considerada fraca, mas dada a complexidade do problema proposto, as correlações foram consideradas razoáveis para continuar as análises.

Para uma análise visual, foi traçado o gráfico da figura 1, representando os valores total de atendimentos comparados com a temperatura média dos últimos 7 dias referente ao dia em questão, que foi a característica o coeficiente de correlação mais representativo. Além disso foi traçada a reta que melhor se ajustou nos pontos representados no gráfico.

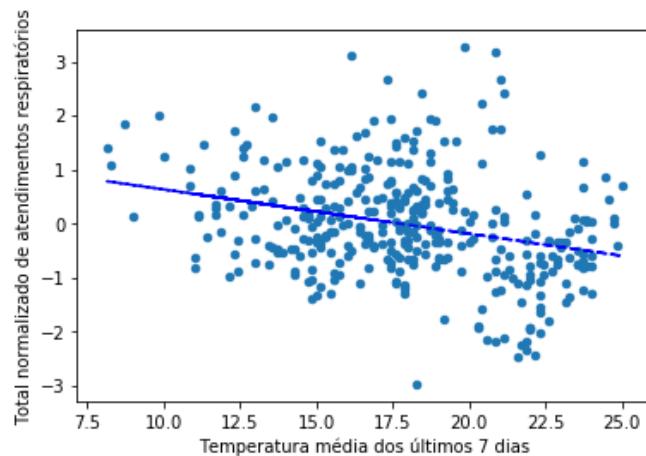


Figura 1. Temperatura média da última semana pelo total normalizado de atendimentos de doenças respiratórias

Pode-se notar que a reta representada na figura 1 tende a decrescer. Apesar de ter uma inclinação leve, pode ser um indicativo de o total de atendimentos aumentar à medida que a temperatura média da semana anterior diminui.

Com a separação dos dados de treinamento e de teste, foram utilizados os 80% referentes aos dados de treinamento para criar o modelo utilizando o método dos mínimos quadrados ordinários.

Os valores-p, ou probabilidades de significância, de cada variável são apresentados na tabela 2, bem como os coeficientes e o erro padrão de cada característica no modelo.

TABELA 2: VALORES-P, COEFICIENTES E ERROS PADRÃO DAS CARACTERÍSTICA DO MODELO

	Valor-p	Coeficiente	Erro padrão
Temperatura mínima do dia	0,149	0,0667	0,046
Média de temperatura dos últimos 7 dias	0,0388	-0,114	0,054
Média de umidade dos últimos 7 dias	0,1309	0,0543	0,036
Média da amplitude térmica dos últimos 7 dias	0,1076	0,1877	0,115

Considerando um nível de significância de 5%, somente seria rejeitada a hipótese nula utilizando a característica de média de temperatura da última semana. Além disso, o coeficiente dessa característica é de -0,114, o coeficiente com maior significância se for

considerado o erro padrão, visto que a característica de média da amplitude térmica da última semana é mais alta, porém o valor-p acima de 5% não rejeitaria a hipótese nula e o erro padrão seria de 11,5%.

O coeficiente de determinação, também chamado de R^2 , encontrado foi de 0,104. Isso significa que 10,4% da variável dependente (total normalizado de atendimentos) consegue ser explicado pelo modelo criado a partir das outras características.

Então, foi utilizado o modelo para tentar prever os dados de teste e validar o modelo. O gráfico da imagem 2 mostra a comparação entre os valores reais e os previstos.

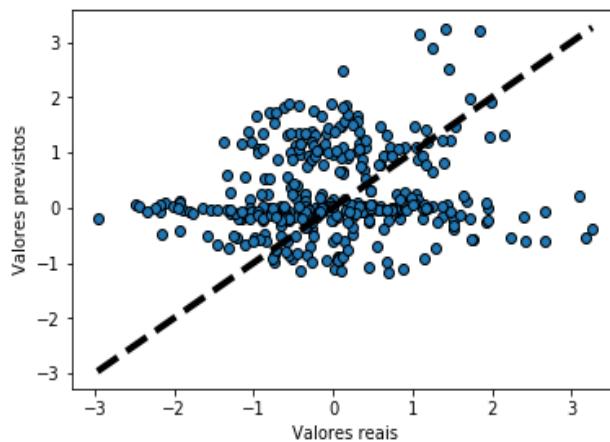


Figura 2: Comparação entre valores previstos e valores reais do total de atendimentos normalizado

A linha tracejada representa a posição ideal dos pontos caso a predição fosse 100% assertiva e foi inserida para melhorar a visualização da distribuição dos pontos.

6. Conclusão

O modelo desenvolvido foi capaz de capturar cerca de 10% da viariação de atendimentos por doenças respiratórias. Considerando que um dia típico tem média de 350 atendimentos em Curitiba, o modelo é capaz de prever cerca de 10 casos para mais ou para menos. Consideramos o resultado relevante para este modelo inicial desenvolvido, sobretudo considerando-se a complexidade do problema.

Diversas variáveis não foram consideradas neste modelo inicial e pretendemos empregá-las em futuros refinamentos do modelo. Por exemplo, seria importante tratar a agregação da temperatura em janelas diferentes (usar 7 dias foi uma decisão arbitrária e de relevância não avaliada). Também pretendemos agregar informações sobre poluição do ar e tratar com maior granularidade a idade e diagnóstico do paciente. Também pretendemos estender o período de dados coletados.

Infelizmente, diversas variáveis importantes associadas ao problema são difíceis de se caracterizar, como por exemplo as variantes dos vírus presentes em um determinado período. De acordo com mutações genéticas imprevisíveis, varia-se de ano a ano a taxa de transmissibilidade, gravidade de sintomas e época de disseminação dos vírus.

A despeito da complexidade do problema, acreditamos que os resultados obtidos são promissores e já poderiam ser considerados para o planejamento de atendimentos. Esperamos também aumentar significativamente a precisão do modelo a partir dos refinamentos previstos.

7. Referências

- B VIVEIROS, JOSÉ. Universidade de Coimbra: A Influência das Alterações Climáticas nas Patologias Respiratórias. Disponível em <<http://estudogeral.sib.uc.pt/bitstream/10316/29245/1/A%20Influ%C3%A7%C3%A1ncia%20das%20Altera%C3%A7%C3%A7%C3%B5es%20Clim%C3%A1ticas%20nas%20fazendo0Patologias%20Respirat%C3%B3rias.pdf>>
- CONNECTED SMART CITIES. O que é uma cidade inteligente? Disponível em <<http://www.connectedsmartcities.com.br/index.php/afinal-o-que-e-uma-cidade-inteligente>>
- SILVA, BRIGIANE. VANDERLINE, MARCOS. Inteligência Artificial, Aprendizado de Máquina. Disponível em <http://www.ceavi.udesc.br/arquivos/id_submenu/387/brigiane_machado_da_silva__marcos_vanderlinde.pdf>
- PREFEITURA DE CURITIBA. Infecções e Doenças Respiratórias. Disponível em <<http://www.saude.curitiba.pr.gov.br/index.php/programas/saude-da-crianca/infeccoes-e-alergias-respiratorias>>
- HOLLEMAN, DONALD. BOWLING, RENEE. GATHY, CHARLANE. Predicting Daily Visits to a Walk-in Clinic and Emergency Department Using Calendar and Weather Data. Disponível em <https://www.researchgate.net/publication/14457443_Predicting_daily_visits_to_a_walk-in_clinic_and_emergency_department_using_calendar_and_weather_data>
- GONÇALVES, FÁBIO. COELHO, MICHELINE. Universidade Federal de Santa Maria: Variação da morbidade de doenças respiratórias em função da variação da temperatura entre os meses de abril e maio em São Paulo. Disponível em <<https://periodicos.ufsm.br/cienciaenatura/article/viewFile/9500/5649>>
- Evans, J. D. Straightforward statistics for the behavioral sciences. Pacific Grove, CA: Brooks/Cole Publishing, 1996.

Predição de Indicadores Zootécnicos de Carcaças Bovinas a Partir de Variáveis de Cria

Thales V. Maciel¹, Vinícius do N. Lampert², Denizar S. Souza³, Rodrigo R. da Silva¹

¹Instituto Federal de Educação, Ciência e Tecnologia Sul-rio-grandense (IFSUL)
Campus Bagé – Av. Leonel de Moura Brizola, 2501 – 96.418-400 – Bagé – RS – Brasil

²Empresa Brasileira de Pesquisa Agropecuária (EMBRAPA)
Unidade Pecuária Sul – Bagé – RS – Brasil

³Centro de Ciências Exatas e Aplicadas (CCEA)
Universidade da Região da Campanha (URCAMP) – Bagé, RS – Brasil

thalesmaciel@ifsul.edu.br, vinicius.lampert@embrapa.br,
denizarSouza@urcamp.edu.br, orki2008@gmail.com

Abstract. This paper describes a method for obtaining decision trees for predicting carcasse zootechnical quality indicators for bovine based on their breeding data. For such, data mining classification tasks were performed after data preprocessing. All numeric attributes were discretized by non-equal frequency binning or by cluster discovery in distinct classification experiments. Obtained results showed that clustering techniques as means for discretization may generate classes in better balancing conjecture when in comparison to the non-equal frequency binning method, allowing the discovery of models that may be applied to real world problems.

Resumo. Este artigo descreve uma metodologia para obtenção de árvores de decisão para previsão de indicadores zootécnicos de qualidade de carcaças bovinas com base em variáveis de cria dos animais. Para tal, procedeu-se a tarefas de mineração de dados com classificação após pré-processamento com discretização dos atributos numéricos por particionamento igualitário do intervalo ou por descoberta de agrupamentos em experimentos distintos de classificação. Os resultados obtidos mostraram que a descoberta de agrupamentos como forma de discretização pode gerar classes com balanceamento de melhor qualidade em comparação ao método tradicional, permitindo a indução de modelos utilizáveis em problemas reais.

1. Introdução

O sistema de produção de gado de corte é o conjunto de tecnologias e práticas de manejo, tipo de animal, propósito de criação, raça e ecorregião onde a atividade é desenvolvida [Euclides Filho 2000]. Compreende uma das principais atividades de exploração econômica no Brasil, onde, há décadas, tem-se afastado o cenário de resistência ao emprego tecnológico, de modo a permitir estudos para o melhoramento dos índices de qualidade na produção de carne, por exemplo, através de computação aplicada [Barbosa 1999].

Em [Costa 2016], foram analisados dados zootécnicos de 401 animais bovinos da raça Hereford com vistas em prever o peso de fazenda e bonificação dos indivíduos. No estudo, foram empregadas redes neurais artificiais como ferramenta para o processo

de descoberta de conhecimento em experimentos distintos para as duas variáveis. Todos os dados envolvidos foram do tipo numérico. Segundo o autor, o trabalho foi concluído com resultados satisfatórios na previsão do peso de fazenda, mas insatisfatórios na previsão da bonificação, atribuindo o não cumprimento do objetivo específico à má qualidade de dados, uma característica não observada no primeiro experimento. O estudo não considerou a praticidade da utilização de redes neurais artificiais pelos produtores pecuários em meio às tarefas cotidianas, tampouco apresentou comparações com outros métodos para descoberta de conhecimento em bancos de dados.

Em [Da Mota et al. 2017], foram empregadas tecnologias de armazém de dados, consultas analíticas e mineração de dados para 1142230 registros de abates bovinos. O objetivo foi o de prever o grau de acabamento e o rendimento das carcaças, em experimentos individuais, que foram conduzidos com algoritmos de classificação e redes neurais artificiais. Os resultados, segundo os autores, foram promissores, devido às acuráncias alcançadas nos experimentos, cuja média em acertos de classificação foi de 62%. Embora tenham composto médias de acuráncias da aplicação de diferentes algoritmos nas tarefas preditivas, os autores falharam em apresentar uma comparação das acuráncias dos algoritmos utilizados individualmente, bem como avaliações mais aprofundadas dos resultados, que fossem além das acuráncias observadas nos experimentos e apresentar os modelos gerados pelos mesmos e que são passíveis desta análise.

Nota-se que trabalhos correlatos publicados recentemente, mesmo que parcialmente eficazes segundo os respectivos autores, não explicam as previsões realizadas pelos experimentos que documentam, ou pela impossibilidade disto ser característica do algoritmo empregado (caixa-preta) ou por não apresentar a totalidade dos resultados da classificação nos resultados obtidos nos testes (matrizes de confusão, por exemplo).

O problema de pesquisa abordado no presente estudo é fundamentado em “quais variáveis podem ser coletadas, pelos criadores, sobre os indivíduos de rebanhos bovinos em etapa de desenvolvimento de cria e que explicam a obtenção de indicadores de qualidade zootécnicos das carcaças ótimos após o abate?”

A hipótese trabalhada é que existe uma relação estatística entre o mês de nascimento, o mês de desmame, a idade de desmame e o peso de desmame com o peso e a idade de abate. Também são consideradas a influência das variáveis citadas sobre o ganho médio diário de peso (GMD) dos animais e a bonificação recebida pelas carcaças após o abate.

O objetivo é obter um modelo gráfico, de fácil interpretação, capaz de orientar os criadores bovinos sobre o desempenho de seus rebanhos, ainda em etapa anterior ao desmame, com previsões dos futuros índices de qualidade que serão obtidos após o abate dos animais.

2. Metodologia

Procedeu-se à descoberta de conhecimento em bancos de dados (DCBD), especificamente com as tarefas de mineração de dados descritas nesta seção.

O processo de DCBD pode ser dividido em três etapas [Maciel et al. 2015]: o pré-processamento, onde o conjunto de dados original é preparado para as próximas etapas do processo através de tarefas de filtragem conforme necessário; o processamento, onde algoritmos de mineração de dados são aplicados sobre o conjunto

de dados pré-processado e; o pós-processamento, onde os padrões descobertos no processamento são analisados e transformados em conhecimento útil sobre o domínio estudado.

Para fins de realização das tarefas e experimentos descritos neste estudo, foi empregado o Waikato Environment for Knowledge Analysis (WEKA), um ambiente para análise de conhecimento desenvolvido pela Universidade de Waikato [Hall et al. 2009].

O WEKA é uma coleção de algoritmos que podem ser utilizados em atividades de mineração de dados diversas, como classificação, regressão, associação e clustering, além de diversos métodos de pré-processamento e visualização de resultados através de interface gráfica, linha de comando ou interface de programação [Witten et al. 2017].

O conjunto de dados analisado teve sua apresentação original em 167 instâncias de animais bovinos e 8 atributos, conforme descrição na Tabela 1.

Tabela 1. Descrição do conjunto de dados analisado

#	Nome do Atributo	Significado	Tipo de Dado
1	nascimento_mes	mês de nascimento (01-12)	nominal
2	desmame_mes	mês de desmame (01-12)	nominal
3	desmame_idade	idade de desmame em meses	numérico
4	desmame_peso	peso de desmame em quilogramas	numérico
5	abate_idade	idade de abate em meses	numérico
6	abate_peso	peso de abate em quilogramas	numérico
7	gmd	ganho médio diário de peso	numérico
8	bonificacao	percentual de bonificação da carcaça após abate	numérico

No preprocessamento, os atributos numéricos foram discretizados, conforme a escala de Likert [Likert 1932], de forma a criar segmentos nominais dentre o intervalo numérico com as denominações: muito baixo, baixo, intermediário, alto e muito alto.

Discretização é o particionamento de um intervalo numérico e sucessiva atribuição de um valor categórico como rótulo de cada partição criada [Witten et al. 2017]. No âmbito deste estudo, dois métodos de discretização distintos foram experimentados:

- Discretização dos atributos em 5 segmentos sem balanceamento de peso entre eles, apenas dividindo o intervalo numérico de cada atributo em 5 frações iguais. Cada fração foi tornada em uma classe;
- Descoberta automatizada de 5 agrupamentos unidimensionais (sobre cada atributo numérico) com base na distância de Manhattan, em aplicação do algoritmo Simple k-means [Arthur e Vassilvitskii 2007]. Cada agrupamento descoberto foi tornado em uma classe.

As Figuras 2 e 3 respectivamente apresentam os histogramas referentes às distribuições de frequência das instâncias de bovinos nas categorias propostas pela escala de Likert nos atributos discretizados pelos métodos do fracionamento igualitário do intervalo numérico e com a descoberta automatizada dos agrupamentos baseados na distância de Manhattan.

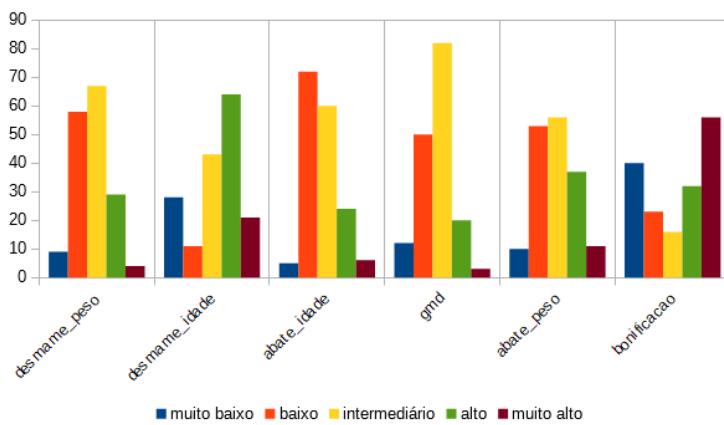


Figura 1. Histogramas referentes aos atributos discretizados por segmentação

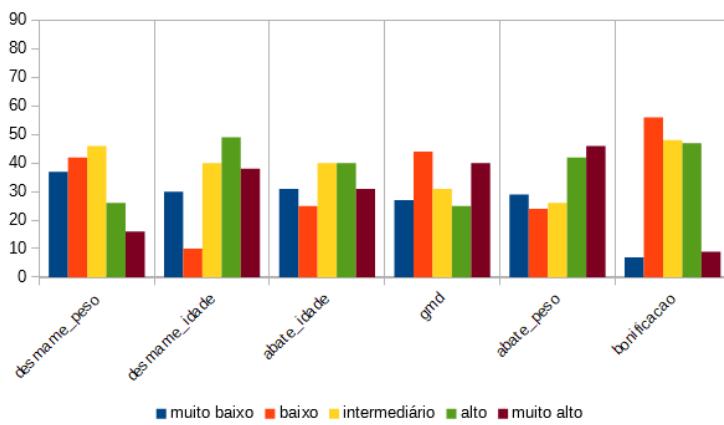


Figura 2. Histogramas referentes aos atributos discretizados por descoberta de agrupamentos.

No total, foram realizados 8 experimentos de predição. Neles, as variáveis de cria (mês de nascimento, mês de desmame, peso de desmame e idade de desmame) foram utilizadas para prever os indicadores zootécnicos de qualidade das carcaças (idade de abate, peso de abate, GMD e bonificação) em experimentos distintos.

Cada indicador zootécnico de qualidade das carcaças foi alvo de predição após discretização dos atributos numéricos através dos dois métodos apresentados. A Tabela 2 elenca as atividades de pré-processamento pelas quais cada atributo foi submetido em cada experimento. Os atributos cujo tipo de dado original é o nominal não passaram pois quaisquer tarefas de pré-processamento (N/A). Os atributos alvo da predição em cada experimento são destacados em sublinhado na Tabela 2.

A mineração de dados é a tarefa de identificação de padrões a partir de dados, de forma automatizada em ambiente computacional, que compreende a etapa de processamento no processo de DCBD [Maciel et al. 2015].

A classificação é um tipo de tarefa da mineração de dados que visa categorizar instâncias supostamente novas com base na análise de dados de instâncias pregressas [Witten et al. 2017]. Há a etapa de treinamento, onde um algoritmo aprende as características inerentes à cada classe e a etapa de teste, onde é verificada a acurácia do modelo criado.

Árvores de decisão são um tipo de modelo de dados utilizado como resultado de tarefas de classificação [Quinlan 1993] e apreciado no contexto deste estudo em virtude de sua simplicidade e interpretabilidade.

Tabela 2. Descrição das tarefas de pré-processamento aplicadas a cada atributo do conjunto de dados nos experimentos realizados.

#	nascimento_mes	desmame_mes	desmame_peso	desmame_idade	abate_idade	abate_peso	gmd	bonificacao
1	N/A	N/A	segmentado	segmentado	<u>segmentado</u>	removido	removido	removido
2	N/A	N/A	segmentado	segmentado	removido	<u>segmentado</u>	removido	removido
3	N/A	N/A	segmentado	segmentado	removido	removido	<u>segmentado</u>	removido
4	N/A	N/A	segmentado	segmentado	removido	removido	removido	<u>segmentado</u>
5	N/A	N/A	agrupado	agrupado	<u>agrupado</u>	removido	removido	removido
6	N/A	N/A	agrupado	agrupado	removido	<u>agrupado</u>	removido	removido
7	N/A	N/A	agrupado	agrupado	removido	removido	<u>agrupado</u>	removido
8	N/A	N/A	agrupado	agrupado	removido	removido	removido	<u>agrupado</u>

O J48 [Hall et al. 2009] é um algoritmo de mineração de dados, especificamente para tarefas de classificação, capaz de induzir árvores de decisão, sendo um dos algoritmos mais utilizados em aplicações do tipo no mundo real. Trata-se da implementação em Java da 8ª revisão [Quinlan 1996], do algoritmo C4.5 [Quinlan 1993], originalmente documentado em linguagem C.

A etapa de processamento em todos experimentos foi realizada com o algoritmo J48 configurado para permitir apenas divisões binárias em galhos formados por atributos nominais e desconsiderar limites inferiores de ocorrências de instâncias em folhas para critérios de poda em seu treinamento. Os demais parâmetros do algoritmo foram mantidos em conformação padrão.

Os testes dos modelos descobertos foram realizados sobre os mesmos conjuntos de dados de entrada para as respectivas etapas de treinamento. Embora este não seja apreciado como o mais adequado método de testes em aplicações no mundo real [Witten et al. 2017], é possível observar nas Figuras 1 e 2 que muitas classes apresentam representatividade baixa dentre as 167 instâncias analisadas. Por tal motivo, é inviabilizada a realização de testes por validação cruzada com 5 frações, por exemplo.

3. Resultados Obtidos

Os resultados obtidos nas tarefas de classificação descritas na Seção 2 foram apresentados na forma de árvores de decisão, matrizes de confusão e acurácia dos respectivos modelos. Também é discutida a praticidade dos modelos descobertos.

As acurárias alcançadas em todos experimentos realizados neste estudo estão dispostas na Tabela 3.

Tabela 3. Acurárias resultantes dos experimentos realizados

Experimento	#1 (%)	#2 (%)	#3 (%)	#4 (%)	#5 (%)	#6 (%)	#7 (%)	#8 (%)
Acurácia	63,47	49,70	61,68	53,29	53,29	51,50	53,89	55,09

Foi feita comparação dos resultados obtidos na classificação com os conjuntos de dados cujos atributos numéricos foram discretizados por segmentação igualitária dos intervalos numéricos os conjuntos de dados cujos atributos numéricos foram discretizados pela descoberta automatizada de agrupamentos por aplicação do algoritmo Simple k-means. Foram analisadas as acurárias e matrizes de confusão resultantes dos

experimentos, onde foram evidenciadas falhas cruciais em alguns dos modelos gerados. A Tabela 4 apresenta as matrizes de confusão encontradas nos testes.

Matrizes de confusão são representadas em forma de tabela, onde as linhas representam as classes verdadeiras para cada instância e as colunas representam as classes onde cada instância foi classificada pelo modelo [Witten et al. 2017]. Desta forma, é possível visualizar a quantificação consolidada dos acertos e erros para cada classe nos experimentos.

Tabela 4. Matrizes de confusão resultantes dos experimentos realizados

		abate_idade					abate_peso					gmd					bonificação				
		mb	b	i	a	ma	mb	b	i	a	ma	mb	b	i	a	ma	mb	b	i	a	ma
classe verdadeira \ prevista																					
muito baixo (mb)		0	5	0	0	0	0	3	6	1	0	1	7	4	0	0	15	1	1	11	12
baixo (b)		0	57	11	4	0	0	25	22	6	0	0	28	22	0	0	3	3	2	4	11
intermediário (i)		0	20	32	8	0	0	9	43	4	0	0	13	62	7	0	1	0	5	4	6
alto (a)		0	5	2	17	0	0	6	16	15	0	0	2	6	12	0	5	0	0	16	11
muito alto (ma)		0	5	0	1	0	0	1	6	4	0	0	0	2	1	0	3	0	0	3	50
		Experimento #1					Experimento #2					Experimento #3					Experimento #4				
classe verdadeira \ prevista		mb	b	i	a	ma	mb	b	i	a	ma	mb	b	i	a	ma	mb	b	i	a	ma
muito baixo (mb)		24	1	3	0	3	16	1	5	2	5	30	1	4	3	2	34	0	14	0	8
baixo (b)		11	16	4	2	7	2	6	3	1	14	8	11	8	4	0	3	0	1	0	3
intermediário (i)		6	1	15	3	6	3	3	20	2	14	1	0	16	0	0	6	0	35	1	5
alto (a)		5	1	1	11	7	1	0	6	7	10	7	2	27	17	1	2	0	4	2	1
muito alto (ma)		7	3	2	5	23	1	3	4	1	37	7	0	8	4	6	15	0	12	0	21
		Experimento #5					Experimento #6					Experimento #7					Experimento #8				

Para previsão da idade de abate, foram realizados dois experimentos, #1 e #5, cujos modelos gerados apresentaram acurárias de 63,47% e 53,29% respectivamente, com diferença de 10,18% em favor do primeiro. Contudo, a matriz de confusão referente ao experimento #1 evidencia a incapacidade do modelo em classificar as instâncias nas idades de abate muito baixa e muito alta, o que sobremaneira impede que o mesmo tenha proveito prático em alinhamento com os objetivos do presente trabalho. Esta problemática não foi presente nos resultados obtidos no experimento #5, cuja árvore de decisão resultante é apresentada na Figura 3.

Os experimentos #2 e #6, referentes à previsão do peso de abate, apresentaram acurárias de 49,70% e 51,50% respectivamente, com diferença de 1,80% em favor do segundo. Observou-se que o experimento #2, além de não ter logrado melhor acurácia em comparação com o experimento #6, expõe a mesma problemática apresentada pelos resultados do experimento #1. À exemplo deste, o experimento #2 foi incapaz de classificar as instâncias de animais bovinos nas categorias muito baixo e muito alto, neste caso acerca do peso de abate. A árvore de decisão resultante do experimento #6 é apresentada na Figura 4.

A previsão do ganho médio diário de peso (GMD) foi abordada nos experimentos #3 e #7. Seus resultados apresentaram as acurárias de 61,68% e 53,89%,

respectivamente, com diferença de 7,79% em favor do primeiro. O experimento #3, bem como os experimentos #1 e #2, resultou em um modelo incapaz de classificar bovinos na categoria muito alto para GMD. A classificação dos bovinos em GMD muito baixo ocorreu de forma semelhante, com a diferença de que o modelo foi capaz de classificar 1 instância nesta categoria e corretamente. O experimento #7 resultou num modelo livre desta problemática, conforme apresentado na Figura 5.

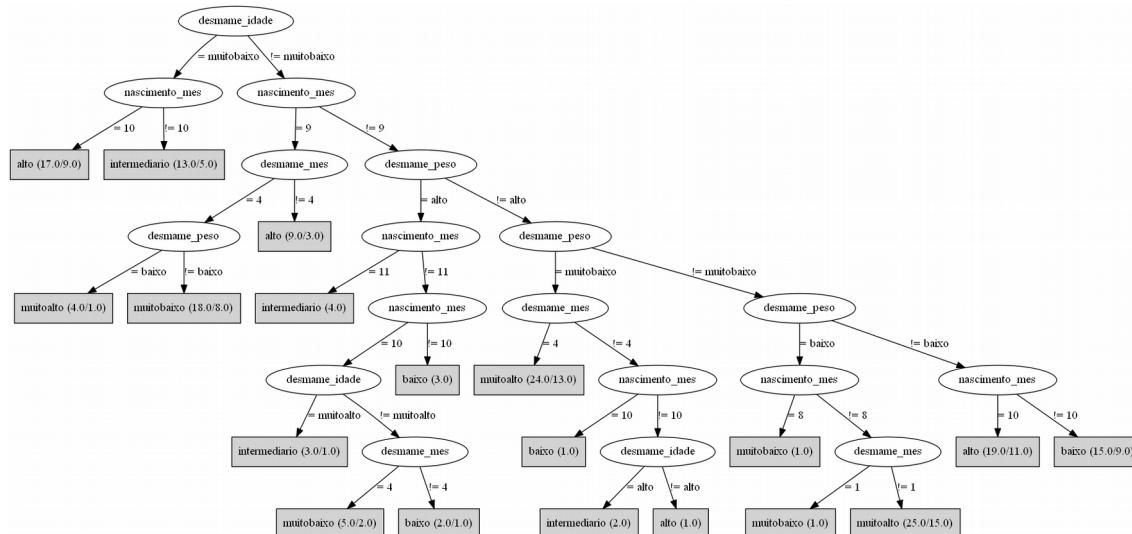


Figura 3. Árvore de decisão para predição da idade de abate

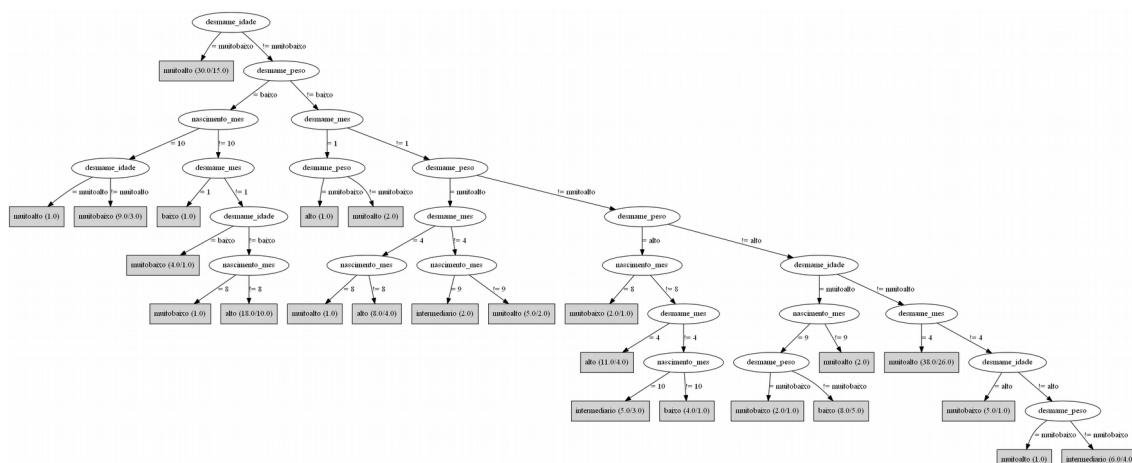


Figura 4. Árvore de decisão para predição do peso de abate

Os experimentos #4 e #8 foram realizados para predição da bonificação e apresentaram os resultados de 53,29% e 55,09%, respectivamente, em acurárias, tendo a diferença de 1,80% em favor do segundo. Contudo, o modelo gerado pelo experimento #8 apresentou incapacidade de classificar instâncias com a bonificação na categoria baixo, tornando o modelo impraticável. O modelo gerado pelo experimento #4 foi isento de tal problemática e sua respectiva árvore de decisão é apresentada na Figura 6.

Em sumário, nos testes realizados, os conjuntos de dados cujos atributos numéricos foram discretizados pela descoberta automatizada dos 5 agrupamentos com aplicações do algoritmo Simple k-means foram as entradas mais adequadas para processamento com tarefas de classificação com o algoritmo J48 sobre as variáveis idade de abate, peso de abate e GMD, ao passo que o conjunto de dados cuja

discretização dos atributos numéricos foi realizada pela segmentação igualitária do intervalo numérico em 5 frações foi a entrada mais adequada para previsão da bonificação.

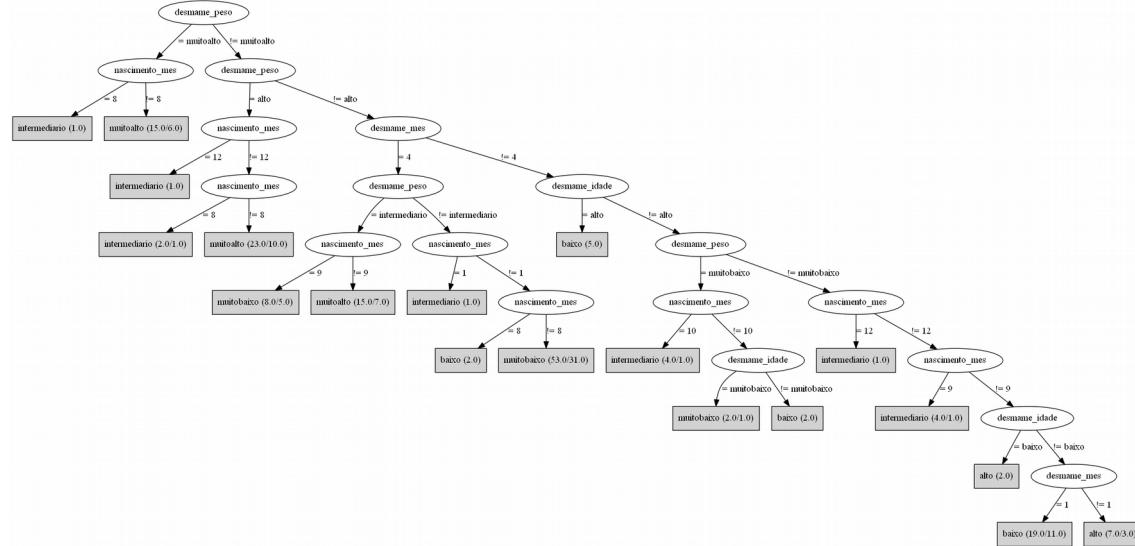


Figura 5. Árvore de decisão para previsão do GMD

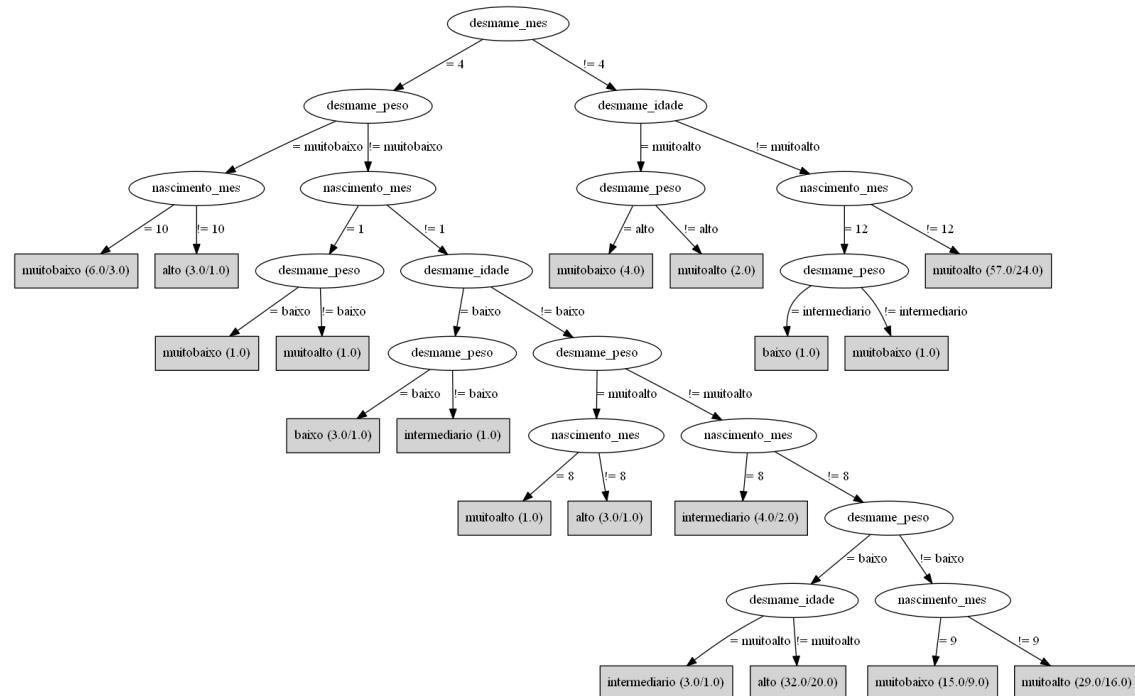


Figura 6. Árvore de decisão para previsão da bonificação

Isto ocorreu em virtude do desbalanceamento entre as frequências das categorias após as tarefas de discretização, que podem ser observadas nos histogramas apresentados nas Figuras 1 e 2. Estas figuras evidenciam as diferenças de balanceamento das classes e permitem a comparação dos resultados da discretização para os dois métodos de discretização utilizados sobre os atributos originalmente numéricos. Entende-se que o desbalanceamento observado em alguns histogramas são a

causa da impossibilidade do algoritmo em gerar modelos que não negligenciam quaisquer classes a partir dos respectivos conjuntos de dados.

4. Conclusão

O presente trabalho buscou um método para descoberta de árvores de decisão capazes de auxiliar os produtores de gado de corte na previsão de indicadores zootécnicos da qualidade das carcaças com base nas respectivas variáveis de cria, ou seja, dados que podem ser coletados entre o nascimento e o desmame dos animais.

Foram realizados experimentos de classificação com o algoritmo J48 após pré-processamento do conjunto de dados para discretização dos atributos numéricos. O método tradicional de discretização, pelo particionamento igualitário do intervalo numérico, se mostrou problemático ao produzir categorias com frequências desbalanceadas para atributos do conjunto de dados apresentado. Diante desta situação, foi proposto que as tarefas de discretização fossem realizadas através da descoberta automatizada das categorias por medida da distância de Manhattan, através de aplicação do algoritmo Simple k-means.

Esta abordagem proveu maior qualidade de discretização para 75% dos experimentos realizados, provendo melhor balanceamento entre as categorias criadas em relação à primeira, que foi capaz de produzir categorias sem problemática de desbalanceamento em 25% dos experimentos realizados.

Finalmente, foi possível descobrir árvores de decisão capazes de explicar a influência das variáveis de cria mês de nascimento, mês de desmame, peso de desmame e idade de desmame nos indicadores zootécnicos de qualidade de carcaças, como idade de abate, peso de abate, ganho médio diário de peso e bonificação, cumprindo de forma satisfatória o objetivo do estudo.

Trabalhos futuros envolvem tarefas adicionais de coleta de dados, com vistas no melhoramento da representatividade das classes descobertas por discretização, de modo a possibilitar maior adequação do método de teste dos modelos descobertos. Também será objetivada a otimização da acurácia das árvores de decisão, o que pode ser abordado por diferentes métodos de discretização dos atributos numéricos, diferentes métodos de descoberta de agrupamentos para aplicação na discretização, experimentações com outros algoritmos de indução de árvores de decisão, empilhamento de classificadores e aprendizado sensível à custo.

Referências

- Arthur, D. and Vassilvitskii S. (2007) k-means++: the advantages of carefull seeding. In: Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms, 1027-1035.
- Barbosa, P. (1999) Raças e estratégias de cruzamento para produção de novilhos precoces. Embrapa Pecuária Sudeste. In: Simpósio de Produção de Gado de Corte, 1. Viçosa, Brasil.
- Costa, C. L. (2016). Utilização de características zootécnicas e de manejo na pecuária para previsão do peso final e bonificação de bovinos empregando redes neurais artificiais. Tabalho de conclusão de curso, Universidade Federal do Pampa.
- Euclides Filho, K. (2000) Produção de bovinos de corte e o trinômio genótipo-ambiente-mercado. Embrapa Gado de Corte - Documentos (Infoteca-E).
- Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P. and Witten, I. (2009) The WEKA Data Mining Software: An Update. SIGKDD Explorations, Volume 11, Issue 1.
- Likert, R. (1932) A Technique for the Measurement of Attitudes. Archives of Psychology. 140: 1–55.

- Maciel, T., Seus, V., Machado, K. and Borges, E. (2015). Mineração de dados em triagem de risco de saúde. *Revista Brasileira de Computação Aplicada*, 7(2), 26-40.
- Mota, F., Souza, K., Ishii, R. and Gomes, R. (2017) BovReveals: uma plataforma OLAP e data mining para tomada de decisão na pecuária de corte. In:: Congresso Brasileiro de Agroinformática, 11. Campinas, Brasil.
- Quinlan, R. (1993) C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, San Mateo, CA.
- Quinlan, J. R. (1996). Improved use of continuous attributes in C4. 5. *Journal of artificial intelligence research*, 4, 77-90. Chicago, IL.
- Witten, I., Frank, E., Hall, M. and Pal, C. (2017) Data mining: practical machine learning tools and techniques. Morgan Kaufmann.

Diferenciação de Perfis de Curva de Carga para Identificação de Perdas Não-Técnicas em Redes de Distribuição Utilizando Mineração de Dados e Aprendizado de Máquina

Jorge Gustavo Sandoval Simão, Raimundo Celeste Ghizoni Teive

Universidade do Vale do Itajaí

jorge.sandoval@univali.br, rteive@univali.br

Abstract. — *It is estimated that, in Brazil, approximately 15% of electricity is consumed fraudulently, a practice known as non-technical loss. This article is an applied study that analyzes the identification of these losses in the distribution networks in order to recognize distortions in relation to the respective consumption patterns of each consumer and, from the analysis of this information, to detect possible deviations that can be classified as frauds. For this purpose, data mining and machine learning techniques were applied to commercial consumers in the State of Santa Catarina, presenting evidence that these techniques can be used to identify patterns of irregular electricity consumption.*

Resumo. É estimado que, no Brasil, aproximadamente 15% da energia elétrica é consumida de forma fraudulenta, prática essa denominada perda não-técnica. Este artigo é um estudo aplicado que analisa a identificação dessas perdas nas redes de distribuição para reconhecer distorções em relação aos respectivos padrões de consumo de cada consumidor e, a partir da análise desta informação, detectar possíveis desvios que possam ser classificados como fraudes. Para tal, foram aplicadas técnicas de mineração e dados e aprendizado de máquina nos consumidores comerciais do Estado de Santa Catarina, apresentando evidências de que essas técnicas podem ser usadas para identificar padrões de consumo de energia elétrica irregulares.

1. Introdução

O crescimento das concessionárias de energia elétrica e o aumento dos prejuízos financeiros causados pelas perdas de energia elétrica consequentes das não-conformidades nas redes de distribuição, têm incrementado a busca por novas tecnologias de detecção de fraudes [Queiroz *et al.* 2016].

As perdas de energia elétrica são divididas em duas categorias: as técnicas e as não-técnicas. As perdas não-técnicas (também conhecidas como perdas comerciais) são causadas pela manipulação ilegal de medidores ou por falhas nas instalações de consumo, enquanto as técnicas são causadas através de efeitos físicos (o efeito Joule, por exemplo), devidos à distribuição de energia [Guerrero *et al.* 2014]. As perdas não-técnicas, dada sua natureza, geram diminuição no registro do consumo de energia, ocasionando assim modificações nas topologias das curvas de carga.

As fraudes podem ser divididas nas categorias de residenciais, industriais e comerciais, sendo estes últimos o foco deste estudo. Os consumidores comerciais representam uma parcela considerável da energia consumida. Em 2014, em Santa Catarina (Brasil), a energia consumida pela classe comercial foi de 3.946.188 MW/h, significando 16,9% do consumo total (representando um aumento em relação à 2013, que teve um consumo de 3.604.418 MW/h) [FIESC 2015]. Considerando que podem haver aumentos anuais do consumo de energia da classe comercial, o desvio passou a tornar-se cada vez mais significativo. Em 2016, também em Santa Catarina, os consumos de energia elétrica residencial e comercial subiram mais do que o PIB (Produto Interno Bruto) com uma taxa de aumento de 3,6% [EPE 2016]. Dados estes registros, a representatividade das perdas não-técnicas sobre a classe comercial torna-se significativa.

Muitos métodos existentes para identificação de perdas não-técnicas impõe um alto custo operacional, e requerem uso extensivo de recursos humanos [Nizar *et al.* 2006], porém há a necessidade de tentar mensurá-las, pois essas fraudes representam perdas econômicas, colocam em risco a segurança pública e criam impactos sociais. A Superintendência de Pesquisa e Desenvolvimento e Eficiência Energética (SPE) da ANEEL (Agência Nacional de Energia Elétrica) propôs, em 2013, a criação de um sistema de informações envolvendo os agentes do setor, que possibilitasse (entre outros objetivos) a criação e uma base de dados consistente para a aplicação de técnicas de inteligência analítica e de mineração de dados [SPE 2013]. Considerando a implementação desta tecnologia para as concessionárias de energia, as técnicas de inteligência computacional também podem ser utilizadas sobre os dados dos clientes de uma distribuidora.

Sendo assim, este estudo propõe um método de detecção de fraudes de consumidores comerciais baseado na utilização de aprendizado de máquina, agrupando as curvas de carga através de sua tipologia utilizando o algoritmo *K-Means* e posteriormente aplicando uma rede neural não-supervisionada (*Autoencoder*) no resultado, visando identificar usuários que fogem ao perfil do *cluster* ao qual pertencem, durante o período analisado.

2. Identificação de Perdas Não-Técnicas

Uma das fraudes na rede de distribuição mais comuns é o caso de medidores com freios (furos e/ou agulhas) no disco de medição. Este tipo de fraude faz com que o disco medidor de energia fique travado em um determinado valor, e não meça toda a energia consumida. Este padrão de curva de consumo também acontece quando é realizada ligação direta antes do medidor de energia e o usuário utiliza a energia passada pelo medidor apenas para determinados fins, diminuindo assim, o consumo registrado pelo medidor [Queiroz *et al.* 2016].

As tipologias de curvas de carga dos consumidores comerciais seguem determinados padrões, baseados na sua atividade econômica e características funcionais das instalações, que permitem seu agrupamento pelas mesmas, pois muitas seguem comportamentos semelhantes. Nota-se uma distorção no perfil de consumo, quando um consumidor apresenta uma modificação de tipologia de curva de carga que o diferencia dos outros consumidores do mesmo *cluster*, indicando uma variação significativa do

padrão de consumo de energia, o que pode sinalizar perda não-técnica. Os tipos de padrões de consumo (anômalos ou não) variam de consumidor para consumidor, mas são mais comuns entre determinados grupos de consumidores e mantém-se padrão ao longo do ano, com pequenas diferenças causadas principalmente pela sazonalidade.

Considerando a existência de padrões de perfil de consumo e as distorções (não-conformidades) causadas no mesmo pelas perdas não-técnicas, é possível detectar padrões de consumo através das tipologias que podem representar possíveis fraudes nas redes de distribuição de energia elétrica, utilizando-se algoritmos inteligentes sobre os dados de curvas de cargas diárias.

3. Metodologia Aplicada

3.1. Definição do Escopo

Para a identificação de perdas não-técnicas deste estudo foram analisados os consumos das divisões CNAE (Classificação Nacional de Atividades Econômicas) catarinenses mais representativas, focando exclusivamente na classe comercial (postos de gasolina, concessionárias de automóveis, mercados, entre outros) em um prazo pré-determinado (ano de 2011).

3.2. Banco de Dados de Curvas de Carga

Os valores de consumo de energia elétrica diário de cada usuário são armazenados pelas concessionárias após a leitura horária dos valores de consumo nos medidores em bases de dados. A base utilizada para este estudo pertence à uma distribuidora de energia brasileira e contém informações de consumo do ano de 2011, contendo o código da unidade consumidora, data e hora do consumo, bem como potência ativa e potência reativa consumida por aquele consumidor.

O número total de consumidores nesta base é de 503, sendo destes 233 consumidores comerciais. Para cada consumidor foram analisadas as tipologias das curvas de carga dos dias da semana (segunda á sexta), para todas as estações, sendo então 240 curvas de carga por consumidor (e 55.920 curvas de carga no total). Como complemento, esta mesma base possui informações fornecidas pelo consumidor para a distribuidora de energia, que apesar de não serem usadas diretamente na análise das tipologias, permitem uma melhor definição do perfil daquele consumidor.

3.3 Tipologia das Curvas de Carga

Os valores de consumo de energia elétrica diária de cada usuário são armazenados pelas concessionárias após a leitura dos valores de consumo nos medidores. Através da análise dos valores de energia consumida, é possível classificar os consumidores por padrões de consumo. Independentemente do tipo de consumidor, o consumo de energia elétrica possui um comportamento sazonal e cíclico que pode ser demonstrado através das tipologias de curvas de carga quando é feita a análise de consumo dos usuários. O comportamento regular desta curva é chamado de padrão ou perfil de consumo.

As curvas de carga podem ser definidas como residenciais, industriais, comerciais ou de serviços, para estações quentes e frias, em dias de semana ou fins de semana, e são obtidas agrupando os perfis de carga de acordo com sua similaridade

[Azad *et al.* 2014]. Nos dados utilizados para esta pesquisa, os perfis de consumo diário dos consumidores são registrados por hora, com dados coletados em um intervalo de cinco minutos. Estes dados também foram divididos sazonalmente, pois há mudança na tipologia das curvas de um mesmo consumidor em função da estação.

Através da identificação de características comuns entre os *clusters* e através da análise das curvas de consumo de energia elétrica, é possível detectar padrões de consumo que podem ser usados para classificar consumidores e detectar anomalias nas redes de distribuição de energia elétrica [Queiroz *et al.* 2016]. Além disso, esses padrões podem ser refinados com a aplicação da rede neural não-supervisionada.

3.4. Knowledge Discovery in Databases e Redução de Dimensionalidade

O KDD é o processo geral de conversão de dados brutos em informações úteis. Ele consiste em uma série de passos de transformação, do pré-processamento dos dados até o pós-processamento dos resultados da Mineração de Dados [Pang-Ning *et al.* 2005].

A maioria dos métodos de Mineração de Dados é baseada na aprendizagem de máquina, reconhecimento de padrões e estatísticas, e possui um impacto direto sobre o desempenho e a eficácia de todo o procedimento. Estes métodos podem ser categorizados em diversos grupos, tais como classificação, análise de regressão, clusterização e sumarização [Tan *et al.* 2015]. Para este estudo foi utilizada a clusterização (*K-Means*) e a classificação com uma rede neural não-supervisionada (*Autoencoder*).

O *K-Means* define o elemento como se fosse um centróide (elemento central), o que normalmente é a média de um grupo de pontos, e é tipicamente aplicado a objetos em um espaço contínuo n-dimensional [Pang-Ning *et al.* 2005]. Ele toma aleatoriamente k pontos de dados (dados numéricos) como sendo os centróides dos *clusters*. Em seguida cada ponto (ou registro da base de dados) é atribuído ao cluster cuja distância deste ponto em relação ao centróide de cada *cluster* é a menor de todas as distâncias calculadas. Um novo centróide é computado pela média dos pontos, caracterizando a configuração do *cluster* para a interação seguinte. O processo termina quando os centróides param de se modificar, ou após um número limitado de iterações especificado pelo usuário [Goldschmidt and Passos 2005]. Os métodos centróides calculam a proximidade entre dois grupos calculando a distância entre os seus centróides.

Para conectar um ponto ao seu centróide mais próximo, é necessária a utilização de uma medida de aproximação que quantifica a noção de “próximo” para os dados especificados. A distância euclidiana é muitas vezes utilizada para pontos em um espaço euclidiano conforme apresentado em (1):

$$D_{(p,q)} = \sqrt{\sum_{i=1}^n (q_i - p_i)^2} \quad (1)$$

Após a aplicação do *K-Means*, as curvas de carga são divididas em *clusters*. Porém, mesmo sendo uma busca que é reduzida uma base diária, ainda não há como identificar determinadas perdas não-técnicas caso elas ocorram eventualmente (como por exemplo, um determinado equipamento que é ligado apenas em dias específicos e está conectado na linha de transmissão fora do medidor). Sendo assim, dados de alta dimensão podem ser convertidos em dados de baixa dimensão treinando uma rede neural multicamadas para reconstruir vetores de entrada de alta dimensão [Hinton and Salakhutdinov 2006], chamado de *Autoencoder*.

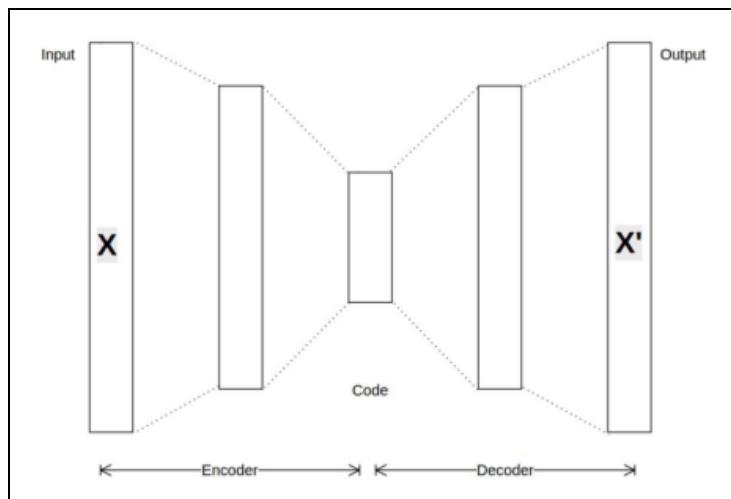


Figura 1. O leiaute de um Autoencoder

Conforme apresentado na Figura 1, um *Autoencoder* é um algoritmo de aprendizagem sem supervisão que tenta replicar a entrada depois de passá-la através de um gargalo de informações. É feito de várias camadas que primeiro comprimem o sinal, colocando-o através de um gargalo de informação. Este gargalo pode ser imposto por ter menos neurônios nas camadas sucessivas na primeira metade do *Autoencoder* (ou seja, o codificador). Também pode ser imposta por ter um critério de redução, pelo qual apenas uma certa fração de neurônios pode ser ativa. O modelo então tenta recriar o sinal o mais próximo possível do sinal original. Isso força o modelo a aprender recursos que podem ser usados para representar efetivamente os dados, apesar do gargalo da informação [Bhat *et al.* 2017]. O *Autoencoder* é treinado usando o algoritmo de *backpropagation*.

4. Resultados

Neste tópico são apresentados os resultados da clusterização *K-Means*, desde a seleção do número ideal de *clusters* até a clusterização dos dados, assim como o resultado da redução de dimensionalidade através da aplicação do *Autoencoder*.

Cada *cluster*, em cada uma das estações, apresenta um comportamento de consumo próprio e uma tipologia de curva de carga diferente para cada um dos consumidores. Porém, essas tipologias são similares entre si no *cluster* ao qual elas pertencem, pois os consumidores comerciais podem ter curvas de carga similares, independentemente da sua área de atuação, bastando apenas que o seu consumo em determinados horários siga padrões equivalentes aos de outro consumidor (o que caracteriza seu perfil de consumo).

Foram realizados testes para cada estação do ano com um número de *clusters* entre 2 (menor diluição possível) e 30 (maior diluição possível), visando encontrar o coeficiente de silhueta mais próximo de 0, mantendo-se um número de *clusters* que não diluisse demasiadamente as características entre as tipologias de curvas de carga. Para a demonstração deste processo, utilizou-se uma estação do ano como exemplo, a da Figura 1, que demonstra os coeficientes de silhueta para os dados durante os meses do inverno.



Figura 2. Análise dos coeficientes de silhueta para o inverno de 2 a 30 clusters

Coeficientes de silhueta próximos a 1 indicam que a amostra está longe da vizinhança. Um valor próximo a 0 indica que a amostra está dentro ou muito próxima do limite de decisão entre os *clusters* e valores negativos indicam que essas amostras foram associadas ao *cluster* errado. Na Figura 1, este coeficiente inicia-se em um valor mais próximo de 0 do que as outras estações analisadas nesta pesquisa (0.3528), o que significa que se não houvesse a necessidade de uma divisão de perfis por características mais específicas, muitas destas curvas de carga poderiam ser divididas em apenas dois *clusters*, considerando a similaridade de comportamento entre as tipologias. O valor de 15 *clusters* (marcado por um círculo vermelho), conforme apontado também pela clusterização hierárquica, foi selecionado para os dados do inverno pois o seu coeficiente de silhueta (0.3781) possui um valor médio mais próximo de 0 do que de 1, e não há uma diluição muito grande do número de características (como haveria dividindo os dados em mais de 20 *clusters*, por exemplo), o que permite um agrupamento por comportamento passível de estudo por esta pesquisa. A Tabela 1 apresenta os resultados dos números ideais de *clusters* apresentados para cada estação de acordo com a função de silhueta.

Tabela 1. Resultado do número ideal de clusters de acordo com a função de silhueta do K-Means

Verão	Outono	Inverno	Primavera
14	15	15	11

Na Figura 3 são apresentados exemplos de curvas de cargas dos *clusters* 8 e 10, para todos os dias da semana no período da primavera. Ela mostra que mesmo os consumidores possuindo perfis similares entre si, há diferenças nas tipologias das curvas

de cargas de cada *cluster*. Estas tipologias de curvas de carga representam, respectivamente e em sua maioria, as divisões CNAE 52 (mercados e lojas), 74 (assessorias contábeis e jurídicas), 70 (compra/locação de imóveis) e 85 (clínicas e hospitais). Os perfis agrupados em *clusters* indicam picos em horários de funcionamento de divisões CNAE que possuem um horário mais fixo (52, 74 e 70) e uma tipologia mais uníssona na 85, pois hospitais e clínicas possuem, em sua maioria, funcionamento 24 horas.

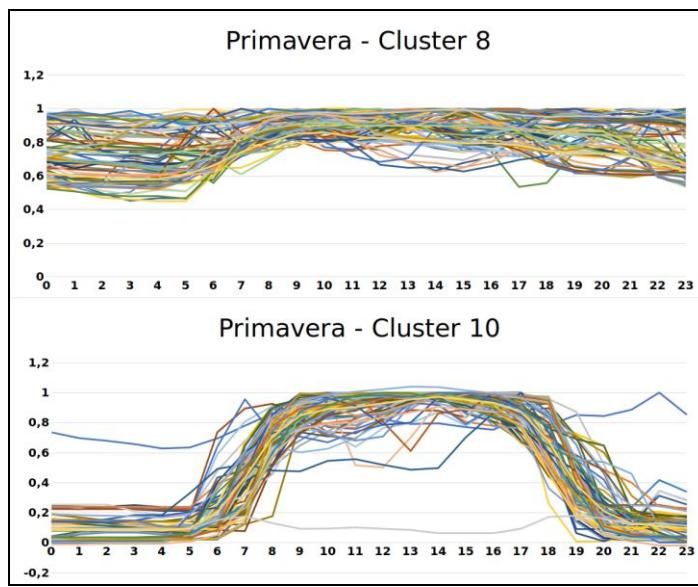


Figura 3. Perfis de consumo de dois clusters da primavera

Esta variação nas tipologias é dada pela diferença horária no consumo energético de cada consumidor comercial. O eixo *x* indica o horário, e o *y* o consumo normalizado pela fase de pré-processamento. Estes dados também foram divididos sazonalmente, pois há mudança no perfil de um mesmo consumidor em estações diferentes.

Tabela 2. Número de registros por cluster de acordo com cada estação.

Cluster/Estação	Verão	Outono	Inverno	Primavera
0	68	49	9	35
1	105	25	49	20
2	34	74	3	22
3	57	3	21	44
4	35	106	55	85
5	26	11	91	84
6	39	29	25	26
7	18	42	96	78
8	51	52	52	77
9	49	5	62	37
10	40	67	20	80
11	23	51	21	-
12	16	85	25	-
13	57	5	35	-
14	-	10	27	-

O perfil de consumo do usuário mantém um padrão similar durante todos os dias, pois as atividades comerciais dos consumidores utilizados neste estudo são regulares (ou seja, possuem intervalos definidos de início e fim, com interrupções sempre nos mesmos horários). As curvas de carga da Figura 3 apresentam o comportamento de cada consumidor que possui um perfil de consumo compatível com o do *cluster* no qual foi alocado. A Tabela 2 demonstra o número de registros por *cluster* utilizado para o estudo, de acordo com a estação.

Para estes resultados foram comparados todos os registros, visando identificar alguma dissociação entre os consumidores e os *clusters* aos quais os mesmos pertencem, ou seja, se durante a análise de todo o grupo de dados, algum consumidor alocava-se em um *cluster* diferente ao do grupo ao qual se associou originalmente.

Após a aplicação do *K-Means* sobre os dados da distribuidora de energia, detectou-se que muitos consumidores tendem a permanecer nos mesmos *clusters* durante todas as estações, ou seja, independentemente da sazonalidade, as variáveis climáticas aplicam sobre todos os membros daquele grupo o mesmo modificador, fazendo com que os mesmos mantenham seus perfis semelhantes durante todo o ano (e consequentemente, compartilhem o mesmo *cluster* em todas as estações).

Os perfis foram agrupados por *clusters*, de acordo com seu consumo em relação à todos os dias da semana. Com a aplicação do *Autoencoder*, utilizando 70% do volume de dados como treinamento e 30% como teste, percebeu-se que determinados dias, mesmo em um consumidor aparentemente constante durante a semana inteira, seu consumo poderia variar. A rede neural não-supervisionada, quando foi aplicada à dados já clusterizados através do *K-Means*, conseguiu reduzir a dimensionalidade, permitindo que distorções pudessem ser visualizadas em dias da semana específicos. A Figura 4 apresenta um perfil discrepante, especificamente às quintas feiras, para um consumidor específico cuja curva deveria seguir o mesmo padrão indicado nas linhas pontilhadas.

Em todos os agrupamentos, a maioria dos consumidores (70% ou mais) permanece junto aos outros consumidores do mesmo *cluster* por todo o ano, pois como a análise foi feita em clientes da distribuidora de uma mesma região, o impacto da variação climática é similar para todos. Quando a clusterização é aplicada, apenas perfis de consumo com uma discrepância evidente do *cluster* ao qual ele pertence são mostrados. Porém, com a aplicação do *Autoencoder* (feito com o uso de MATLAB), foi possível identificar curvas de cargas que possuíam padrões fraudulentos apenas em dias específicos, conforme apresentado na Figura 4. Assim, após o seu treinamento, a rede neural demonstrou curvas de cargas individuais em dias específicos que possuíam um comportamento suspenso. A Figura 4 apresenta os desvios padrões superior e inferior de um consumidor no inverno, em relação à media do *cluster*, representada pela linha azul. Os círculos vermelhos indicam os momentos do dia em que houveram desvios por parte do usuário, aproximadamente às 06:00h, 14:00h, 16:00h e 19:00h. Essas diferenças no comportamento do perfil da curva de carga podem indicar que houve algum tipo de adulteração ou desvio no medidor, fazendo com que parte da energia consumida não fosse processada pelo mesmo.

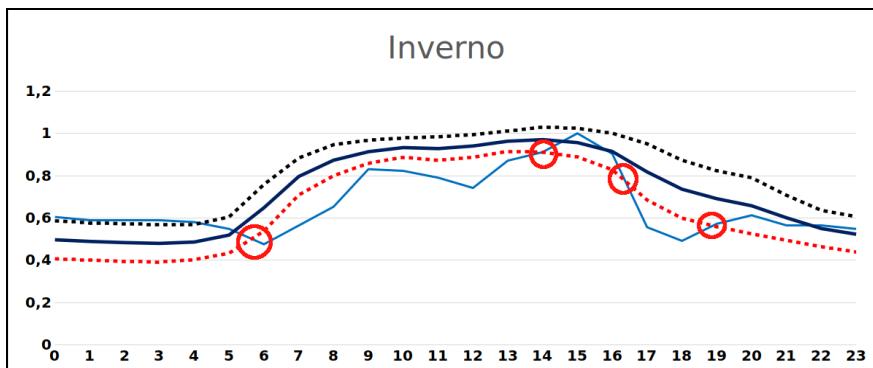


Figura 4. Média da tipologia das curvas de carga durante as quintas-feiras para o inverno no cluster 5.

Sendo assim, dentro do escopo desta pesquisa, qualquer usuário que apresentar desvios fora do padrão pode ser considerado suspeito de praticar fraude. De acordo com os resultados deste trabalho, então, as perdas não-técnicas podem ser inicialmente (pois maiores conjuntos de dados podem ser utilizados) identificadas pela sua diferença de comportamento em relação ao *cluster* ao qual esse usuário pertence, e se o mesmo possui desvios nesse padrão durante o horário comercial.

5. Conclusões

Considerando-se a quantidade de dados armazenados diariamente com os registros das leituras mensais de consumo de energia elétrica, é possível afirmar que a aplicação de mineração de dados e aprendizado de máquina para transformar tais dados em conhecimento pode auxiliar na identificação de distorções nos perfis de consumo dos usuários desta distribuidora.

Dentre os resultados obtidos através da clusterização permitiu-se, então, a divisão de consumidores pelo seu perfil de consumo (representado pela tipologia da curva de carga), possibilitando assim identificar quais elementos definem e diferenciam um consumidor que pertence à um determinado perfil. A classificação através do *Autoencoder* ampliou o nível de detalhamento, permitindo que um perfil pudesse ser analisado até dentro dos dias da semana.

Conclui-se, através da análise dos resultados obtidos, que determinados consumidores modificam o seu comportamento mesmo sofrendo influência da mesma variável, a sazonalidade. Como consumos referentes a dias típicos, tais como feriados, por exemplo, foram filtrados na fase de pré-processamento, infere-se que tais mudanças possam evidenciar possíveis distorções no consumo.

As variáveis que envolvem a modificação de uma tipologia estão associadas à uma variação frequente de consumo (tal como a prática de fraude), o que significa que os clientes da distribuidoras que estão apresentando uma diferença muito evidente no seu perfil, pode representar um comportamento suspeito que precisa ser submetido à uma avaliação mais detalhada. Apesar desta metodologia não garantir que um usuário possa ser responsável pelas perdas não-técnicas, os resultados obtidos com a aplicação da mesma pode direcionar as fiscalizações manuais que são realizadas pelas concessionárias de energia a fim de encontrar possíveis fraudes nas redes de distribuição de energia elétrica.

Referências

- Azad, S. A., Ali, A. B. M. S. and Wolfs, P. (nov 2014). Identification of Typical Load Profiles using K-means Clustering Algorithm. In *Asia-Pacific World Congress on Computer Science and Engineering*. . IEEE. <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5301726>.
- Bhat, R. R., Trevizan, R. D., Sengupta, R., Li, X. and Bretas, A. (2017). Identifying nontechnical power loss via spatial and temporal deep learning. *Proceedings - 2016 15th IEEE International Conference on Machine Learning and Applications, ICMLA 2016*, n. November 2017, p. 272–279.
- EPE (2016). Consumo de Energia Elétrica por Classe (Regiões e Subregiões). [http://www.epe.gov.br/mercado/Paginas/Consumomensaldeenergialetricaporclasse\(r%20egioes\),](http://www.epe.gov.br/mercado/Paginas/Consumomensaldeenergialetricaporclasse(r%20egioes),) [accessed on May 1].
- FIESC (2015). Santa Catarina em Dados 2015. http://fiesc.com.br/sites/default/files/medias/sc_em_dados_site_correto.pdf, [accessed on Mar 21].
- Guerrero, J. I., León, C., Monedero, I., Biscarri, F. and Biscarri, J. (2014). Improving Knowledge-Based Systems with Statistical Techniques, Text Mining, and Neural Networks for Non-Technical Loss Detection. *Knowledge-Based Systems*, v. 71, p. 376–388.
- Hinton, G. E. and Salakhutdinov, R. R. (28 jul 2006). Reducing the Dimensionality of Data with Neural Networks. *Science*, v. 313, n. 5786, p. 504–507.
- Nizar, A. H., Dong, Z. Y., Jalaluddin, M. and Raffles, M. J. (nov 2006). Load Profiling Method in Detecting non-Technical Loss Activities in a Power Utility. In *2006 IEEE International Power and Energy Conference*. . IEEE. <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4154468>.
- Pang-Ning, T., Vipin, K. and Steinbach, M. (2005). *Introduction to Data Mining*. Boston: Pearson Addison Wesley.
- Queiroz, A. de S., Franco, E. M. C. and López, G. P. (2016). Detecção de Fraudes nas Redes de Distribuição de Energia Elétrica Utilizando Técnicas de Inteligência Computacional. In *VI Simpósio Brasileiro de Sistemas Elétricos*.
- SPE (2013). SIASE – Sistema de Inteligência Analítica do Setor Elétrico. . http://www2.aneel.gov.br/arquivos/PDF/PD%20Estrat%C3%A9gico%2018-2013_SIASE.pdf.
- Tan, Y., Zhang, C., Ma, Y. and Mao, Y. (jun 2015). Knowledge discovery in databases based on deep neural networks. In *2015 IEEE 10th Conference on Industrial Electronics and Applications (ICIEA)*. . IEEE. <http://ieeexplore.ieee.org/document/7334293/>.

aper:180194_1

Aplicações de Mineração de Dados na Pecuária de Corte: Previsão de Indicadores de Qualidade de Carcaças

Rodrigo R. da Silva¹, Thales V. Maciel¹, Vinícius do N. Lampert², Denizar S. de Souza³

¹Instituto Federal de Educação, Ciência e Tecnologia Sul-rio-grandense (IFSUL)
Campus Bagé – Av.Leonel de Moura Brizola, 2501 – 96.418-400 – Bagé – RS – Brasil

²Empresa Brasileira de Pesquisa Agropecuária (EMBRAPA)
Unidade Pecuária Sul - CPPSUL – BR 153, Km 603 – Bagé – RS – Brasil

³Centro de Ciências Exatas e Aplicadas (CCEA)
Universidade da Região da Campanha (URCAMP) – Bagé – RS – Brasil

orki2008@gmail.com, thalesmaciel@ifsul.edu.br

vinicius.lampert@embrapa.br, denizar.souza@urcamp.edu.br

Abstract. Considering that cattle breeders are able to acquire influence variables along the breeding process, this paper aims to provide a method for predicting carcasse bonus, daily average weight gain, age at slaughter and weight at slaughter based on influe variables to be collected until the bovine ablactation. For such, data mining applications were performed through linear regression applied to a 167 bovine instances dataset. Obtained results showed that carcasse bonus and daily average weight gain may be predicted with zero or insignificant error, maywhile age and weight at slaughter produced higher error rates upon prediction.

Resumo. Considerando que o produtor rural pode obter algumas variáveis de influência ao longo do processo produtivo do gado de corte, objetiva-se prever se as variáveis de influência obtidas até o desmame dos bovinos podem explicar a bonificação, ganho médio diário, idade de abate e peso de fazenda. Para tanto procede-se a mineração de dados através da regressão linear, em um conjunto de dados de 167 bovinos. Deste modo, observa-se que para a bonificação e ganho médio diário os modelos gerados apresentaram erros baixos, enquanto que para idade de abate e peso de fazenda os erros foram maiores, o que permite concluir que os atributos não foram o suficientes para predizer a idade de abate e peso de fazenda, mas bons para a bonificação e ganho médio diário.

Introdução

No contexto da pecuária de corte, o sistema produtivo pode ser conceituado como um conjunto de tecnologias e práticas de manejo. Bem como o perfil do animal, a intenção da criação, a raça ou grupamento genético e a região onde a atividade é desenvolvida [Euclides Filho 2000].

Segundo [Gomes et al. 1997], os negócios agropecuários revestem-se da mesma complexidade e dinâmica dos demais setores da economia, requerendo do produtor uma

nova visão diferenciada dos seus negócios, principalmente pela necessidade de se distanciar da posição de fazendeiro tradicional para assumir o papel de empresário rural.

Para analisar um sistema produtivo da pecuária de corte, é indispensável mensurar seus indicadores de qualidade, pois somente assim o produtor rural terá embasamento para tomada de decisão. Para [Oiagen 2010] a mensuração e análise de indicadores que retratam o funcionamento rural são fundamentais para a tomada de decisão. Estes indicadores, de acordo com [Oiagen and Barcellos 2008], são conhecidos como variáveis de influência, ou seja, informações gerenciais de ordem técnica ou econômica que contribuem com avaliações precisas dos processos internos da propriedade rural.

Ainda segundo [Oiagen 2010], deve ficar claro que para a empresa rural, interessa, sobretudo, a rentabilidade, que é o elemento mais importante na avaliação da atividade econômica praticada em moldes capitalistas. Este indicador de desempenho deve situar-se em nível adequado para que o investimento se justifique. No âmbito do criador e das informações que estão acessíveis a ele, os indicadores devem possuir relevância para serem aplicados nos casos de estudos de caso.

O problema de pesquisa abordado neste trabalho é "existem variáveis de cria, ou seja, dados coletados sobre indivíduos de rebanhos bovinos entre nascimentos e desmames, que explicam bons indicadores de qualidade zootécnicas?". A hipótese é que os dados mês de nascimento, mês de desmame, peso de desmame e idade de desmame têm correlação suficiente com o peso de fazenda e idade de abate para explicar bons valores em tais indicadores. Adicionalmente, ganho médio diário de peso e bonificação também são investigados.

O objetivo deste trabalho é descobrir a relação estatística entre as variáveis de cria e os indicadores zootécnicos de qualidade de carcaças após abate. Quantificar o peso dos atributos e hipóteses dos respectivos domínios de valores nos indicadores de qualidade inferidos. Para tal, foram realizadas tarefas de mineração de dados no âmbito de descoberta de conhecimento em banco de dados. O foco da atividade ocorreu com experimentos de regressão, conforme descrito na metodologia.

Referencial Teórico

Nesta seção é apresentado um referencial teórico sobre descoberta de conhecimento com mineração de dados, seguido de um levantamento de suas aplicações na pecuária de corte.

Descoberta de Conhecimento em Banco de Dados

Observa-se que uma grande quantidade de dados cresce de forma acelerada em diversos campos de conhecimento, fato que dificulta a sua interpretação, pois o volume destes dados é maior que o poder de interpretá-los [Vieira and Oliveira 2014]. Desta forma, surgiu a necessidade do desenvolvimento de ferramentas e técnicas automatizadas para minimizar esta situação, as quais pudessem auxiliar o analista a transformar os dados em conhecimento [Han et al. 2011].

Grande parte dessas técnicas e ferramentas podem ser encontradas no processo de descoberta de conhecimento em bases de dados (DCBD). Segundo [Fayaad et al. 1996], DCBD é definida como um processo não trivial que busca identificar padrões novos, potencialmente úteis, válidos e compreensíveis, com o objetivo de melhorar o entendimento de um problema ou um procedimento de tomada de decisão.

O processo de DCBD compreende três principais etapas: pré-processamento, mineração de dados e pós-processamento [Tan et al. 2005]. No pré-processamento os dados são coletados e tratados para serem utilizados nas próximas etapas. A limpeza e a remoção de dados ruidosos também ocorre no pré-processamento, visando assegurar a qualidade dos dados selecionados. Subsequentemente, ocorre a mineração de dados, que são processos aplicados para explorar e analisar os dados em busca de padrões, previsões, erros, associações entre outros [Amaral 2016]. A etapa final consiste no pós-processamento, que engloba a interpretação dos padrões descobertos e a possibilidade de retorno a qualquer um dos passos anteriores. Assim, a informação extraída é analisada (ou interpretada) em relação ao objetivo proposto, sendo identificadas e apresentadas as melhores informações [Corrêa and Sferra 2003]. As tarefas de mineração de dados podem ser divididas em quatro grupos: classificação, regressão, agrupamentos e regras de associação.

A regressão é um tipo específico de classificação. Enquanto a classificação trata de previsão de valores nominais ou categóricos, chamados de classes, a regressão mantém o objetivo de realizar previsões, mas tem como alvo valores numéricos. No agrupamento não existe classe, o objetivo é criar grupos e atribuir instâncias a estes grupos a partir de características, ou atributos destas instâncias. Regras de associação buscam relações entre os ítems, gerando regras que determinam a associação entre esses ítems [Amaral 2016]. Este estudo tem foco em tarefas de regressão.

Revisão dos Trabalhos Correlatos

No âmbito da pecuária de corte, foram identificados trabalhos relacionados ao problema investigado nesta pesquisa.

No trabalho de [Mota et al. 2017] foi proposta uma abordagem de análise de dados com *data warehouse*, consultas analíticas online e mineração de dados, auxiliando o produtor na tomada de decisão do melhor momento para o abate. A abordagem se divide em 4 etapas: 1) responsável pela extração, transformação e carga dos dados; 2) etapa de criação do modelo multidimensional para armazenagem dos dados; 3) etapa de visualização e exploração dos dados armazenados no *data warehouse*; e 4) a aplicação de algoritmos de *data mining* por meio da ferramenta Weka. Na quarta etapa, há indícios de que a adoção de algoritmos de *data mining* fornecem uma taxa média de acerto acima de 62% em relação à predição do grau de acabamento e do rendimento de carcaça.

Já no estudo de [Costa 2016] foram analisados um conjunto de características zootécnicas para gerar um modelo afim de prever o rendimento dos bovinos, através das variáveis peso de fazenda (PF) e bonificação (BN). Para tanto o autor utilizou a técnica de Redes Neurais Artificiais (RNA's). Segundo aponta o autor, o resultado para o modelo de previsão de bonificação apresentou erro bem elevado, baixa correlação e generalização insatisfatória devido a uma limitação da ferramenta e da escolha dos dados utilizados na matriz de entrada da rede. Cabe ressaltar o trabalho não proveu comparações de desempenho com outros métodos de inferência de dados, tampouco indicações de peso de cada variável de entrada no produto de saída.

O trabalho difere-se dos demais por usar a tarefa de regressão no como técnica de processamento na descoberta de conhecimento, além disto, os atributos utilizados são diferentes, pois neste trabalho optou-se por analisar a influência das variáveis de cria em relação as variáveis de qualidade zootécnicas, contribuindo desta maneira para novas

abordagens, relatos e discussões sobre a temática da pecuária de corte.

Metodologia

O conjunto de dados analisado foi constituído por 167 instâncias de animais bovinos da raça Hereford. As nomenclaturas e respectivas descrições dos atributos do conjunto analisado são apresentadas na Tabela 1.

Tabela 1. Atributos utilizados

Nomenclatura	Tipo de Dado	Descrição
abate_peso	Numerico	Peso de abate na fazenda
nascimento_mes	Nominal	Mês de nascimento (1,8,9,10,11,12)
abate_idade	Numerico	Idade de abate
desmame_idade	Numerico	Idade de desmame
desmame_mes	Nominal	Mês de desmame (1,4,5)
desmame_peso	Numerico	Peso de desmame
gmd	Numerico	Ganho médio diário de peso
diff_abate_desmame	Numerico	Diferença entre abate/desmame
bonificação	Numerico	Bonificação

Como ferramenta para a realização das tarefas de pré-processamento e aplicações das tarefas de mineração de dados, foi utilizados o *software Waikato Environment for Knowledge Analysis* (WEKA), um ambiente para análise de conhecimento desenvolvido pela Universidade de Waikato, Nova Zelândia [Hall et al. 2009]. O WEKA tem como objetivo agregar algoritmos provenientes de diferentes abordagens dedicando-se ao estudo de aprendizagem de máquina. O grande número de algoritmos de aprendizado de máquina implementados pela WEKA é um dos maiores benefícios de usar a plataforma.

O experimento realizado dividiu-se em três etapas. Na primeira etapa os dados foram recuperados em formato .CSV afim de serem utilizados no *software WEKA*, o conjunto de dados original constava com 53 atributos e 1015 instâncias de bovinos de diversas raças. A etapa de pré-processamento deste conjunto de dados contou com tarefas de transformação, remoção de atributos irrelevantes, remoção atributos com dados faltantes e dos que não faziam parte do escopo dos experimentos, resultando no conjunto de dados descritos pela Tabela 1, realizou-se o calculo da diferença entre a data de abate e a data de desmame. Após o WEKA ser alimentado com os dados, foi aplicado o filtro *weka.filters.unsupervised.attribute.NumericToNominal* sobre os atributos desmame_mes e nascimento_mes de modo que os dados foram convertidos no formato numérico para nominal, afim de evitar que na forma numérica os meses constituíssem pesos quem afetassem os modelos descobertos.

A segunda etapa consistiu no processamento do conjunto de dados, que ocorreu com a tarefa de regressão linear, através do algoritmo *weka.classifiers.functions.LinearRegression* [Witten et al. 2016]. A regressão linear é utilizada basicamente com duas finalidades, prever o valor de y a partir do valor de x e estimar quanto x influencia ou modifica y . Adotou-se este algoritmo pois ele gera um modelo de comportamento, também produz o valor da correlação entre os atributos utilizados nos experimentos e o atributo alvo. Além disso, só usa as colunas que contribuem estatisticamente para a precisão, descartando e ignorando as colunas que não

ajudam a criar um bom modelo. Foram executados testes para cada uma das 4 variáveis alvo. Tabela 2 apresenta os atributos selecionados para cada um dos experimentos.

Tabela 2. Variáveis utilizadas nos experimentos

Variáveis	Bonificação	Peso de fazenda	Idade de abate	Ganho médio diário
Idade de desmame				
Mês de desmame				
Peso de desmame				
Mês de nascimento				
Diferença abate/desmame			Removido	
Bonificação	_____	Removido	Removido	Removido
Peso de Fazenda	Removido	_____	Removido	Removido
Idade de abate	Removido	Removido	_____	Removido
Ganho médio diário	Removido	Removido	Removido	_____

Na Figura 1 observa-se os modelos descobertos para os quatro experimentos realizados. O modelo é o resultado gerado pela tarefa de regressão linear. Nele, os atributos relevantes têm pesos atribuídos, de forma a comporem uma fórmula matemática para o cálculo do atributo alvo.

```

abate_peso =
22.6907 * nascimento_mes=11,10,9,1 +
51.6212 * nascimento_mes=1 +
0.5664 * desmame_peso +
0.3494 * desmame_idade +
34.22 * desmame_mes=1 +
0.1935 * diff_abate_desmame +
153.0857
                                         gmd =
                                         0.0335 * nascimento_mes=11,10,9,1 +
                                         -0.0163 * nascimento_mes=9,1 +
                                         0.0675 * nascimento_mes=1 +
                                         0.0007 * desmame_peso +
                                         -0.0002 * desmame_idade +
                                         0.0243 * desmame_mes=1,5 +
                                         -0.025 * desmame_mes=5 +
                                         -0.0003 * diff_abate_desmame +
                                         0.6084

(a) PF                               (b) GMD

bonificacao =
0.0112 * nascimento_mes=1,11,10 +
0.0002 * desmame_peso +
0.0001 * desmame_idade +
0.0155 * desmame_mes=1 +
0 * diff_abate_desmame +
-0.0139
                                         abate_idade =
                                         -1.7433 * desmame_peso +
                                         1138.5243

(c) BN                               (d) IA

```

Figura 1. Modelos descobertos

Para o peso de fazenda o experimento gerou um modelo onde os atributos utilizados foram mês de nascimento, peso de desmame, idade de desmame, mês de desmame e diferença abate/desmame, sendo o atributo mês de nascimentos = 1 o mais relevante pois apresenta dois coeficientes no modelo, dando um maior peso a este atributo, outro fato a se ressaltar está na circunstância de os mês de nascimento = 8, 9 e 12 não serem utilizados no modelo, o mesmo ocorre com o atributo mês de desmame = 4 e 5, sendo utilizado apenas o mês de desmame = 1.

Para o ganho médio diário o modelo gerado também apresenta o mês de nascimento = 1 como atributo de maior relevância, porém neste modelo, é gerado três coeficientes para este atributo, sendo que o mês de nascimento = 8 ou 12 não foram utilizados no modelo. Nota-se também que o mês de desmame = 5 apresenta dois coeficientes enquanto mês de desmame = 4 não foi utilizado no modelo.

O modelo gerado para a bonificação assemelha-se ao do peso de abate, salvo pelos valores dos coeficientes e de que o atributo mês de nascimento = 1 aparecer apenas uma vez no modelo, não sendo utilizado o mês de nascimento = 8, 9 ou 12, comportamento semelhante ocorre com mês de desmame. O modelo de idade de abate foi o mais simples, levando em conta apenas o atributo peso de desmame desconsiderando os outros atributos.

Análise dos Resultados

Além dos modelos, cada experimento apresentou o relatório de valores reais para cada valor previsto, o valor previsto e a diferença entre eles(erro na previsão). Analisando os erros de cada instância com o valor real, observa-se que os erros para o ganho médio diário e bonificação foram baixos, as maiores diferenças entre o valor real e o previsto ocorreram na idade de abate. O peso de fazenda apresentou um desenho razoável por não apresentar uma variação muito elevada do erro.

A Tabela 3 apresenta os valores para comparação dos coeficientes de correlação e erros médios absolutos, calculados pelo algoritmo de regressão linear.

Tabela 3. Correlação e erro médio absoluto

Classes	Correlation coefficient	Mean absolute error
Bonificação	0.4067	0.0136
GMD	0.7736	0.0316
Idade de Abate	0.4092	83.9369
Peso de Fazenda	0.5882	27.2316

A correlação é uma medida estatística que indica a força e a direção da relação entre variáveis numéricas [Amaral 2016]. Ou seja, a correlação é um índice que indica o quanto duas variáveis estão relacionadas, sendo os valores retornados sempre dentro do intervalo de -1 e 1. Quanto mais próximas de -1 e 1, maior será a correlação entre as variáveis, e da mesma forma, quanto mais próxima de 0, mais fraca ela é.

O indicador de direção é dado pelo sinal da correlação, uma correlação positiva indica que enquanto uma variável cresce, a outra, correlacionada, também cresce, já na correlação negativa, enquanto uma variável cresce a outra diminui [Amaral 2016].

Analizando a Tabela 3 nota-se que GMD foi o que apresentou a maior correlação entre as variáveis preditoras, indicando que o modelo gerado teve uma boa métrica de qualidade, pois todas as variáveis utilizadas possuem uma boa correlação, além disso o erro médio ficou baixo. Mesmo caso do erro ocorreu com a bonificação,ou seja, o algoritmo quando não acertou o valor, errou por pouca diferença,para mais ou para menos. Para a idade de abate a média de erro ficou em 83.9369 dias e o peso de abate na fazenda em 27.2316 quilos.

As figuras 2, 3, 4 e 5 apresentam as distribuições de frequências para os erros

ocorridos nos experimentos referentes a bonificação, ganho médio diário, idade de abate e peso de fazenda.

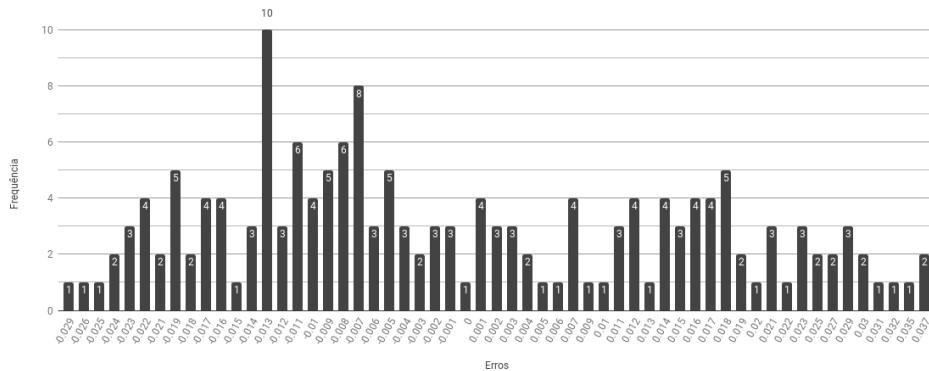


Figura 2. Frequência de Erros para Bonificação

Observando a Figura 2, nota-se que para a bonificação as maiores frequências então entre o intervalo $-0.013 \leftarrow -0.005$, indicando que o modelo calculou os valores e os erros concentraram-se neste intervalo, além disso a tendência linear indica que para os maiores valores positivos dos erros, a tendência da frequência é o valor 2(dois), caso contrário, para os menores valores negativos dos erros a tendência é 4(quatro), para o erro 0(zero), ou seja, que o modelo acerto o valore real, a tendência é 3(três), nota-se que a tendência corrobora com o intervalo das maiores frequências.

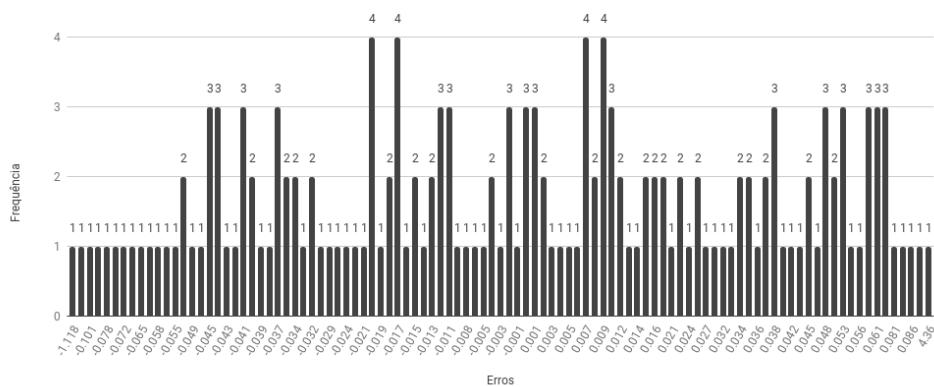


Figura 3. Frequência de Erros para GMD

Conforme a Figura 3, para o ganho médio diário, as maiores frequências então entre o intervalo $-0.02 \leftarrow 0.01$, demonstrando uma distribuição homogênea em torno do valor 0(zero), diferentemente da bonificação. No caso do GMD, a tendência linear para os menores valores negativos dos erros a tendência é 1.5(um ponto cinco) e para os maiores valores positivos dos erros, a tendência da frequência é o valor 2(dois), para o erro 0(zero) a tendência ficou aproximada a 1.75(um ponto setenta e cinco).

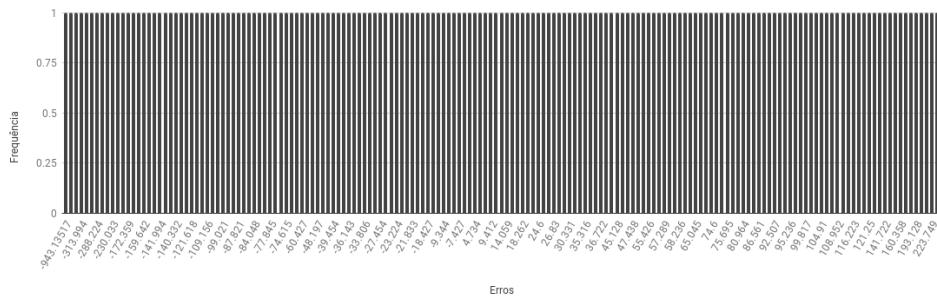


Figura 4. Frequência de Erros para Idade de Abate

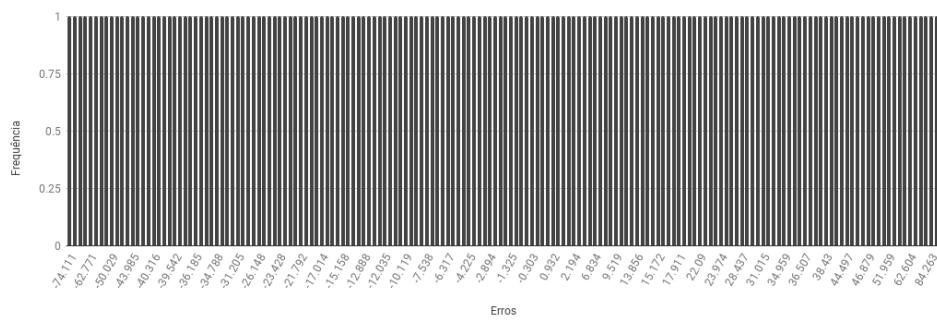


Figura 5. Frequência de Erros para Peso de Fazenda

Analisando as Figuras 4 e 5, a idade de abate e peso de fazenda, não houve repetição de erros, o que gerou uma frequência de 1(um) para todas as instâncias.

Com referência aos modelos gerados, os atributos não utilizados podem ter influenciado o resultado dos experimentos. Conforme Figura 1(d), para idade de abate o único atributo considerado relevante no modelo foi peso de desmame, desconsiderando as informações constantes nos outros atributos, podendo ter sido gerado um modelo pobre de previsão. Os outros modelos, Figuras 1(a), 1(b) e 1(c), também desconsideraram algumas informações como mês de nascimento = 8 e 12 e mês de desmane = 4, podendo estes terem seus desempenhos afetados por essas exclusões de informações.

O algoritmo de regressão linear ainda fornece o valor de R^2 , que é o coeficiente de determinação. Ele fornece uma informação auxiliar ao resultado da análise de variância da regressão, como maneira de se verificar se o modelo proposto é adequado ou não para descrever o fenômeno estudado. O valor de R^2 varia no intervalo de 0 a 1. Valores próximos de 1 indicam que o modelo proposto é adequado para descrever o fenômeno. Tabela 4 apresenta os valores do R^2 para os experimentos realizados.

Tabela 4. Valores dos coeficientes de determinação

Classes	Coeficiente de determinação - R^2
Bonificação	0.1654
GMD	0.5985
Idade de Abate	0.1675
Peso de Fazenda	0.3464

Analisando os valores de R^2 encontrados, observa-se que apenas o ganho médio diário apresentou um coeficiente relativamente elevado. Os outros indicam que os modelos descobertos não são adequados para descrever as variáveis zootécnicas de qualidade.

Conclusão

O presente trabalho teve como objetivo descobrir a influência e a relação das variáveis de produção e manejo na bonificação, peso de fazenda, ganho médio diário e idade de abate, em relação as variáveis de cria com o uso de aplicações de mineração de dados, almejando que os modelos descobertos possam empregados com a finalidade de auxiliar os produtores na gestão eficiente do negócio.

Foi realizada a aquisição e seleção de indicadores de qualidade zootécnicos e de cria, o tratamento dos dados e uso da tarefa de mineração de dados sobre os mesmos. Todos os resultados foram discutidos na análise dos resultados, evidenciando a razão pela qual os mesmos foram obtidos.

Pode-se concluir que os resultados foram parcialmente alcançados, pois com as tarefas de regressão configuradas conforme descritas na metodologia, mostraram que as variáveis de cria usadas possuem boa correlação apenas para o ganho médio diário, e os modelos gerados para a bonificação e ganho médio diário apresentaram erros baixos, indicando que as variáveis de cria usadas podem explicar um ganho médio diário alto ou baixo, assim como a bonificação. Para a idade de abate e peso de fazenda, os modelos apresentaram diferenças maiores entre o valor real e o valor previsto, e baixa correlação, podendo significar que as variáveis de cria usadas nos experimentos não sejam suficientes para explicar o peso de fazenda e a idade de abate, além disso a média dos erros ficou elevada e fora dos pradões esperados. A frequência dos erros ficou heterogênea, com a tendência linear igual a 1(um) para as instâncias destes dois atributos. O coeficiente de determinação indica que no contexto da pecuária de corte, o modelo descoberto para o ganho médio diário poderá ser utilizado como ferramenta de consulta para os produtores, os outros modelos necessitam que um melhor tratamento.

Trabalhos futuros envolvem à adoção de novos indicadores, como por exemplo o peso de nascimento, tipo de alimentação da mãe do bovino enquanto este ainda mama, entre outros, para testar e observar como os modelos se comportam, pode-se também empregar outros tipos de técnicas de mineração de dados como o algoritmo M5P e redes neurais, com treinamento e configuração das camadas ocultas. Também se sugere a expansão do banco de dados, através de parcerias com outros produtores rurais, e do estudo para consideração de outras raças bovinas. Apresentando ao produtores os resultados obtidos e demonstrando que é possível aumentar seu rendimento com técnicas adequadas.

Referências

- Amaral, F. (2016). *Aprenda Mineração de Dados - Teoria e Prática*. Rio de Janeiro: Alta Books, 1th edition.
- Corrêa, Â. M. J. and Sferra, H. (2003). Conceitos e aplicações de data mining. *Revista de ciência & tecnologia*, 11:19–34.
- Costa, C. L. (2016). *Utilização de características zootécnicas e de manejo na pecuária para previsão do peso final e bonificação de bovinos empregando redes neurais artificiais*. Tabalho de conclusão de curso, Universidade Federal do Pampa.

- Euclides Filho, K. (2000). Produção de bovinos de corte e o trinômio genótipo-ambiente-mercado. *Embrapa Gado de Corte-Documentos (INFOTECA-E)*.
- Fayaad, U. M., Shapiro, G. P., and Smyth, P. (1996). From data mining to knowledge discovery: An overview.
- Gomes, A., Carneiro, A., Yamaguchi, L., Passos, L., Carvalho, M., and Campos, O. d. (1997). Acompanhamento de fazendas produtoras de leite.
- Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18.
- Han, J., Pei, J., and Kamber, M. (2011). *Data mining: concepts and techniques*. Elsevier.
- Mota, F. d., Souza, K., Ishii, R., and Gomes, R. d. C. (2017). Bovreveals: uma plataforma olap e data mining para tomada de decisão na pecuária de corte. In *Embrapa Gado de Corte-Artigo em anais de congresso (ALICE)*. In: CONGRESSO BRASILEIRO DE AGROINFORMÁTICA, 11., 2017, Campinas. Anais... Campinas: Embrapa Informática Agropecuária; Unicamp, 2017.
- Oiagen, R. and Barcellos, J. (2008). Gerenciamento e custo de produção. *MOURA, JA et al. Programa de atualização em medicina veterinária*. Porto Alegre: ARTMED, pages 51–88.
- Oiagen, R. P. (2010). *Avaliação da competitividade em sistemas de produção de bovino-cultura de corte nas regiões sul e norte do Brasil*. Tese de doutorado em zootecnia, Universidade Federal do Rio Grande do Sul.
- Tan, P.-N., Steinbach, M., and Kumar, V. (2005). Association analysis: basic concepts and algorithms. *Introduction to Data mining*, pages 327–414.
- Vieira, F. and Oliveira, S. d. M. (2014). Mineração de dados: conceitos e um estudo de caso sobre certificação racial de ovinos. *Embrapa Informática Agropecuária-Capítulo em livro científico (ALICE)*.
- Witten, I. H., Frank, E., Hall, M. A., and Pal, C. J. (2016). *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann.

Análise da Popularidade de Tuítes com Base em Características Extraídas de seu Conteúdo

Lucas L. de Oliveira¹, Sérgio L. S. Mergen²

¹Centro de Tecnologia – Universidade Federal de Santa Maria (UFSM)
97105-900 -- Santa Maria -- RS — Brasil

²Departamento de Linguagens e Sistemas de Computação
Universidade Federal de Santa Maria (UFSM)

{loliveira,mergen}@inf.ufsm.br

Abstract. Twitter is a powerful platform for opinions diffusion. Knowing this, companies and influential personalities use the platform as a way to connect with their audience, with the goal of boosting their popularity. On Twitter, the tools which are able to measure popularity are the retweets and likes that each post receives. In this paper, the purpose is to analyze the content of the messages transmitted through the platform and correlate them with their popularity. Characteristics of the messages such as the feeling, its size in characters, banality and the use of URLs or hashtags are evaluated. It was possible to identify, in a general way, a preference for tweets with a good feeling and median banality, while the message's size had a different influence on retweets and likes.

Resumo. O Twitter é uma plataforma poderosa para a difusão de opiniões. Sabendo disso, empresas e personalidades influentes usam a plataforma como um meio de se conectar com seu público com o objetivo de alavancar a popularidade. No Twitter, os mecanismos capazes de medir a popularidade são os retuítes e curtidas que cada publicação recebe. Neste trabalho, o objetivo é analisar o conteúdo das mensagens veiculadas através da plataforma e correlacioná-las com sua popularidade. Foram avaliadas características da mensagem como o sentimento, tamanho em caracteres, banalidade e o uso de URLs ou hashtags. Foi possível identificar, de maneira geral, uma preferência do público por tuítes com sentimento positivo e banalidade mediana, enquanto o tamanho das mensagens influenciou de maneira diferente nos retuítes e curtidas.

1. Introdução

O Twitter é um meio de veiculação de mensagens que se destaca por sua simplicidade e objetividade. Segundo o portal de estatísticas Statista¹, é a 11º rede social mais utilizada no mundo, usado por mais de 330 milhões de usuários diariamente.

Uma das preocupações de usuários do Twitter é alavancar sua popularidade, através do aumento no número de seguidores. Essa preocupação é fundamental para empresas e personalidades públicas que utilizam suas imagens para fins monetários. Nesses casos, o uso das redes sociais deve ser planejado e monitorado. Quando isso é realizado da maneira correta, a marca e/ou a pessoa ficam muito mais próximos de seus fãs e seguidores, o que consequentemente, faz sua popularidade e influência aumentar.

¹Statista: <https://www.statista.com/topics/737/twitter/>

Como pode ser observado no trabalho de [Cha et al. 2010], um dos fatores que mede a influência de um usuário do Twitter é a quantidade de retuítes que ele recebe. Levando isso em consideração, pode-se afirmar empiricamente que o aumento na quantidade de retuítes leva a um aumento na quantidade de seguidores, devido a propagação exponencial daquele conteúdo.

Como afirma [Suh et al. 2010], a propagação de um tuíte está diretamente ligada ao conteúdo e valor informativo contido nele. Nesse sentido, os autores avaliaram um conjunto de características extraídas das mensagens. Os resultados mostraram que a presença de *hashtags* e URLs são os fatores que mais ajudam a impulsionar uma publicação. Apesar de ser um resultado relevante, o trabalho não realizou uma análise exaustiva das características que se pode extraír das mensagens.

Nesse contexto, este trabalho realiza uma análise para verificar a influência de determinadas características sobre a popularidade dos tuítes, das quais três delas não foram contempladas pelo estudo de [Suh et al. 2010]: o tamanho em caracteres, o sentimento (que mede a emoção transmitida) e a banalidade (que mede a relevância da mensagem). Para fins de comparação, a presença de *hashtags* e URLs também foi avaliada.

Este artigo está estruturado nas seguintes seções. A seção 2 apresenta os trabalhos relacionados. A seção 3 apresenta a arquitetura de extração de tuítes usada, que realiza desde a coleta até a preparação dos dados para análise. A seção 4 apresenta as análises realizados a partir dos dados coletados. A seção 5 apresenta as considerações finais.

2. Trabalhos Relacionados

Vários autores já apresentaram em seus trabalhos razões pelas quais empresas tornam-se cada vez mais interessadas na utilização de mídias sociais, como Twitter e Facebook. O interesse visa melhorar a comunicação com o consumidor, ou público alvo. Porém, a simples utilização destes serviços online não é suficiente para agregar valor de mercado ao negócio [Culnan et al. 2010]. É preciso de estratégia para que este tipo de intervenção gere bons resultados. Tratando-se da utilização do Twitter, a influência de uma conta pode estar diretamente ligada à relevância do conteúdo por ela disseminado [Valiati et al. 2012].

Nesse sentido, o trabalho de [Suh et al. 2010] realiza uma análise em larga escala (com mais de 74 milhões de registros coletados) sobre fatores que impactam no índice de redistribuição de tuítes. Foram considerados fatores como a utilização de URLs e *hashtags* distintas, quantidade de seguidores e amigos (contas sendo seguidas), o tempo de existência da conta e a quantidade de tuítes antigos do autor. Através de análises observou-se que, com exceção da quantidade de tuítes antigos, os demais fatores interferem na probabilidade de redistribuição. De acordo com os estudos, os fatores que exercem maior influência são o uso de URLs (que pode variar dependendo do domínio), o uso de *hashtags* e a quantidade de seguidores da conta.

O trabalho de [Suh et al. 2010] também propôs a criação de um modelo de predição, elaborado usando a Análise de Componentes Principais (PCA, sigla em inglês para *Principal Components Analysis*) e Modelagem Linear Generalizada (GLM, sigla em inglês para *Generalized Linear Modeling*). O resultado foi um conjunto de coeficientes aplicados em uma equação para prever a taxa de redistribuição dos tuítes. Esse modelo

corrobora com a descoberta sobre a influência das características dos tuítes sobre a taxa de propagação. Uma das características não analisadas pelo trabalho de [Suh et al. 2010] é o sentimento do tuítes. Os trabalhos de [Bigonha et al. 2012] e [Mehta et al. 2012] propõem a detecção do poder de influência do usuário, através de um modelo de cálculo que considera também a análise linguística dos dados coletados e a conexão entre os usuários. Estes estudos mostram que o sentimento pode sim ter uma ligação direta com o poder de influência de um determinado usuário, o que reforça e incentiva a realização do presente trabalho.

Já os trabalhos de [Agarwal et al. 2011] e [Lakshmi et al. 2017] propõem a elaboração de modelos capazes de classificar o sentimento de um tuíte em positivo, negativo ou neutro. Ambos os casos apresentam técnicas em que realizam a coleta, pré-processamento e classificação dos dados. Na etapa de pré-processamento, além de palavras, são também considerados *emoticons*, acrônimos e letras repetidas, o que torna a classificação ainda mais precisa. Já a etapa de classificação foi baseada em modelos diversos, como *Naive Bayes* e *Tree Kernel*. Apesar de não usarem o sentimento para nenhum tipo de medição, os modelos são relevantes como uma forma alternativa de extração de características.

Outros trabalhos baseados em análises usando dados do Twitter são os de [Engel 2016] e [Tumasjan et al. 2010]. Ambos tem o intuito de relacionar a opinião dos usuários no Twitter com as eleições em seus países (Estados Unidos e Alemanha, repescivamente). O trabalho desenvolvido por [Engel 2016] busca distinguir e exibir em tempo real o sentimento da população, baseada na sua localização, quanto aos candidatos a presidência. Já o trabalho de [Tumasjan et al. 2010] analisa a influência que a plataforma tem sobre as eleições. Neste trabalho realiza-se uma análise das mensagens considerando 12 dimensões do sentimento político. Esta análise é realizada através do software LIWC2007 (*Linguistic Inquiry and Word Count*), que avalia componentes emocionais, cognitivos e estruturais de amostras de texto. Tumasjan pôde concluir que estudos com dados do Twitter de fato servem como preditores do resultado de eleições, chegando perto até mesmo das pesquisas tradicionais.

3. Proposta

O propósito deste trabalho é correlacionar a popularidade dos tuítes em função de um conjunto de características. Para atingir esse objetivo, é necessário coletar os tuítes e extrair suas características. Esta seção apresenta a arquitetura de coleta e extração utilizada neste trabalho.

A arquitetura, exibida na Figura 1, tem os seguintes módulos: (a) **Coleta** dos tuítes publicados por cada uma das contas acompanhadas; (b) **Extração** das características de cada tuíte; e (c) **Atualização** periódica dos dados coletados.

3.1. Coleta de tuítes

O módulo de coleta é responsável por extrair tuítes de usuários específicos. A extração ocorre de forma contínua, usando recursos de *streaming* disponibilizados pela API do Twitter². São coletados todos tuítes publicados a partir do momento que o *streaming* entra em execução.

²Twitter Developer Platform: <https://dev.twitter.com>

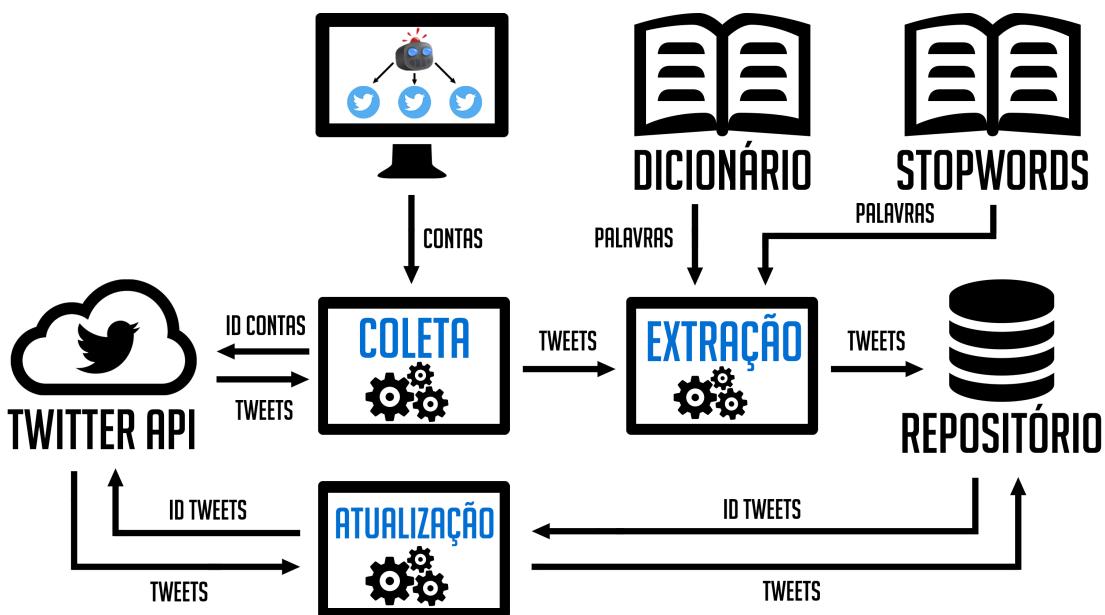


Figura 1. Arquitetura adotada no desenvolvimento do projeto

A especificação das contas a serem seguidas é feita a partir de uma conta raiz. São extraídos os tuítes de todos usuários seguidos por essa conta raiz. Essa estratégia permite que novas contas sejam adicionadas à lista sem que haja interrupções. O módulo também conta com tratamento de exceções para que a coleta não seja interrompida devido à problemas temporários de acesso aos dados, como indisponibilidade do serviço ou extrapolação do limite de requisições permitido por instante de tempo.

Na Tabela 1 podem ser visualizadas as informações extraídas de cada tuíte através da API. O campo “mensagem” é usado para a extração das características. Já os “retuítes” e “curtidas” são usados como forma de medir a popularidade do tuíte. Por sua vez, os campos “identificação” e “data/hora” são usados pelo módulo de atualização.

Tabela 1. Dados coletados para cada tuíte

Informação	Conteúdo
autor	código e nome da conta que originou o tuíte
identificação	código do tuíte (permite a consulta posterior)
mensagem	texto de no máximo 280 caracteres
data e hora	data e hora da publicação do tuíte em seu país de origem
retuítes	quantidade de retuíte que a mensagem recebeu
curtidas	quantidade de vezes que o tuíte foi favoritado

3.2. Extração de características de tuíte

Esta etapa corresponde a extração das características de cada um dos tuítes coletados. A extração ocorre imediatamente após a coleta. Os itens abaixo mostram como cada característica foi extraída.

Presença de URLs e hashtags: O uso desses recursos na mensagem é detectado pela presença de prefixos específicos no corpo da mensagem. Por exemplo, o prefixo

“http” indica que URLs foram usadas. Já o prefixo “#” denota o uso de *hashtags*.

Tamanho da mensagem: O tamanho é a contagem da quantidade de caracteres usados no corpo do tuíte. A contagem desconsidera caracteres usados em URLs, assumindo que *hyperlinks* não transmitam nenhuma mensagem. A remoção de URLs foi realizada a partir da aplicação de uma expressão regular.

Extração do sentimento: O sentimento de uma mensagem é um valor que classifica o teor da mensagem como positivo ou negativo. Para realizar a extração do sentimento, assim como no trabalho de [Engel 2016], foi utilizada a biblioteca TextBlob da linguagem Python [Loria et al. 2014]. Essa biblioteca permite a obtenção da polaridade e subjetividade de conteúdos textuais na língua inglesa. A API também fornece a possibilidade de tradução do conteúdo de textos escritos em outras linguagens.

A extração do sentimento se baseia em Árvores de Decisão e no modelo de classificação *Naive Bayes*, que também é utilizado no trabalho de [Lakshmi et al. 2017]. A classificação da mensagem retorna um valor decimal entre o intervalo de -1 e 1, onde -1 corresponde a uma mensagem totalmente negativa, 0 corresponde a neutra e 1 corresponde a totalmente positiva.

Extração da banalidade: No contexto deste trabalho, a banalidade corresponde à importância do que foi escrito. A forma adotada para medir a banalidade leva em consideração a presença de palavras que são frequentemente usadas em textos escritos. Quanto maior o número de palavras frequentes, mais banal é a mensagem.

A Equação 1 mostra como computar o valor da banalidade:

$$\frac{\sum_{i=1}^n(freq(P_i))}{n} \quad (1)$$

onde o conjunto $\{P_1, \dots, P_n\}$ são as palavras da mensagem após a remoção de *stopwords* (preposições e artigos que normalmente são descartados durante o processamento de um texto). Já a função $freq(P)$ retorna 1 caso a palavra P seja frequente e zero caso não seja. A verificação da frequência utiliza um dicionário de palavras previamente construído. Também são removidas as *hashtags* e menções a outros usuários, por entender que não se tratam de palavras que podem ser caracterizadas como banais ou não.

Na Tabela 2 pode ser visto um exemplo dos dados extraídos nesta etapa.

Tabela 2. Dados obtidos na etapa de Extração

Informação	Conteúdo
sentimento	valor entre -1 e 1 correspondente a polaridade do texto
URL	valor 1 se houver URL no texto e 0 se não houver
#hashtag	valor 1 se houver hashtag no texto e 0 se não houver
tamanho	quantidade de caracteres utilizados na mensagem
banalidade	somatório baseado na no uso de palavras frequentes

3.3. Atualização dos dados de retuítes e curtidas

Como o módulo de coleta funciona por meio de *streaming*, os tuítes são coletados no instante de sua criação. Nesse momento, a quantidade de retuítes e curtidas recebidos têm

o valor zero. Dessa forma, é necessária uma conferência periódica para a obtenção dos dados atualizados.

A atualização é realizada através de um recurso da API do Twitter que obtém informações de um tuíte a partir do seu código de identificação. Para evitar sobrecarga de processamento, apenas os tuítes publicados no intervalo de uma semana são atualizados. Como o status de tuítes mais antigos é raramente modificado, o acesso a eles seria ao mesmo tempo custoso e improdutivo.

4. Resultados

Esta seção apresenta análises que correlacionam características extraídas de tuítes com a sua popularidade. A importância das características é medida por dois indicadores: o número de retuítes e o número de curtidas.

A coleta de tuítes foi realizada tendo como base as contas de personalidades influentes que utilizam o Twitter periodicamente. Ao todo, foram usadas 23 contas de diversas áreas de atuação, como por exemplo Donald J Trump (atual presidente dos Estados Unidos), Jimmy Fallon (famoso apresentador de TV americano) e Katy Perry (cantora detentora da conta com o maior número de seguidores no Twitter). A escolha deve-se ao fato de que a análise do impacto de publicações em redes sociais é mais relevante para esse tipo de usuário. A etapa de coleta permaneceu em execução durante o período entre Novembro de 2017 e Fevereiro de 2018, totalizando cerca de 5500 registros.

Uma análise inicial sobre os dados gerou constatações que serviram para orientar o trabalho. Um dado interessante é que o número de curtidas é muito superior ao número de retuítes. Assim, esses dois indicadores são analisados de forma independente.

Outro dado interessante é que algumas características são mais empregadas do que outras. Por exemplo, 78% dos tuítes coletados possuem URLs em seu conteúdo, enquanto apenas 28% possuem hashtags. Para evitar distorções causadas por esse desbalanceamento, os indicadores por característica são normalizados. A Equação 2 mostra como normalizar o número de curtidas.

$$\frac{\sum_{i=1}^n (\text{count_likes}(T_i))}{n} \quad (2)$$

O cálculo usa o conjunto de tweets $\{T_1, \dots, T_n\}$ onde a característica apareça. A função $\text{count_likes}(T)$ retorna o número de curtidas atribuídos ao tuíte T . Ou seja, o valor final corresponde a soma de todas as curtidas recebidas dividida pela quantidade de tuítes. A normalização do número de retuítes se baseia no mesmo princípio.

4.1. Popularidade x Existência de URL ou hashtag

O objetivo deste experimento é verificar se o uso de hashtags e/ou URLs contribui (de forma positiva ou negativa) para alavancar a popularidade de um tuíte.

A Figura 2 apresenta os resultados. Como pode-se ver na Figura 2(b), tuítes que não usaram hashtags foram mais populares (em retuítes e curtidas) do que aqueles que usaram esse marcador. O mesmo não se pode dizer quanto ao uso de URLs. Nesse caso, Figura 2(a), não há uma diferença significativa no número de retuítes. Já o número de curtidas foi menor para mensagens que não usaram URLs.

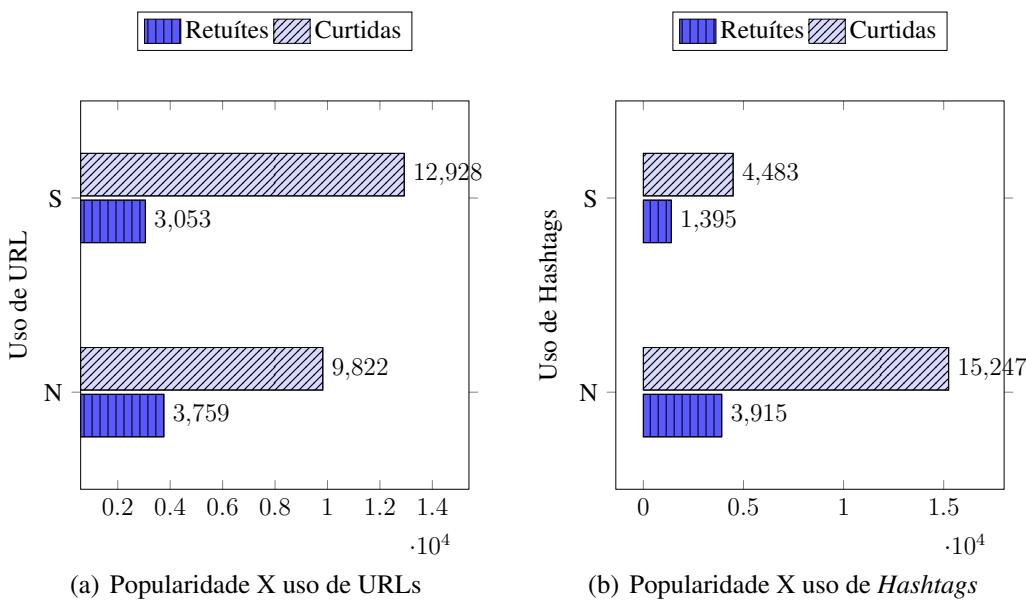


Figura 2. Análise da taxa de popularidade pelo uso de URLs e hashtags. No eixo Y, N corresponde a Não e S corresponde a Sim

Apesar de pequena a diferença, esse resultado reforça o estudo publicado por [Suh et al. 2010], que considera a presença de URLs importante para que um tuíte seja redistribuído. Essa pequena diferença nos resultados pode ser atribuída ao uso de dados diferentes. Enquanto [Suh et al. 2010] usou dados de usuários aleatórios, neste trabalho os usuários foram escolhidos com base na sua popularidade. Assim, é possível que essa característica dependa do perfil do autor da publicação.

Em uma análise individual das contas, observou-se que esse comportamento pode ser inverso, onde o tuítes sem a presença de URLs recebem um maior número de curtidas e retuítés. Este foi o caso da conta do presidente dos Estados Unidos, Donald J. Trump.

4.2. Popularidade x Tamanho de Mensagem

O objetivo deste experimento é identificar a existência de uma relação entre a popularidade do tuíte e seu tamanho, em caracteres. Dessa forma, é possível saber se existe uma preferência por textos mais extensos ou mais enxutos.

A Figura 3 apresenta os resultados. O resultado mostra claramente que mensagens curtas, marcadas com “X” na Figura 3(b), receberam mais curtidas. A exceção ocorre para textos com tamanhos variando de 120 a 130 caracteres. Curiosamente, o número de curtidas nesse caso foi bastante superior aos demais tamanhos de texto. Também chama a atenção o fato que são mensagens cuja extensão se aproxima do tamanho máximo de texto que a plataforma do Twitter costumava disponibilizar (140).

Já o número de retuítés aparece bem distribuído em todas faixas de tamanho. Mesmo assim, percebe-se uma concentração maior em textos de tamanho próximo aos 140 caracteres, destacados com “X” na Figura 3(a). O motivo pode estar relacionado ao hábito de usar textos dentro desse intervalo. Corrobora para essa tese a constatação de que, mesmo tendo o tamanho máximo aumentado para 280 caracteres, as mensagens mais extensas coletadas não ultrapassaram os 190 caracteres.

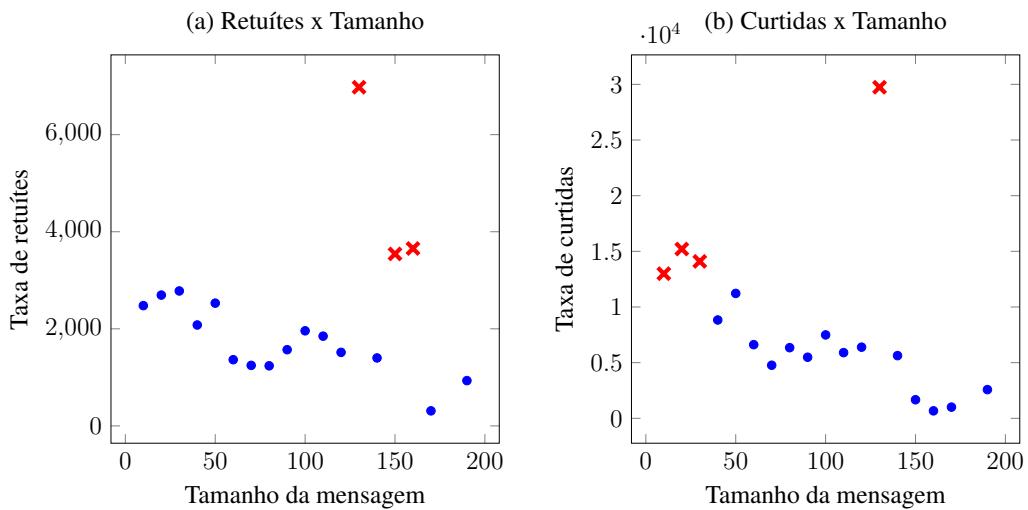


Figura 3. Análise da taxa de popularidade pelo tamanho de mensagens

4.3. Popularidade x Polaridade de Sentimento

Neste experimento, a popularidade do tuíte é confrontada com a polaridade do sentimento. Dessa forma, é possível saber a preferência dos seguidores por um conteúdo mais bem humorado ou mal humorado.

A Figura 4 apresenta os resultados. A polaridade do sentimento foi considerada até uma casa decimal para facilitar a visualização e desfragmentar as faixas de sentimento. Os resultados demonstram que tuítes “mal humorados” são os menos populares, tanto em número de curtidas quanto em número de retuítos. De 20% dos registros com menor popularidade, destacados com “X” nos gráficos, 3/4 deles estão no extremo negativo.

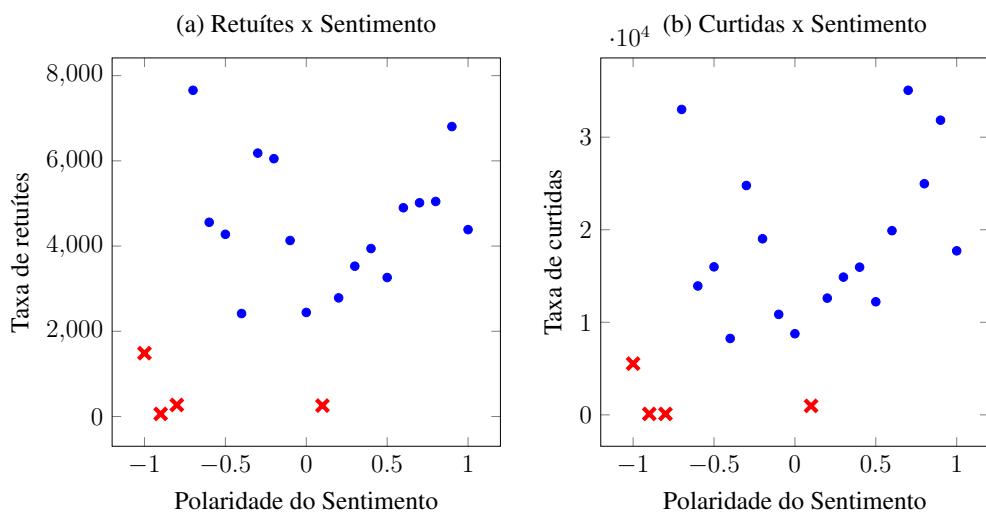


Figura 4. Análise da relação entre popularidade e polaridade sentimento.

Vale destacar, que na análise individual das contas, uma delas apresentou um comportamento no qual 25% dos tuítes com a menor taxa de polaridade encontrava-se nos dois extremos, tanto negativo quanto positivo (que também foi o caso da conta de Donald J. Trump). Apesar de esse resultado individual apontar para outra direção, ainda assim é

possível perceber que o sentimento exerce influência na popularidade, muito embora a correlação possa variar de conta para conta.

4.4. Popularidade x Banalidade da mensagem

Neste experimento o intuito é identificar se tuítes que fazem maior uso de palavras frequentes recebem mais ou menos retuítes e curtidas. Neste processo, foram consideradas apenas as mensagens escritas na língua inglesa. O dicionário usado possui 3.000 palavras frequentes desta linguagem³.

A Figura 5 apresenta os resultados. Como pode-se ver, tanto os tuítes com taxa baixa e alta de banalidade são os menos populares, os quais também foram destacados com “X” nos gráficos. A impopularidade dos tuítes banais pode estar relacionada à ausência de originalidade das mensagens. Já a impopularidade dos tweets que estejam no outro extremo pode estar associada ao uso de uma linguagem pouco acessível ao público.

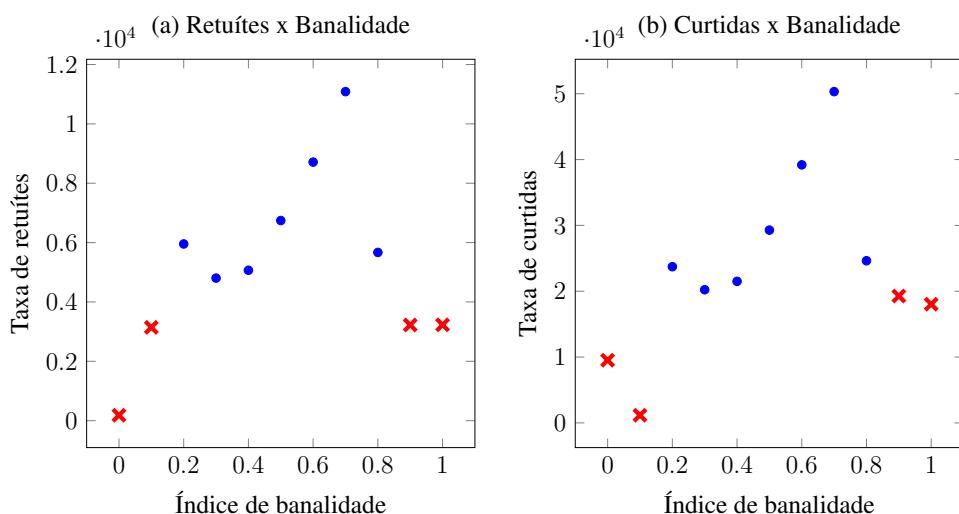


Figura 5. Análise da relação entre popularidade e banalidade da mensagem.

É interessante também analisar como as características aliadas influenciam na importância atribuída a um tuíte. Por exemplo, a Figura 3 sugere que tuítes curtos são os mais curtidos. Por outro lado, tuítes banais são pouco curtidos. Assim, possivelmente a importância de tuítes curtos seja intensificada caso seu conteúdo não seja banal. A verificação de hipóteses desse tipo depende de uma análise conjunta das características extraídas, o que foge do escopo deste artigo.

5. Considerações Finais

Medir a variação do índice de popularidade pode ser do interesse de administradores de grandes contas do Twitter, pois pode indicar o sucesso ou fracasso de uma determinada campanha realizada ou a dimensão de um escândalo e, tendo consciência disso, ações preventivas ou corretivas podem ser tomadas e sua repercussão pode ser monitorada. Tratando-se de personalidades públicas, é importante identificar quais assuntos e/ou

³3000 most common words in English: <https://www.ef.com/english-resources/english-vocabulary/top-3000-words/>

abordagens agradam mais o público-alvo para que assim possam ser mantidas ou evitadas. Assim, pode ser relevante entender quais características de um tuíte publicado interferem no interesse do público por aquele conteúdo.

Apesar de incipientes, os resultados alcançados mostram que as características avaliadas parecem exercer influência no nível de interesse, e podem ser levadas em consideração na elaboração de estratégias que visem impulsionar publicações.

Como trabalhos futuros, pretende-se avaliar a taxa de popularidade quando as características são combinadas entre si. Por exemplo, aliar a taxa de banalidade da mensagem com o seu tamanho pode fornecer um indicador de popularidade importante. Além disso, outra possibilidade de trabalho futuro envolve a elaboração de preditores de popularidade baseados nas características estudadas. Com preditores específicos para cada conta, seria possível verificar o alcance de uma determinada mensagem antes mesmo de publicá-la.

Referências

- Agarwal, A., Xie, B., Vovsha, I., Rambow, O., and Passonneau, R. (2011). *Sentiment Analysis of Twitter Data*. In *Proceedings of the Workshop on Languages in Social Media*, LSM '11, pages 30–38, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Bigonha, C., Cardoso, T. N. C., Moro, M. M., Gonçalves, M. A., and Almeida, V. A. F. (2012). *Sentiment-based influence detection on Twitter*. *Journal of the Brazilian Computer Society*, 18(3):169–183.
- Cha, M., Haddadi, H., Benevenuto, F., and Gummadi, P. K. (2010). Measuring user influence in twitter: The million follower fallacy. *Icwsrm*, 10(10-17):30.
- Culnan, M., J. McHugh, P., and I. Zubillaga, J. (2010). *How Large U.S. Companies Can Use Twitter and Other Social Media to Gain Business Value*. 9:243–259.
- Engel, A. (2016). Election 2016 twitter sentiment map.
- Lakshmi, V., Harika, K., Bavishya, H., and Harsha, C. S. (2017). *SENTIMENT ANALYSIS OF TWITTER DATA*.
- Loria, S., Keen, P., Honnibal, M., Yankovsky, R., Karesh, D., Dempsey, E., et al. (2014). Textblob: simplified text processing. *Secondary TextBlob: Simplified Text Processing*.
- Mehta, R., Mehta, D., Chheda, D., Shah, C., and Chawan, P. M. (2012). *Sentiment analysis and influence tracking using twitter*. *International Journal of Advanced Research in Computer Science and Electronics Engineering (IJARCSEE)*, 1(2):pp–72.
- Suh, B., Hong, L., Pirolli, P., and Chi, E. H. (2010). *Want to be retweeted? large scale analytics on factors impacting retweet in twitter network*. In *Social computing (social-com), 2010 ieee second international conference on*, pages 177–184. IEEE.
- Tumasjan, A., Sprenger, T. O., Sandner, P. G., and Welpe, I. M. (2010). Predicting elections with twitter: What 140 characters reveal about political sentiment. *Icwsrm*, 10(1):178–185.
- Valiati, H., Silva, A., Guimaraes, S., and Meira Jr, W. (2012). *Detecção de Conteúdo Relevante e Usuários Influentes no Twitter*.

aper:180199_1

Extração de elementos textuais em imagens capturadas por smartphones: análise da relação entre as características das imagens e a eficácia da extração

Daniel M. Kuhn¹, Cristiano R. Cervi¹, Edimar Manica²

¹Instituto de Ciências Exatas e Geociências – UPF - Passo Fundo - RS

²Campus Ibirubá - IFRS - Ibirubá - RS

138714@upf.br, cervi@upf.br, edimar.manica@ibiruba.ifrs.edu.br

Abstract. Optical character recognition software is designed to convert textual elements of documents into editable and searchable text. This task presents specific challenges when submitted to images captured by smartphone cameras. This work experimentally analyzes the relationship between extraction efficiency in images captured by smartphones and their characteristics. The experiments demonstrate that images with curvilinear text and light variation do not substantially compromise extraction efficiency, whereas images with inclined texts as well as images with unclear characters have the lowest extraction rates.

Resumo. Softwares de reconhecimento óptico de caracteres têm como propósito converter elementos textuais de documentos em texto editável e pesquisável. Essa tarefa apresenta desafios específicos quando submetida a imagens capturadas por câmeras de smartphones. Este trabalho analisa experimentalmente a relação entre a eficácia de extração em imagens capturadas por smartphones e suas características. Os experimentos demonstram que imagens com texto curvilíneo e variação de iluminação não comprometem substancialmente a eficácia de extração, ao instante que imagens com textos inclinados, bem como imagens que possuem caracteres pouco nítidos, apresentam os menores índices de extração.

1. Introdução

A análise de Big Data é um aspecto chave da sociedade moderna uma vez que permite criar conhecimento a partir de dados. Essa análise traz o conhecimento para o indivíduo de uma forma direta e facilitada permitindo a emancipação das pessoas e as habilitando a agirem e tomarem decisões com mais embasamento [Manica, Dorneles and Galante 2017]. Problemas de heterogeneidade, escalabilidade, complexidade e privacidade impedem o progresso de todos os estágios do *pipeline* que extrai valor a partir de dados [Labrinidis and Jagadish 2012]. Nesse contexto, os problemas iniciam durante a aquisição de dados porque muitos dados não estão nativamente em um formato estruturado e estruturar tal conteúdo para análise futura é o principal desafio [Agrawal et al 2012].

Um exemplo de dados relevantes em um formato não estruturado é observado em textos presentes em imagens postadas nas redes sociais. Estima-se que só no Instagram - atualmente a maior rede social de fotografias - são postadas em média 52 milhões de fotografias todos os dias [Statistic Brain, 2017]. A extração de elementos textuais contidos em imagens de trechos de livros postadas na rede social pode ser útil para identificar o que os usuários estão lendo e então recomendar outros livros semelhantes.

A extração de conteúdos textuais em imagens é realizada através do uso de softwares de Reconhecimento Óptico de Caracteres (OCR – *Optical Character Recognition*). O OCR é um processo de reconhecimento visual que converte documentos de texto em texto editável e pesquisável [Berchmans and Kumar 2014]. Nos últimos trinta anos um número substancial de pesquisas acerca de mecanismos de OCR foram realizadas [Islam and Noor 2016]. Em suma, a grande maioria dos esforços destinou-se a solucionar problemas decorrentes da digitalização de documentos de texto através do uso de dispositivos de *scanner*, o que resultou na obtenção de altas taxas de precisão de extração em documentos desta natureza [Asad et al 2016].

Entretanto, os métodos de pré-processamento de imagens aplicados em documentos escaneados são em diversos casos inapropriados ou insuficientes quando destinados a otimizar o reconhecimento de caracteres de imagens capturadas por câmeras de *smartphones*. Isso se deve ao fato de que as características encontradas em imagens escaneadas são, em sua grande maioria, distintas das características presentes em arquivos de imagens obtidas através da câmera de *smartphones*. As imagens capturadas por câmeras podem apresentar baixa resolução, desfocagem e distorção de perspectiva, apresentando layouts complexos e interação entre o conteúdo e o plano de fundo [Liang, Doermann and Li 2005].

Este trabalho tem como objetivo geral avaliar experimentalmente a eficácia da extração de um software de OCR em imagens capturadas por câmeras de *smartphones*. O objetivo específico é relacionar as características das imagens com a eficácia da extração. De acordo com os experimentos realizados, as características que mais impactam a eficácia da extração são: (i) linhas de texto inclinadas; (ii) caracteres pouco nítidos.

Este artigo está organizado da seguinte forma. Na Seção 2, são discutidos os trabalhos relacionados. A Seção 3 descreve a metodologia dos experimentos. Na Seção 4, são discutidos os resultados dos experimentos. Finalmente, a Seção 5 apresenta as considerações finais e os trabalhos futuros.

2. Trabalhos relacionados

O trabalho de [Asad et al 2016] apresenta um sistema de OCR baseado em redes LSTM (*Long Short Term Memory*), capaz de reconhecer caracteres borradinhos decorrentes de movimentos indesejados. Redes LSTM, são um tipo especial de redes neurais recorrentes (RNN – *Recurrent neural network*) com capacidade de recordar informações por longos períodos de tempo [Olah 2015].

Em [Smith 1987] foi proposta uma nova abordagem para reconhecimento de

caracteres. Esse trabalho deu origem ao motor de OCR *Tesseract* [Tesseract 2015]. Em 2005, *Tesseract* passou a ser um projeto *Open Source* e desde 2006 vem sendo desenvolvido pela *Google Inc.* *Tesseract* provê suporte a Unicode, capaz de reconhecer mais de 100 linguagens diferentes [Tesseract 2015]. *Tesseract* é totalmente treinável, sendo possível adicionar novos símbolos e até mesmo novos idiomas inteiros. A possibilidade de aplicar processos de treinamento, bem como, o fato de ser um projeto *Open Source*, foram fatores determinantes para a escolha do *Tesseract* como extrator.

Em [Kuhn, Cervi and Manica 2017] foi avaliada a eficácia da extração de elementos textuais em imagens capturadas por *smartphones* submetidas ao *Tesseract*. Este trabalho diferencia-se por expandir os experimentos e realizar uma análise dos resultados identificando a relação entre as características das imagens e a eficácia da extração.

3. Metodologia

Nesta seção, é descrita a metodologia adotada nos experimentos. A Figura 1 apresenta o fluxo de execução dos experimentos, composto por 6 etapas:

1. **Obtenção** - onde foram coletadas as imagens para compor a base de dados dos experimentos;
2. **Anotação** - onde foram transcritos manualmente os elementos textuais das imagens;
3. **Definição** - onde definiu-se as características de interesse a serem analisadas;
4. **Identificação** - onde identificou-se a presença das características definidas nas imagens da base de dados;
5. **Configuração e execução** - onde submeteu-se a base de dados a um extrator para obter os elementos textuais contidos na imagem de forma automática;
6. **Análise** - onde os resultados obtidos foram analisados, relacionando a eficácia da extração com as características das imagens.

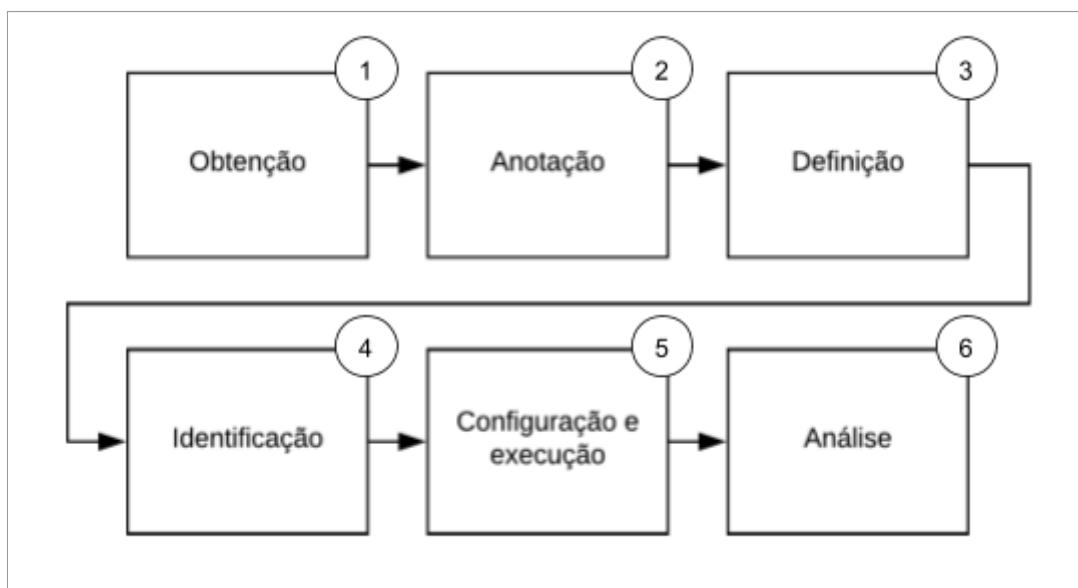


Figura 1. Etapas dos experimentos realizados.

As subseções a seguir detalham as etapas que integram os experimentos.

3.1. Obtenção

Para a obtenção da base de dados, foram coletadas 160 imagens de trechos de livros capturadas por *smartphones*. Desse total, 82 imagens (51.25%) foram coletadas manualmente de publicações de redes sociais. As 78 amostras restantes (48.75%) foram fornecidas por um grupo de voluntários, que possuíam *smartphone* e tinham idade entre 14 e 29 anos.

Os voluntários receberam a tarefa de fotografar pequenos trechos de um livro qualquer utilizando seus *smartphones*. Após, deveriam recortar o trecho, segmentando entre toda a imagem, o trecho de real interesse. Por fim, deveriam enviar a imagem resultante por e-mail ou rede social. Ressalta-se que alguns voluntários não segmentaram a imagem corretamente, ou seja, algumas imagens possuem ruído.

3.2. Anotação

O conteúdo textual contido em cada uma das 160 imagens foi manualmente transscrito por um especialista. O texto em formato digital foi então armazenado na base de dados, mantendo a devida relação entre o arquivo da imagem e seu respectivo conteúdo textual. Dessa forma, gerou-se o gabarito de extração. Todas as imagens estavam legíveis o suficiente para permitir o reconhecimento manual de todas as palavras nelas contidas.

A Figura 2 (a) apresenta um exemplo de imagem da base de dados. A Figura 2 (b) apresenta o gabarito para essa imagem, ou seja, os elementos textuais identificados pelo especialista. Observa-se que essa imagem não foi segmentada corretamente, uma vez que possui caracteres que não pertencem ao texto de interesse.

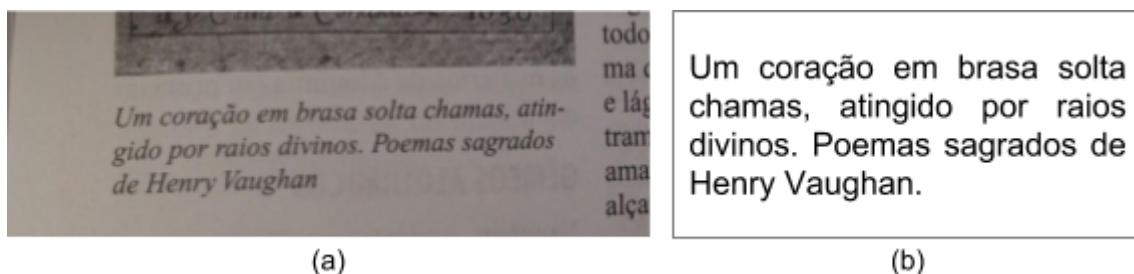


Figura 2: Exemplo de imagem da base de dados e seu respectivo gabarito

3.3. Definição

Nesta etapa, foram selecionadas as características das imagens a serem avaliadas com relação ao seu impacto na eficácia da extração. Foram definidas quatro características: (i) linhas de texto inclinadas; (ii) linhas de texto com aspecto curvilíneo; (iii) variação de iluminação; e (iv) caracteres pouco nítidos. A seguir, cada característica é explicada e exemplificada.

3.3.1. Linhas de texto inclinadas

Esta característica refere-se ao aspecto de inclinação das linhas da imagem. Conforme pode-se observar na Figura 3, a imagem pode estar excessivamente rotacionada no

sentido horário, ou seja, possui graus de inclinação negativos, ou então, pode estar excessivamente rotacionada no sentido anti-horário, e nesse caso, apresenta graus de inclinação positivos.

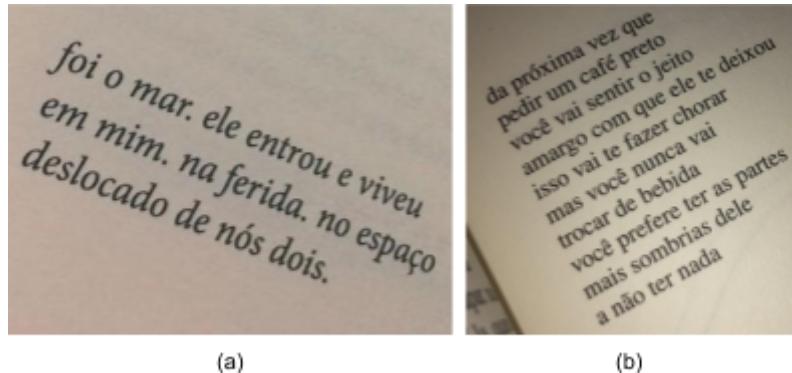


Figura 3: Exemplo de imagens apresentando graus de inclinação: (a) imagem com grau de inclinação negativo, (b) imagem com grau de inclinação positivo.

Essa característica resulta do posicionamento da câmera em relação ao texto a ser fotografado. Ao contrário das imagens escaneadas, onde há a presença de um suporte que sugere a posição correta, as fotografias não possuem posicionamento pré-definido.

3.3.2. Linhas de texto com aspecto curvilíneo

Esta característica é resultado decorrente da perspectiva das páginas em relação à câmera. Visto que as páginas dos livros tendem a curvar-se, quando fotografadas dessa maneira, produzem imagens com distintas perspectivas ao longo da página. Como resultado, as linhas de texto tendem a apresentar aspectos curvilíneos.

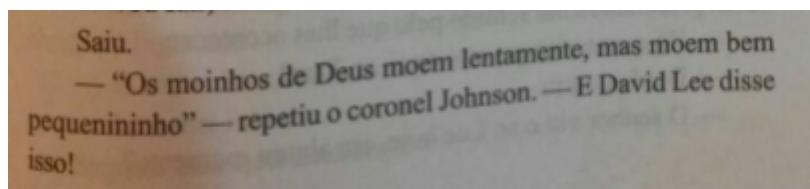


Figura 4. Exemplos de imagens apresentando linhas de texto com aspecto curvilíneo

Pode-se observar na Figura 4 que a imagem apresenta aspecto curvilíneo.

3.3.3. Variação de iluminação

Esta característica refere-se a variações de iluminação sobre a imagem. Nessa categoria, são incluídas também, imagens com presença total ou parcial de sombras.

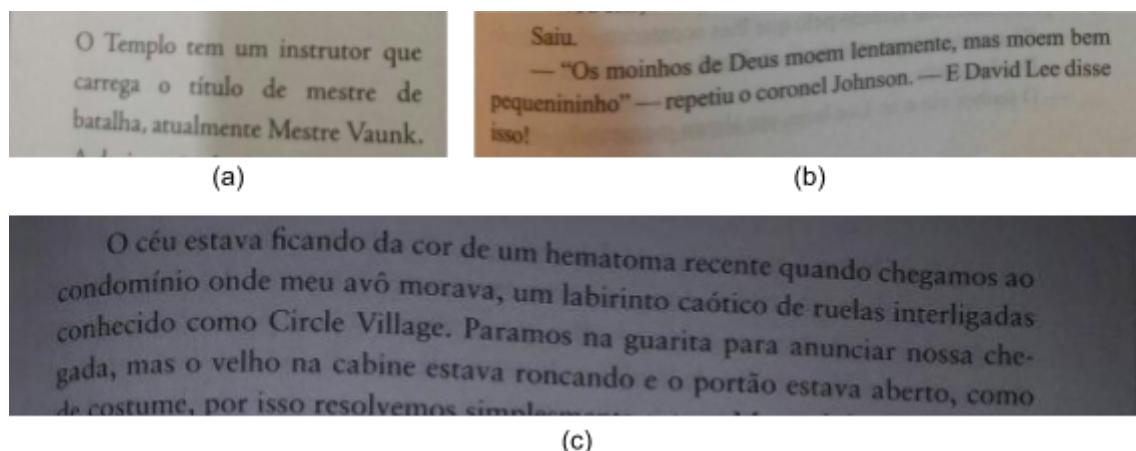


Figura 3. Exemplo de imagens com variação de iluminação

As imagens apresentadas na Figura 3 (a) e 3 (b) apresentam presença parcial de sombra sobre o texto de interesse. A imagem apresentada na Figura 3 (c) apresenta presença de sombra sobre todo o texto de interesse.

3.3.4. Caracteres pouco nítidos

Foram consideradas imagens com caracteres pouco nítidos, aquelas que possuíam caracteres desfocados ou borrados. Essa característica decorre de diversos fatores, entre eles: (i) qualidade da câmera; (ii) foco da imagem; (iii) iluminação - abordado neste trabalho como variação de iluminação; (iv) perspectiva - também abordado neste trabalho com a denominação de linhas de textos com aspecto curvilíneo.

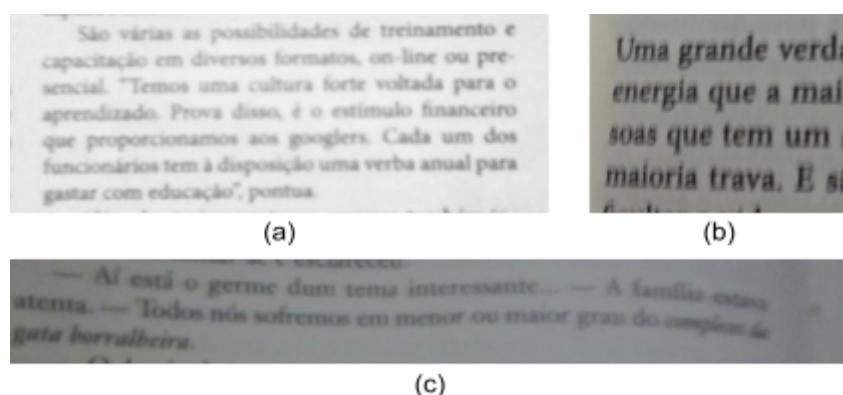


Figura 4: Exemplo de múltiplos cenários relacionados à pouca nitidez dos caracteres

A Figura 4 demonstra imagens contendo diferentes aspectos em que há ocorrência de caracteres pouco nítidos. A Figura 4 (a) foi fotografada com um dispositivo com câmera de baixa qualidade. Os caracteres da Figura 4 (b) apresenta efeito fantasma, decorrente de movimentos indesejados no momento da fotografia. A Figura 4 (c) apresenta uma imagem não focalizada.

4. Identificação

Nesta etapa, identificou-se entre as características de interesse, quais estavam presentes nas imagens da base de dados. A identificação de características nas imagens foi dividida em processos manuais e automatizados. Como processos manuais, com possibilidade binária de resposta (Afirmativo ou Negativo) as seguintes questões foram respondidas por um usuário especialista: (i) as linhas de texto apresentam aspectos curvilíneos?; (ii) é possível identificar variações de iluminação ou presença parcial ou total de sombra sobre a imagem?; (iii) os caracteres do conteúdo textual estão nítidos?. Este processo foi repetido para as 160 imagens.

Para a identificação da inclinação das linhas de texto, utilizou-se um algoritmo¹ capaz de identificar o número (em graus) de inclinação das linhas de texto. O algoritmo utiliza métodos de transformação morfológicas para evidenciar os pixels que formam o segmento da linha de texto e utiliza o método conhecido como Transformada de Hough [OpenCV 2017] para identificar o segmento das linhas de texto com base no número de pontos (*pixels*) em uma reta. Em seguida, calcula-se o grau de inclinação formada pela linha identificada. Foram consideradas imagens rotacionadas, aquelas que apresentaram graus de inclinação fora do intervalo [-4, 4] graus.

Tabela 1. Ocorrência das características de interesse na base de dados

Característica	Ocorrência na base	Ocorrência na base (%)
Linhas de texto inclinadas	23 (15)	14,38% (9,38%)
Linhas de texto curvilíneas	30 (10)	18,75% (6,25%)
Variação de iluminação	65 (42)	40,63% (26,25%)
Caracteres pouco nítidos	28 (16)	17,5% (10%)
Ausência das características de interesse	48	30%

A Tabela 1 apresenta a ocorrência de cada característica na base de dados. Observa-se que o maior percentual (40,63%) refere-se a imagens com variação de iluminação. Este percentual decorre pelo fato de que, ao contrário dos *scanners*, que possuem mecanismos especialmente projetados para iluminar a superfície da imagem de maneira uniforme, as fotografias obtidas através de câmeras estão suscetíveis a variações de iluminação do ambiente. A menor incidência refere-se às imagens com linhas de texto inclinadas, são 23 imagens correspondendo a 14,38% das amostras. As imagens que não apresentam nenhuma das características de interesse, representam 30% da base.

É preciso ressaltar que há imagens que apresentam mais de uma característica, visto que a base de dados é constituída de imagens reais. Dessa forma, para realizar o experimento, considerou-se apenas as imagens que possuíam uma única característica de interesse, as quais estão representadas entre parênteses na segunda coluna da Tabela 1.

¹ Algoritmo de autoria de Vecsei (2016). Disponível em: <<https://github.com/gaborvecsei/Straighten-Image>>

4.1. Configuração e extração

O extrator utilizado no experimento foi o *Tesseract*. A versão utilizada no experimento foi a versão 3.04, compilada e construída fazendo uso do conjunto de ferramentas *Android NDK*². O modo de extração foi definido como *OEM_DEFAULT*, utilizando os arquivos padrão de dados do idioma Português Brasil³.

Optou-se por utilizar o *Tesseract* em um *smartphone*, objetivando desenvolver uma aplicação cliente capaz de realizar a extração sem a necessidade de requisitar o serviço de extração a um servidor Web.

Após construída e configurada a aplicação, as imagens foram submetidas isoladamente ao *Tesseract*, armazenando o resultado obtido da extração junto à base de dados.

4.2. Análise

Para avaliar a eficácia da extração, foram adotadas as métricas tradicionais de recuperação de informação: precisão (*precision*), revocação (*recall*) e F1 (*F-measure*). Para este trabalho, contextualizando a definição de Precisão e Revocação apresentada em [Baeza-Yates and Ribeiro-Neto 2013], assumimos que a precisão mede a fração dos termos recuperados que é relevante e a revocação mensura a fração dos termos relevantes que foi recuperada. Portanto:

$$\begin{aligned} \text{precisão} &= \frac{|\text{Termos relevantes} \cap \text{Termos recuperados}|}{|\text{Termos recuperados}|} \\ \text{revocação} &= \frac{|\text{Termos relevantes} \cap \text{Termos recuperados}|}{|\text{Termos relevantes}|} \end{aligned}$$

Um termo relevante é aquele que foi extraído de uma imagem corretamente e, portanto, está presente no gabarito. Por fim, o *F1* consiste na média harmônica entre os índices de precisão e revocação, com o objetivo de fornecer um só índice de medida.

$$F1 = 2 \cdot \frac{\text{precisão} \cdot \text{revocação}}{\text{precisão} + \text{revocação}}$$

Ressalta-se que acentuações, distinção de letras maiúsculas e minúsculas, bem como, pontuações foram desconsideradas na análise. Os resultados obtidos são discutidos na seção que segue.

5. Experimento e Resultados

Esta seção apresenta o experimento realizado, bem como os resultados obtidos a partir deste experimento. O objetivo do experimento foi responder a questão: **qual o impacto de cada característica na eficácia da extração?** Para isso foi analisada a F-measure da extração em imagens com as características de interesse.

² O conjunto de ferramentas Android NDK permite a implementação de partes de aplicativos fazendo uso de linguagens de código nativas como C/C++.

³ Pacote de treinamento padrão para o idioma Português Brasil para a versão 3.04 do *Tesseract*. Disponível em: <<https://github.com/tesseract-ocr/langdata/tree/master/por>>. Acessado em 16 de agosto de 2017.

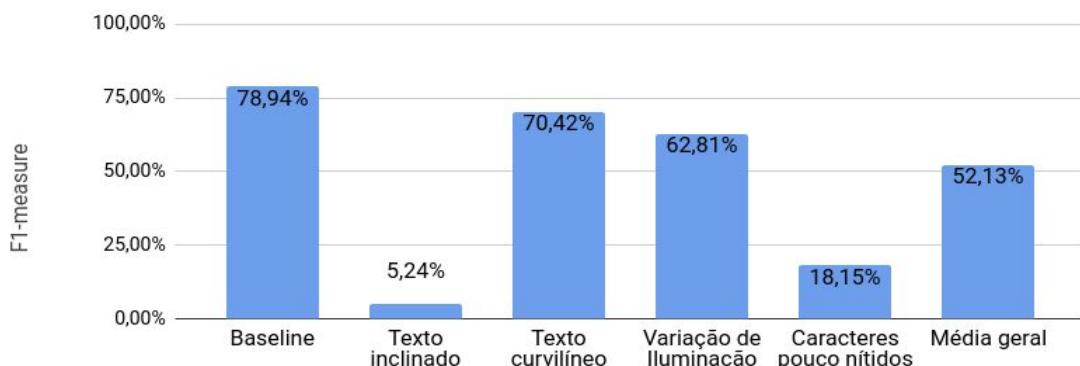


Figura 6: Eficácia da extração em relação às características de interesse

A Figura 6 apresenta os resultados obtidos, onde as colunas representam o percentual de F1. A primeira coluna denominada baseline refere-se às imagens que não possuem nenhuma das características de interesse. A segunda coluna refere-se às imagens com textos inclinados. A terceira coluna diz respeito às imagens que contém textos com aspectos curvilíneos. A quarta coluna apresenta as imagens que possuem variação de iluminação. Imagens que possuem caracteres pouco nítidos estão representados na quinta coluna. Por fim, a sexta coluna apresenta a média geral da eficácia obtida por todas as imagens da base de dados.

Observa-se que as imagens que não possuem nenhuma das características de interesse obtiveram uma média de F1 de 78,94%. As imagens que possuem texto inclinado obtiveram o menor percentual de eficácia (5,24%). Isso se deve ao fato do *Tesseract* não dispor de etapas de pré-processamento para corrigir adversidades relacionadas a inclinação das linhas de texto. Imagens com texto curvilíneo (70,42%) e imagens com variação de iluminação (62,81%), são as características que menos comprometeram a eficácia da extração. A média geral da eficácia da extração alcançou 52,13%, sugerindo que outras características não consideradas neste trabalho podem estar comprometendo a eficácia de extração.

6. Considerações finais

Este trabalho apresentou uma análise da relação entre a eficácia de extração de elementos textuais em imagens e suas características.

Após a realização de experimentos, constatou-se que a inclinação de textos (5,24%), bem como caracteres pouco nítidos (18,15%), são as principais características relacionadas aos baixos índices de eficácia na base de dados utilizada, ao instante que imagens com texto curvilíneo (70,42%) e variação de iluminação (62,81%) não comprometem substancialmente a eficácia de extração.

Como trabalhos futuros, destacam-se: (i) realizar testes em uma base de dados maior; (ii) analisar a influência de outras características; (iii) verificar a relação da eficácia da extração por intervalos de graus de inclinação.

Referências

- E. Manica; C. F. Dorneles; R. Galante. (2017). R-Extractor: a method for data extraction from template-based entity-pages. In *Computer Software and Applications Conference (COMPSAC), IEEE 41st Annual*. IEEE. p. 778-787.
- A. Labrinidis, H. V. Jagadish. (2012). Challenges and opportunities with big data, *Proceedings of VLDB Endowment*, v. 5, n.12, pp. 2032-2033.
- D. Agrawal, P. Bernstein, E. Bertino, et. al. (2012). *Challenges and Opportunities with Big Data - A community white paper developed by leading researchers across the United States*.
- Statistic Brain. (2017). Instagram Company Statistics. Disponível em: [brain https://www.statisticbrain.com/instagram-company-statistics](https://www.statisticbrain.com/instagram-company-statistics). Acessado em: 15 de Janeiro de 2018.
- D. Berchmans; S. S. Kumar. (2014). Optical character recognition: An overview and an insight. In *2014 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT)*, Kanyakumari, pp. 1361-1365.
- N. Islam; Z. Islam; N. Noor. (2016). A Survey on Optical Character Recognition System. *Journal of Information & Communication Technology-JICT* Vol. 10 Issue.2.
- F. Asad et al. (2016) High Performance OCR for Camera-Captured Blurred Documents with LSTM Networks. In *Document Analysis Systems (DAS)*, 2016 12th IAPR Workshop on. IEEE. p. 7-12.
- J. Liang, D. Doermann, and H. Li. (2005). Camera-based analysis of text and documents: a survey, *International Journal on Document Analysis and Recognition (IJDAR)*, v. 7, n. 2-3, pp. 84–104.
- C. Olah. (2015). Understanding LSTM. Disponível em: <http://colah.github.io/posts/2015-08-Understanding-LSTMs/>. Acesso em: novembro de 2017.
- R. W. Smith. (2017). The Extraction and Recognition of Text from Multimedia Document Images, PhD Thesis, University of Bristol, November 1987.
- Tesseract (2015). Tesseract. Disponível em: <https://github.com/tesseract-ocr/tesseract>. Acesso em: novembro de 2017.
- D. M. Kuhn; C. R. Cervi; E. Manica. (2017). Avaliação da eficácia da extração de texto em imagens. VI MOEPEX, IFRS, Campus Ibirubá. Disponível em: <https://eventos.ifrs.edu.br/index.php/MoEPEXIbiruba/6MOEPEX/paper/view/3332>.
- OpenCv. (2017). Hough Line Transform. Disponível em: https://docs.opencv.org/2.4/doc/tutorials/imgproc/imgtrans/hough_lines/hough_lines.html. Acessado em: dezembro de 2017.
- R. Baeza-Yates; B. Ribeiro Neto. (2013). Recuperação de Informação: Conceitos e Tecnologia das Máquinas de Busca. Porto Alegre: Bookman Editora.

Artigos Curtos de Aplicações/Experiências

AplicExp

Análise da situação dos redutores de velocidade de Curitiba	123
<i>Gabriely Simette (Universidade Tecnológica Federal do Paraná - Brasil), Yussef Parcianello (Universidade Tecnológica Federal do Paraná - Brasil), Nádia Kozievitch (Universidade Tecnológica Federal do Paraná - Brasil), Keiko Fonseca (Universidade Tecnológica Federal do Paraná - Brasil)</i>	
Research.NET Web: uma aplicação para análise e exibição de redes de colaboração acadêmica utilizando grafos dinâmicos	127
<i>Erick Lopes (Universidade Federal do Rio Grande - Brasil) e Eduardo Borges (Universidade Federal do Rio Grande - Brasil)</i>	
Publicação de Dados da Internet das Coisas usando Recursos da Web Semântica: Um estudo de Caso usando uma Tomada Inteligente	131
<i>Vinícius Tomazetti (Universidade Federal de Santa Maria - Brasil), Dióvane Soligo (Universidade Federal de Santa Maria - Brasil), Alencar Machado (Universidade Federal de Santa Maria - Brasil), Daniel Lichtnow (Universidade Federal de Santa Maria - Brasil)</i>	
DINO: uma ferramenta para importação de dados em bancos de dados NoSQL	135
<i>Angelo Frozza (Instituto Federal Catarinense - Campus Camboriú - Brasil), Geomar Schreiner (Universidade Federal de Santa Catarina - Brasil), Rian Brüggemann (Universidade Federal de Santa Catarina - Brasil), Ronaldo Mello (Universidade Federal de Santa Catarina - Brasil)</i>	

Análise da situação dos redutores de velocidade de Curitiba

Gabriely Simette¹, Yussef Parcianello¹, Nádia P. Kozievitch¹, Keiko V. O. Fonseca¹

¹Universidade Tecnológica Federal do Paraná - (UTFPR)
Avenida Sete de Setembro – 3165 – 80.230-901 – Curitiba – PR – Brasil

gabrielysimette@alunos.utfpr.edu.br, yussef.parcianello@ifsc.edu.br
{nadiap,keiko}@utfpr.edu.br

Abstract. Problems related to Urban Mobility on large urban centers cause countless damages not only to the citizen but also to the environment. To deal with this problem, the Speed Controller Device (SCD) is a fundamental resource used by Public Administration in urban and road planning. This paper presents an analysis of official georeferenced data from roads and speed control devices of Curitiba, aiming to verify if the SCD are installed in accordance to current legislation.

Resumo. Problemas relativos a Mobilidade Urbana nos grandes centros acabam por causarem inúmeros prejuízos não só ao cidadão como também ao meio ambiente. Para equalizar este problema, os Dispositivos Redutores de Velocidade (DRV) são recursos fundamentais, utilizados pela administração pública no planejamento urbano e viário. Este artigo apresenta uma análise de dados georreferenciados das vias públicas e dos dispositivos redutores de velocidade de Curitiba, com o intuito de verificar se os DRV estão de acordo com as legislações vigentes.

1. Introdução

A relação entre problemas de mobilidade urbana e o aumento do número de acidentes de trânsito já foi observado em várias pesquisas [French et al. 1993]. Uma das estratégias empregadas para equalizar tais problemas é o uso de dispositivos redutores de velocidades. Neste sentido, este artigo analisa se os dispositivos redutores de velocidade de Curitiba-PR estão instalados em locais adequados e conforme as legislações vigentes. Para tanto, foram analisadas as legislações que regulamentam os padrões e critérios para a utilização dos dispositivos reguladores de velocidade e confrontadas com dados georreferenciados que representam os eixos das ruas¹ da cidade de Curitiba e com os dados que tratam da localização dos dispositivos redutores de velocidade² daquela cidade (dados estes disponibilizados pelo sítio web do IPPUC³ e da SETRAN⁴, respectivamente).

A resolução 600/2016 do DENATRAN traz uma série de definições acerca da instalação de ondulações transversais. Tal resolução define que para um dispositivo disposto próximo a uma intersecção, por ex., exige-se uma distância mínima de 15 metros

¹Disponível em: http://ippuc.org.br/geodownloads/SHPES/EIXO_RUA.zip. Acesso em: Mar. 2018

²Disponível em: <http://setran.curitiba.pr.gov.br/servicos/fiscalizacao-eletronica>. Acesso em: Mar. 2018

³IPPUC – Instituto de Pesquisa e Planejamento Urbano de Curitiba.

⁴SETRAN - Secretaria Municipal de Trânsito de Curitiba.

entre a lombada física e o alinhamento do meio-fio da via transversal. No que diz respeito a faixa de pedestres elevada, a Resolução 495/2014 do CONTRAN traz uma série de definições acerca da sua instalação.

Este trabalho está organizado da seguinte forma: na Seção 2 são apresentados alguns trabalhos correlatos. A Seção 3 trata do desenvolvimento do trabalho. Por fim, conclui-se o trabalho na Seção 4.

2. Trabalhos correlatos

[Nakonetchnei et al. 2017] comparou os dados abertos do sistema de transporte público de Curitiba e de Nova Iorque, trazendo uma série de informações e de desafios relacionados ao tema. Já [Costa et al. 2017] abordou os desafios e explorou as oportunidades que a integração de dados abertos georreferenciados pode oferecer para o planejamento e gestão dos redutores de velocidade no transporte público de Curitiba. [Kozievitch et al. 2016a] propôs uma abordagem para desenvolver um planejador de rotas para usuários de cadeiras de rodas, baseando-se em dados referentes a eixos de ruas e enquadramento de Curitiba. Alguns avanços da referida pesquisa podem ser vistos em [Kozievitch et al. 2016b].

Apesar da relevância dos estudos relacionados e dos subsídios fornecidos por eles para a presente pesquisa, nenhum deles explorou os dados abertos de Curitiba no intuito de verificar se os dispositivos redutores de velocidades daquela cidade estão instalados em locais adequados conforme as legislações vigentes.

3. Desenvolvimento

Neste trabalho foram analisados dois conjuntos de dados: um contendo dados referentes a ruas e outro contendo os dados referentes aos radares e lombadas de Curitiba. Para a realização das análises e tratamento dos dados, foi utilizado o PostgreSQL 9.4⁵, o PostGIS 2.4⁶ e a ferramenta para visualização de dados georreferenciados QGis 2.18.10 64 bits⁷. A Tabela 1 traz uma breve caracterização de tais dados.

Tabela 1. Breve caracterização dos dados utilizados na pesquisa.

Descrição	Fonte	Qtd Tuplas	Formato	Georref.
Eixos das ruas	IPPUC	39.948	dbf, prj, sbn, sbx, shp e shx	Sim
Lombadas e radares	SETRAN	405	Tabela HTML	Não

Ao analisarmos os dados dos dispositivos redutores de velocidades de Curitiba, foi identificado que os dados estavam incompletos, alguns eram redundantes, e alguns não puderam ser georreferenciados (como ilustra a Tabela 2).

Para a identificação de possíveis dispositivos redutores de velocidades localizados a menos de 15 metros de cruzamentos, utilizou-se as funções ST_Dwithin() e ST_Closestpoint(). Assim, foram identificados 85 dispositivos localizados a menos de 15 metros de um cruzamento (conforme mostra a Figura 1A).

⁵Disponível em: <https://www.postgresql.org/download/>. Acesso em: Mar. 2018

⁶Disponível em: <http://postgis.net>. Acesso em: Mar. 2018

⁷Disponível em: <http://www.qgis.org>. Acesso em: Mar. 2018

Tabela 2. Breve caracterização dos dados utilizados na pesquisa.

Descrição	Quantidade
Redutores de velocidade redundantes	23
Redutores de velocidade sem latitude/ longitude	48

Cabe ressaltar que a granularidade dos dados referentes aos eixos das ruas pode ter comprometido a precisão dos resultados. O eixo de rua não representa fielmente uma via pública, mas sim uma abstração desta. A Figura 1B mostra a visualização dos eixos de ruas (linhas em vermelho) referentes ao cruzamento da rua Buenos Aires com a avenida Sete de Setembro. Percebe-se que o cruzamento, isto é, o ponto de intersecção daqueles eixos de rua (indicado em azul) está distante dos cantos do meio-fio daquele cruzamento (indicado em verde).

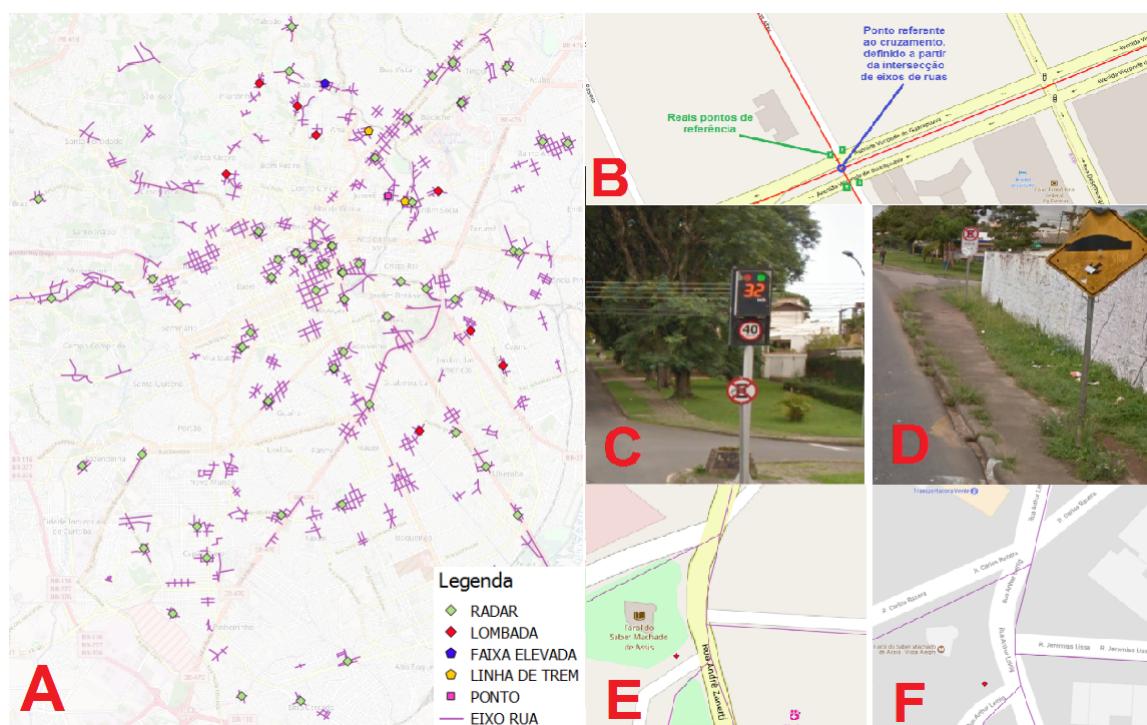


Figura 1. Problemas identificados durante a realização da pesquisa.

Também tornaram-se evidentes problemas relativos a categorização dos dispositivos redutores de velocidade. A informação referente ao tipo dos redutores de velocidade constantes na base de dados analisada muitas vezes não confere com as nomenclaturas definidas através das legislações vigentes (vide Art. 1º da Resolução 396/11 do CONTRAN⁸, Art. 1º da Resolução 495/14 do CONTRAN⁹ e Art. 3º da Resolução 600/16 do CONTRAN¹⁰). A Figura 1C e a Figura 1D mostram diferentes dispositivos cadastrados

⁸Disponível em: http://www.denatran.gov.br/download/Resolucoes/RESOLUCAO_CONTRAN_396_11.pdf. Acesso em: Mar. 2018

⁹Disponível em: <http://www.denatran.gov.br/download/Resolucoes/Resolucao4952014.pdf>. Acesso em: Mar. 2018

¹⁰Disponível em: http://www.denatran.gov.br/images/Resolucoes/Resolucao6002016_new.pdf. Acesso em: Mar. 2018

como "LOMBADA": um "controlador eletrônico de velocidade" localizado no cruzamento das ruas Dr João Evangelista Espíndola e Raphael Papa e uma "elevação transversal" localizada próxima do cruzamento da rua Amauri Lange Silverio com a José Moraes, respectivamente.

Além disso, outro problema que pode ter comprometido os resultados deste trabalho é o fato de os eixos das ruas serem diferentes de ferramentas online. Tal fato é evidenciado através da Figura 1E e da Figura 1F: o traçado do eixo das ruas (linhas na cor lilás), embora possam ser parecidos, não conferem com o traçado das vias públicas, nem do Open Street Maps, tampouco do Google Maps, respectivamente.

Além disso, faltava informação acerca de qual elemento do dispositivo foi georreferenciado (o sensor ou o display, no caso dos dispositivos eletrônicos) e de qual ponto foi utilizado como base para obtenção das coordenadas geográficas (antes, encima ou depois do dispositivo, a exemplo da elevação transversal).

4. Conclusões

Este trabalho apresentou uma breve análise, confrontando dados de DRV e a legislação vigente. Concluiu-se que há indícios de que existam dispositivos redutores de velocidade instalados a menos de 15 metros de um cruzamento e, portanto, em situação irregular. Porém, em função das limitações dos dados, não pôde ser possível precisar quantos e nem quais. Deste modo, as limitações identificadas são questões de devem ser equalizadas no sentido de permitir que estudos mais aprofundados possam ser realizados. Como trabalhos futuros, sugere-se a realização deste mesmo estudo, mas (i) utilizando mais dados; (ii) utilizando uma menor granularidade de dados para realizar os cálculos; (iii) incluindo detalhes mais específicos, como altura de lombada, fonte da localização GIS, entre outros. **Agradecimentos.** Os autores agradecem a Prefeitura Municipal de Curitiba, IPPUC e ao projeto EU-BR EUBra-BigSea (*MCTI/RNP 3rd Coordinated Call*).

Referências

- [Costa et al. 2017] Costa, G. and Kozievitch, N. P., Fonseca, K., Gadda, T., and Berardi, R. (2017). Integração de dados de redutores de velocidade no transporte público de curitiba. *Escola Regional de Banco de Dados*, pages 123–126.
- [French et al. 1993] French, D. J., West, R. J., Elander, J., and Wilding, J. M. (1993). Decision-making style, driving style, and self-reported involvement in road traffic accidents. *Ergonomics*, 36(6):627–664.
- [Kozievitch et al. 2016a] Kozievitch, N. P., Almeida, L. D. A., Silva, R. D., and Minetto, R. (2016a). An alternative and smarter route planner for wheelchair users: Exploring open data. *International Conference on Smart Cities and Green ICT Systems*, pages 1–6.
- [Kozievitch et al. 2016b] Kozievitch, N. P., Minetto, R., Silva, R. D., Dell, L., Almeida, A., and Santi, J. (2016b). Shortcut suggestion based on collaborative user feedback for suitable wheelchair route planning. *International Conference on Intelligent Transportation Systems*, (19):2372–2377.
- [Nakonetchnei et al. 2017] Nakonetchnei, E. C., Kozievitch, N. P., Cappiello, C., Vitali, M., and Akbar, M. (2017). Mobility open data: Use case for curitiba and new york. *Escola Regional de Banco de Dados*, pages 111–114.

aper:180168_1

Research.NET Web: uma aplicação para análise e exibição de redes de colaboração acadêmica utilizando grafos dinâmicos

Érick Luiz Fonsêca Lopes¹, Eduardo N. Borges¹

¹Centro de Ciências Computacionais – Universidade Federal do Rio Grande (FURG)
Av. Italia, km 8, 96203-900. Rio Grande – RS

si.erickluiz@gmail.com, eduardoborges@furg.br

Resumo. Este artigo descreve o desenvolvimento de uma solução Web que permite a visualização dinâmica de redes de colaboração acadêmica. A aplicação utiliza dados da plataforma Lattes para identificar e exibir a relação entre conjuntos de pesquisadores.

Abstract. This paper describes the development of a Web solution that allows the dynamic visualization of academic collaboration networks. The application uses data from the Lattes platform to identify and display the relationship between sets of researchers.

1. Introdução

Atualmente não é incomum a necessidade de avaliar redes formadas a partir da colaboração entre autores de determinado grupo, tanto no meio acadêmico quanto fora dele. Para [MENA-CHALCO and CESAR-Jr 2009], estas avaliações podem ser utilizadas, por exemplo, para documentar a produção científica do grupo. Extrair valores reais de uma rede de colaboradores e exibi-los de maneira simples não é tarefa fácil em tempos em que a informação é produzida de forma constante e abrangente. Isto implica em se utilizar uma boa fonte de dados para a absorção do conhecimento requerido.

Referente à pesquisa, o Brasil possui uma particularidade excepcional: “a existência de um cadastro nacional de currículos de pesquisadores, a Plataforma Lattes”, onde se concentram informações referentes à vida no meio científico/acadêmico dos profissionais cadastrados [DIGIAMPIETRI 2015]. Sistemas que extraem informações da plataforma Lattes têm como uma das principais funcionalidades a análise biométrica de possíveis redes de colaboração geradas pelo conhecimento obtido [FARIAS and BORGES 2013] e [CORREA et al. 2017].

A bibliometria é o estudo da produção científica em determinado contexto. Dentre as preocupações que a análise biométrica possui, a mais significativa dentro do escopo deste artigo é a sua utilização para controle bibliográfico que permite conhecer o tamanho e a característica do acervo: número real de produções, por exemplo [ARAÚJO 2006].

Deste modo, Research.NET Web é uma solução computacional que auxilia na observação de relacionamentos de coautoria construídos a partir dos currículos dos pesquisadores envolvidos no estudo, sendo estes obtidos através da plataforma Lattes¹. Sua funcionalidade principal é identificar o número real de publicações da rede, percebendo as coautorias e exibindo de forma gráfica os relacionamentos obtidos. A principal

¹<http://lattes.cnpq.br/>

contribuição deste trabalho é proporcionar uma visualização dinâmica da rede sem a necessidade de reprocessamento para avaliar períodos distintos.

2. Trabalhos Relacionados

Ainda que existam diversas aplicações com o propósito de analisar redes formadas por colaboração em pesquisa como ArnetMiner² e CiênciaBrasil³, este trabalho tomou como base o software Research.Net, desenvolvido na Universidade Federal do Rio Grande.

O Research.Net é “[...] um sistema de informação que facilita a análise das redes de colaboração acadêmica entre membros de uma ou mais IES” [FARIAS and BORGES 2013]. Desenvolvido na linguagem de programação Java, O sistema se apresenta em uma versão *desktop* para execução local [FARIAS and BORGES 2013].

Para gerar suas visualizações a ferramenta utiliza o software scriptLattes, que extrai informações dos currículos dos pesquisadores da rede advindos da Plataforma Lattes [MENA-CHALCO and CESAR-Jr 2009]. A utilização da plataforma Lattes se dá pela dificuldade encontrada em avaliar as produções de uma Instituição de Ensino Superior (IES), tendo em vista o alto número de publicações e que ferramentas de indexação de trabalhos acadêmicos podem não abranger as conferências e revistas científicas que contêm estas publicações [FARIAS et al. 2012].

O scriptLattes é um *script* utilizado para compilar listas de produções e tratar as duplicatas⁴ identificadas. Ademais, possibilita criação de grafos de coautoria e mapa de geolocalização dos membros inseridos na análise. Possui a licença GNU-GPL e opera de forma semiautomática devido a validação de acesso - CAPTCHA - adotada pela plataforma Lattes, o que torna sua utilização mais complexa [MENA-CHALCO and CESAR-Jr 2009].

3. Research.NET Web

O Research.NET Web é um sistema de informação utilizado para facilitar a análise de redes de colaboração acadêmicas construídas a partir do conhecimento extraído do currículo Lattes dos pesquisadores da rede. Tendo como base o Research.Net, a aplicação foi implementada do zero e esta disponível⁵ na plataforma GitHub. Sua capacidade de verificar o número real de produções de um grupo e como os pesquisadores se relacionam por meio de seus trabalhos em eventos o tornam um sistema de análise bibliométrica. O funcionamento da aplicação ocorre por meio de três módulos: o recebimento de currículos, o construtor de redes e o visualizador de redes.

O módulo de recebimento de currículos recebe arquivos no formato XML, verifica sua existência e, caso o currículo ainda não tenha sido cadastrado, salva o currículo na aplicação. A construção da rede requer que os currículos estejam cadastrados, sendo assim, uma rede só pode ser construída após a submissão dos arquivos a este módulo.

O construtor de redes recebe os pesquisadores e constrói uma nova rede através dos arquivos do diretório de currículos. Após identificar as duplicatas e quantificar o

²<http://www.arnetminer.org/index.jsp>

³<http://www.cienciabrasil.org.br>

⁴Referências bibliográficas semanticamente equivalentes

⁵<https://github.com/researchnetWeb/Research.NET-WEB>

acervo da rede a aplicação salva um arquivo XML com a estrutura gráfica que a representa. Este deverá ser executado antes do visualizador de redes.

Por fim, a aplicação conta com o módulo visualizador de redes que recebe uma rede a ser lida e requisita ao servidor o arquivo da rede (previamente construído). Após a leitura do arquivo o módulo exibe a rede conforme demonstrado na Figura 1.

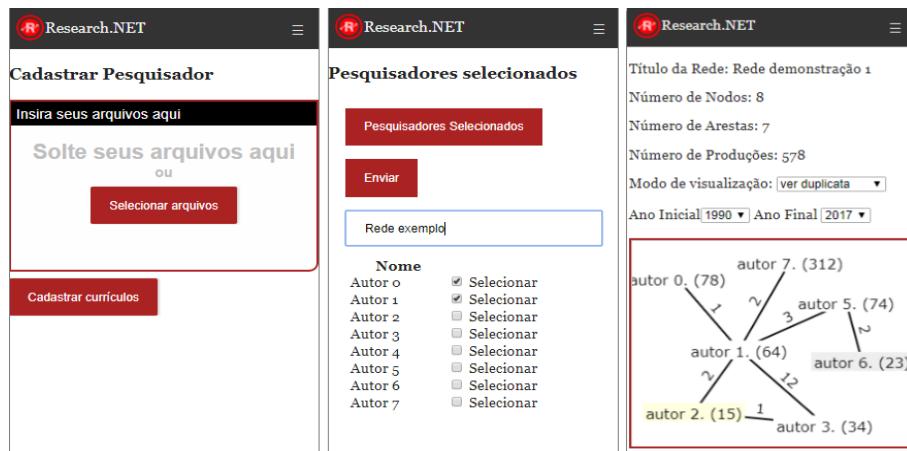


Figura 1. Interfaces gráficas das principais funcionalidades em versão mobile.
A identidade dos autores foi preservada neste exemplo. Junto do nodo e aresta é possível identificar o número de produções do pesquisador e coautorias, respectivamente.

Seguindo o fluxo de funcionamento da aplicação de cadastrar os currículos, criar a rede e acessar a página de visualização, o usuário tem acesso a algumas informações sobre pesquisadores e redes cadastradas que auxiliam na análise, tais como: número de nodos ou vértices, número de arestas, número real de produções da rede (eliminando duplicatas), período de análise, relações e as produções que as constituem.

A Figura 2 apresenta uma rede exemplificando as propriedades.

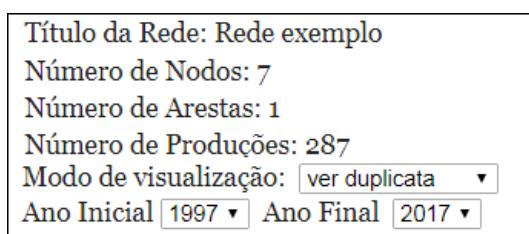


Figura 2. Informações apresentadas pelo módulo visualizador.

Ao manipular o grafo, o usuário ainda pode visualizar informações dos nodos, como nome completo do pesquisador, número de produções e grau para o período selecionado, e *link* do currículo Lattes. Além disso, consultando informações de cada aresta, são apresentadas as produções que lhe deram origem, ou seja, extraídas dos currículos dos pesquisadores e identificadas como duplicatas pelo Research.NET Web através de seu algoritmo de deduplicação que considera a distância levenshtein⁶ para comparar os títulos

⁶A distância levenshtein retorna o número de edições necessárias pra transformar um conjunto de caracteres em outro

das produções.

4. Conclusões

Este trabalho trouxe como contribuição uma solução computacional que cria redes a partir de currículos extraídos da plataforma Lattes, calcula suas métricas e exibe de forma dinâmica os grafos de saída. A análise da rede é feita para cada ano em que existam produções, isto permite visualizar a rede em períodos diferentes sem a necessidade de um novo processamento, apenas modificando o ano inicial ou final no descritor da mesma. Ademais, garante que o currículo de um pesquisador não será cadastrado duas vezes sem que o mesmo esteja com uma data de atualização mais recente.

Ainda limitada, a solução necessita de aprimoramento na inserção de métricas para um aproveitamento melhor do conhecimento obtido. Sobre a leitura de currículos, na versão atual a aplicação se limita em trabalhos em eventos e deve ser ampliada para utilizar também outros tipos de produções como artigos em periódicos, livros e capítulos de livros.

Por fim, entender uma rede e seu comportamento é um processo massivo que pode ser demorado quando não são utilizadas aplicações que o facilitem. Neste ponto, o trabalho soma ao ajudar na visualização, não apenas da rede, mas dos dados extraídos dela. Considera-se que uma nova fase do projeto irá proporcionar uma ampliação considerável no público alvo e no alcance da aplicação. Disponibilizá-lo online como um serviço da Universidade Federal do Rio Grande (FURG) pode auxiliar pesquisadores na construção e análise de redes de colaboração, principalmente em IES.

Referências

- ARAÚJO, C. A. (2006). Bibliometria: evolução histórica e questões atuais. *Em Questão*, 12(1):11–32.
- CORREA, T. S., SUZUKI, M. B., CINTRA, P. R., and COSTA, L. S. F. O. (2017). O fim do scriptlattes? uma análise de suas funcionalidades, alternativas para o presente e perspectivas para o futuro. *Revista do EDICC*, 3(3):138–148.
- DIGIAMPIETRI, L. A. (2015). Análise da rede social acadêmica brasileira. Livre docência, Escola de Artes, Ciências e Humanidades, Universidade de São Paulo, São Paulo.
- FARIAS, L. R. and BORGES, E. M. (2013). Research.net: um sistema para análise de redes de colaboração baseado na plataforma lattes. Trabalho de conclusão de curso – monografia de graduação em engenharia de computação, Universidade Federal do Rio Grande, Rio Grande.
- FARIAS, L. R., VARGAS, A. P., and BORGES, E. N. (2012). Um sistema para análise de redes de pesquisa baseado na plataforma lattes. In *Escola Regional de Banco de Dados*. Sociedade Brasileira de Computação.
- MENA-CHALCO, J. P. and CESAR-Jr, R. M. (2009). scriptlattes: An open-source knowledge extraction system from the lattes platform. *Journal of the Brazilian Computer Society*, 15(4):31–39.

Publicação de Dados da Internet das Coisas usando Recursos da Web Semântica: Um estudo de Caso usando uma Tomada Inteligente

Vinícius C. Tomazetti¹, Diovane Soligo¹, Alencar Machado¹, Daniel Lichtenow¹

¹Colégio Politécnico – Universidade Federal de Santa Maria (UFSM)
Av. Roraima, nº1000, Campus UFSM – 97.105-900 – Santa Maria – RS – Brasil

vinicios.c.tomazetti@gmail.com, diovane_soligo@hotmail.com,
alencar.machado@ufsm.br, dlichtnow@politecnico.ufsm.br

Abstract. One of the challenges in the Internet of Things – IoT is how to publish and share the data related to devices. One almost unexplored possibility is to use Semantic Web resources. Thus, this work aims to explore the use of some of Semantic Web resources to publish and share data related to a smart socket.

Resumo. Um dos desafios na Internet das Coisas está em como publicar e compartilhar dados relacionados aos dispositivos. Uma possibilidade, ainda pouco explorada, é fazer uso de recursos da Web Semântica. Assim, este trabalho tem por objetivo explorar o uso de recursos da Web Semântica para publicar e compartilhar dados relacionados a uma tomada inteligente.

1. Introdução

Na Internet das Coisas - *Internet of Things* - IoT objetos físicos/dispositivos estão conectados a Internet, podendo ser monitorados e controlados remotamente. Um exemplo de dispositivo é uma tomada inteligente, que consiste basicamente de uma tomada que possui sensores (e.g. corrente elétrica) a partir dos quais são obtidos dados que podem ser utilizados para monitoramento e realização de ações a partir do uso de atuadores presentes em um dispositivo (e.g. interromper a corrente que passa na tomada, desligando consequentemente o aparelho ligado a tomada). No cenário proposto na Internet das Coisas, um desafio consiste em garantir a interoperabilidade entre os objetos e o acesso aos dados a eles associados (dados que descrevem os dispositivos e dados obtidos pelos seus sensores). Para tanto, as tecnologias relacionadas à Web Semântica podem auxiliar [Charpenay et al, 2016].

Embora sejam reconhecidos os benefícios do uso dos recursos da Web Semântica na IoT, estes não vêm sendo ainda largamente utilizados [Sotres et al, 2017]. Assim, este trabalho tem por objetivo explorar o uso dos recursos da Web Semântica no contexto da IoT, usando como estudo de caso uma tomada inteligente. Inicialmente são descritas algumas das ontologias relevantes para IoT que foram identificados ao longo do trabalho (Seção 2). É então feito o relato de como destas ontologias foram usadas na descrição dos dados relacionados à tomada inteligente (Seção 3). Finalmente, são apresentadas as considerações finais (Seção 4).

2. Ontologias no contexto da IoT

Para a descrição dos dispositivos, no contexto da IoT, existem diversas ontologias definidas usando os recursos da Web Semântica. Dentre as mais conhecidas estão a *SSN* (*Semantic Sensor Network Ontology*), a *IoT-Lite* e a *FIESTA-IoT*. A ontologia *SSN* permite descrever sensores [Compton et al., 2012] e consiste de 41 conceitos e 39 propriedades de objetos, sendo que houve na sua criação a preocupação com a integração com ontologias externas, que definem conceitos não diretamente relacionados a sensores (unidades de medida, por exemplo). A ontologia *SSN* é considerada a mais conhecida no contexto da IoT, mas também vista como complexa [Lanza et al, 2016]. A complexidade da *SSN* foi uma das motivações para a definição da *IoT-Lite* [Bermudez-Edo et al, 2016]. A *IoT-Lite* consiste de uma customização da *SSN*, buscando oferecer uma ontologia mais leve. Conceitos centrais da *IoT-Lite* são entidades ou objetos; recursos ou dispositivos e serviços, que são nomeados na *IoT-Lite* respectivamente como *iot-lite:Object*, *ssn:Device* e *iot-lite:Service*. A *IoT-LITE*, disponibiliza elementos capazes de caracterizar dispositivos bem como seus sensores, *tags* acopladas (etiquetas *RFID/NFC*) e seus atuadores, descrevendo o tipo de informação que um dispositivo manipula/gera, sua localização geográfica, serviço de *endpoint* disponível, dentre outras características.

Já a *FIESTA-IoT* [Agarwal et al 2016] também reutiliza conceitos de outras ontologias, especialmente da *SSN* e *IoT-Lite*. Além disto, possui elementos para descrição de dados e da sua coleta por sensores que não estão presentes na *SSN* e na *IoT-Lite* (e.g. no caso do sensor de temperatura, a unidade de medida, o valor coletado, data que o valor foi coletado, sensor que coletou, dentre outras). Suporta ainda os formatos mais comuns de descrição semântica (e.g. *JSON-LD*, *N3*, *RDF/XML*, *OWL/XML*, *TURTLE*).

3. Estudo de Caso

Uma tomada inteligente foi o dispositivo escolhido neste trabalho para ser representado com o uso das ontologias estudadas. Esta tomada possui diversos sensores: corrente elétrica, tensão elétrica, temperatura e umidade do cômodo em que foi instalada. Além de realizar a coleta de dados através de sensores, a tomada possui um atuador, um relé de carga capaz de acionar ou desativar o dispositivo que estiver conectado a esta tomada inteligente através de requisições *HTTP* enviadas por uma aplicação. Os dados obtidos pelos sensores eram originalmente armazenados em um banco de dados relacional. O desenvolvimento da tomada inteligente usada é detalhado em [Soligo e Machado, 2017].

Foi utilizada, a ontologia *IoT-Lite* para representar a tomada inteligente e seus componentes (sensores e atuadores). Estes dados foram publicados na *Web*, para testes, utilizando o servidor local *Apache Jena Fuseki*, que armazena as triplas *RDF* da ontologia e disponibiliza uma *URI* a partir do qual sistemas externos podem ter acesso a mesma, bem como realizar consultas *SPARQL* (Figura 1). São observados os princípios de dados abertos [Bizer et al, .2008].

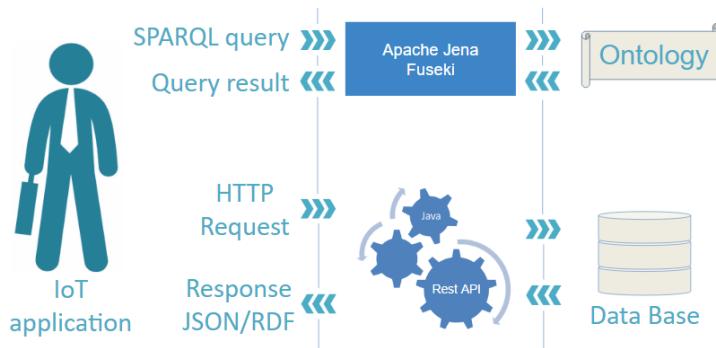


Figura 1. Interação entre aplicação, ontologia e dados do dispositivo

Já usando elementos da ontologia *FIESTA-IoT*, especialmente aqueles referentes a parte de descrição de observações, foi gerado um arquivo *RDF* dos dados de obtidos pelos sensores da tomada (voltagem, por exemplo). Estes dados estavam originalmente em uma base de dados relacional, usada para armazenar os dados da coleta (361.066 registros). O framework Java Apache Jena foi utilizado para criar os modelos *RDF* com as informações coletadas do banco. Novamente, os dados foram disponibilizados localmente, em caráter experimental, no servidor *Apache Jena Fuseki*, onde são passíveis de consultas *SPARQL* por sistemas externos. A Figura 2 mostra a consulta em *SPARQL* para recuperar os dados sobre a variação da voltagem na tomada.

```
prefix tomada:<http://purl.oclc.org/NET/UNIS/fiware/iot-lite#>
prefix geo: <http://www.w3.org/2003/01/geo/wgs84_pos#>
prefix iot-lite: <http://purl.oclc.org/NET/UNIS/fiware/iot-lite#>
prefix qu: <http://purl.org/NET/ssnx/qu/qu#>
prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
prefix owl: <http://www.w3.org/2002/07/owl#>
prefix qu-rec20: <http://purl.org/NET/ssnx/qu/qu-rec20#>
prefix xsd: <http://www.w3.org/2001/XMLSchema#>
prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
prefix time: <http://www.w3.org/2006/time#>
prefix j.0: <http://www.loa.istc.cnr.it/ontologies/DUL.owl#>
prefix m3-lite: <http://purl.org/iot/vocab/m3-lite#>
prefix dc: <http://purl.org/dc/elements/1.1/>
prefix ssn: <http://purl.oclc.org/NET/ssnx/ssn#>
SELECT ?id ?voltagem WHERE {
    tomada:smart-outlet-001 ?p ?o;
    tomada:id ?id;
    ssn:hasSubSystem ?o2.
    ?o2 ssn:madeObservation tomada:voltage-service;
    ?o4 ?o5.
    ?o5 ssn:observationResult ?o6.
    ?o6 ssn:hasValue ?o7.
    ?o7 j.0:hasDataValue ?voltagem .
}
```

Figura 2. Exemplo de dados e consulta SPARQL

É possível constatar os benefícios dos dados serem publicados usando recursos da Web Semântica, considerando cenários relacionados a Ambientes Inteligentes. Um exemplo é o de um ambiente onde se deseja manter o conforto ambiental, levando em

conta o consumo de energia. Neste cenário, uma aplicação, em intervalos regulares de tempo, executaria consultas *SPARQL* para recuperar a temperatura do ambiente e o consumo (obtido a partir da tensão e corrente) para então realizar ações automatizadas (e.g. regular a temperatura de um ar-condicionado). Aqui, o uso de padrões da Web Semântica para todos os dispositivos envolvidos favoreceria a obtenção dos dados e a identificação dos atuadores.

4. Considerações Finais

Dentro do contexto da IoT, neste trabalho foi descrito o uso de tecnologias de Web Semântica para descrição de uma tomada inteligente. Em vários trabalhos foi verificado que as tecnologias da Web Semântica poderão vir a fornecer um importante apoio, mas poucos trabalhos vem as utilizando, em parte dado ao fato de que muitos padrões ainda estão sendo definidos. Pretende-se, na sequência do trabalho, buscar definir e construir aplicações que demonstrem a utilidade destes recursos na troca de dados entre dispositivos e na criação de aplicações voltadas para Ambientes Inteligentes.

Agradecimentos

Trabalho apoiado pelo programa de bolsas de ensino, de pesquisa e de extensão do Colégio Politécnico da UFSM.

Referências

- Agarwal, R., Fernandez, D. G., Elsaleh, T., Gyrard, A., Lanza, J., Sanchez, L., ... & Issarny, V. (2016). Unified IoT ontology to enable interoperability and federation of testbeds. In Internet of Things (WF-IoT), 2016 IEEE 3rd World Forum (pp. 70-75).
- Bermudez-Edo, M., Elsaleh, T., Barnaghi, P., & Taylor, K. (2016). IoT-Lite: a lightweight semantic model for the Internet of Things. In Ubiquitous Intelligence & Computing, Advanced and Trusted Computing, 2016 IEEE Conferences (pp. 90-97).
- Bizer, C., Heath, T., Idehen, K., & Berners-Lee, T. (2008). Linked data on the web (LDOW2008). In Proc. of the 17th international conference on (pp. 1265-1266).
- Charpenay, V., Käbisch, S., & Kosch, H. (2016). Introducing Thing Descriptions and Interactions: An Ontology for the Web of Things. In SR+SWIT@ISWC (pp. 55-66).
- Compton, M., Barnaghi, P., Bermudez, L., GarcíA-Castro, R., Corcho, O., Cox, S., (2012). The SSN ontology of the W3C semantic sensor network incubator group. Web semantics: science, services and agents on the World Wide Web, 17, 25-32.
- Lanza, J., Sanchez, L., Gomez, D., Elsaleh, T., Steinke, R., & Cirillo, F. (2016). A proof-of-concept for semantically interoperable federation of IoT experimentation facilities. Sensors, 16(7), 1006.
- Soligo, D.; Machado, A. (2017). Internet das Coisas Aplicada à Eficiência Energética. Trabalho de conclusão do Curso de Sistemas para Internet – UFSM.
- Sotres, P., Santana, J. R., Sánchez, L., Lanza, J., & Muñoz, L. (2017). Practical lessons from the deployment and management of a smart city Internet-of-Things infrastructure: The smartsantander testbed case. IEEE Access, 5, 14309-14322.

DINo: uma ferramenta para importação de dados em bancos de dados NoSQL

**Angelo Augusto Frozza^{1,2}, Geomar Schreiner¹,
Rian Brüggemann¹, Ronaldo dos Santos Mello¹**

¹Universidade Federal de Santa Catarina (UFSC)
Campus Universitário Trindade - CP 476 - CEP88040-900 - Florianópolis (SC), Brasil

²IFC - Instituto Federal Catarinense - Campus Camboriú
Rua Joaquim Garcia, S/N - CP 2016 - CEP 88340-055 - Camboriú (SC), Brasil

angelo.frozza@ifc.edu.br, {geomarschreiner, riancvb}@gmail.com, r.mello@ufsc.br

Abstract. This paper presents *DINo*, a tool to help import relational data to NoSQL databases. Its main characteristics are: be multiplatform; support various types of DBMS (both relational and NoSQL); be flexible, allowing the user to make changes to the data mapping, publicly available. Tests were performed with data from an OpenStreetMap database.

Resumo. Este artigo apresenta o *DINo*, uma ferramenta para importação de dados relacionais para bancos de dados NoSQL. Suas principais características são: multiplataforma; suportar diversos tipos de SGBDs (relacionais e NoSQL); permitir alteração no mapeamento dos dados; e, estar disponível publicamente. São apresentados testes realizados com dados do OpenStreetMap.

1. Introdução

A explosão no uso de *Big Data* fez com que grandes empresas demandassem por bancos de dados (BDs) capazes de gerenciar grandes volumes de dados de forma eficaz e com um alto desempenho. Nesse contexto, os tradicionais BDs relacionais apresentam algumas limitações quanto a alta concorrência em operações de leitura e escrita, armazenamento de *Big Data* de forma eficiente, suporte a escalabilidade horizontal, além da garantia de um serviço rápido e ininterrupto (alta disponibilidade) [Han et al. 2011].

Levando em consideração essas necessidades, uma variedade de novos sistemas de BDs surgiu, focados principalmente no baixo custo de operação e manutenção. Esses BDs são popularmente conhecidos por NoSQL, que é um acrônimo para "Not Only SQL". Os BDs NoSQL geralmente são classificados de acordo com o modelo de dados adotado: *chave-valor*, *documentos*, *colunar* e *orientado a grafos* [Sadalage and Fowler 2012] [Cattell 2011, Sadalage and Fowler 2012, Ruiz et al. 2015], sendo os três primeiros também chamados de modelos baseados em chave.

Como toda tecnologia nova, o surgimento dos BDs NoSQL também trouxe a necessidade de desenvolvimento de novas ferramentas que permitam melhorar a produtividade de quem usa esse tipo de BD [Ruiz et al. 2015], com destaque neste artigo para ferramentas que auxiliam na importação de dados relacionais para BDs NoSQL.

O objetivo deste artigo é apresentar o *DINo* - uma ferramenta para importação de dados relacionais para BDs NoSQL, que visa auxiliar desenvolvedores de aplicações, em

especial aqueles que tem pouco domínio sobre NoSQL, facilitando a migração de dados. O artigo está organizado da seguinte forma: na seção 2 são discutidos alguns trabalhos relacionados; a seção 3 apresenta a ferramenta e suas estratégias de mapeamento de BD relacional para NoSQL; e, a seção 4 apresenta as considerações finais e trabalhos futuros.

2. Trabalhos relacionados

Em geral, a tarefa de migração pode ser realizada de diversas maneiras: (i) através de *scripts* para migração de um BD antigo para um novo BD [Mojeprojekty 2017]; (ii) pelo uso de ferramentas de migração para casos específicos [Murari et al. 2016]; ou (iii) pelo uso de ferramentas com suporte a diversos modelos de BDs [Vale and Rocha 2011].

A ferramenta proposta por [Murari et al. 2016] permite a migração de um BD *Firebird* para o BD NoSQL *MongoDB* e foi desenvolvida para realizar a migração de dados de uma aplicação específica. [Vale and Rocha 2011] propõem uma ferramenta que suporta diversos SGBDs (Sistemas Gerenciadores de Banco de Dados) relacionais como entrada, bem como, diversas abordagens NoSQL como saída. No entanto, até o momento, só foi implementado o suporte para os BDs *MySQL* e *MongoDB*, respectivamente.

[Santos Neto et al. 2013] propõem um conjunto de requisitos para uma ferramenta de migração, destacando: envolver estruturas diferentes (na entrada e na saída); executar de forma paralela ou distribuída; permitir migração incremental; permitir reengenharia de dados; suportar paradigmas de BDs diferentes; migrar BDs com modelagens diferentes; testabilidade; e, boa usabilidade. O DINO busca atender a maior parte dos requisitos.

No processo de migrar uma base relacional para um BD NoSQL é importante analisar como pode ser feito o mapeamento entre os modelos. Algumas propostas são apresentadas por [Zhao et al. 2014, Schreiner et al. 2015, Claudino et al. 2015, Poffo 2016].

3. A ferramenta DINO

O *DINO - Data Insertion in NoSQL* tem por objetivo auxiliar na migração de dados de um BD relacional para um BD NoSQL. Suas principais características são: suporte a diferentes SGBDs relacionais; suporte a diferentes modelos de dados NoSQL; processamento paralelo, através do uso de *threads*; interface simples e interativa.

A interface é dividida em três partes: (1) *Source*; (2) *Target*; (3) *Execute* (Figura 1). Em *Source*, são inseridas as informações para conexão com o SGBD de origem. Atualmente, o DINO suporta o *PostgreSQL*, mas pode ser atualizado para fornecer suporte a outros SGBDs relacionais através da alteração nos componentes das conexões JDBC. Não são necessários outros mapeamentos, uma vez que estes são realizados no *script SQL* de leitura de dados. Após conectar com o SGBD, o usuário deve selecionar um *database* e, em seguida, uma tabela desse *database*. Uma lista de colunas da tabela é apresenta ao usuário (Figura 2). Na sequência, é escolhido o banco NoSQL de destino e estabelecida a conexão com esse banco (Figura 2). Por fim, o usuário define como deve ser feito o mapeamento dos dados do BD *source* para o BD *target*, ou seja, define a composição da *chave* e do *valor* (Figura 3). Após definidos os parâmetros de conexão e as informações a serem importadas, os próximos passos são gerar o *script* de importação através do botão “*Generate sql*” e selecionar o botão “*Import*” para que o processo tenha início. O mapeamento automatizado de tabelas relacionais para modelos NoSQL é um tópico complexo, que é

altamente sensível ao domínio. Desta forma, este trabalho realiza a exportação de tabelas individualmente para que o usuário tenha total controle dos mapeamentos realizados. Além disso, o campo de edição da SQL flexibiliza o mapeamento na ferramenta, permitindo, por exemplo, a criação de junções com mais de uma tabela no banco relacional ou modificar o formato de algum dado. Diferente dos bancos de dados relacionais, os BDs NoSQL não implementam o conceito de relacionamento. Assim, o DINo não realiza o mapeamento das chaves estrangeiras (ou relacionamentos), os valores são tratados como colunas normais. Quando é feito o mapeamento de um BDR para um NoSQL, os relacionamentos devem ser resolvidos no momento do mapeamento. Caso deseje, o usuário pode realizar o mapeamento utilizando junções alterando o *script* SQL gerado pelo DINo antes de iniciar a importação dos dados.

O processo de migração é muito oneroso, implicando que cada registro do BD relacional seja consultado e inserido no BD NoSQL. Desta forma, a fim de melhorar o desempenho na migração dos dados, utiliza-se paralelismo para ler e inserir os dados. Durante a importação dos dados, o DINo busca quantos núcleos do processador estão ociosos e para cada núcleo ocioso é criada uma *thread* de importação. O total de dados a ser importado é dividido pelo número de *threads* (cada *thread* é representada por uma página dos dados relacionais). As threads não tomam conhecimento uma da outra e não possuem mecanismos para importar dados de chaves estrangeiras ou outras restrições, cada uma apenas recebe um conjunto de dados e os insere sequencialmente no BD NoSQL.

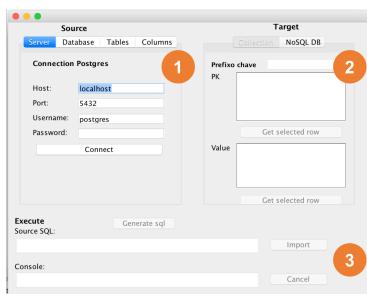


Figura 1. Interface do DINo

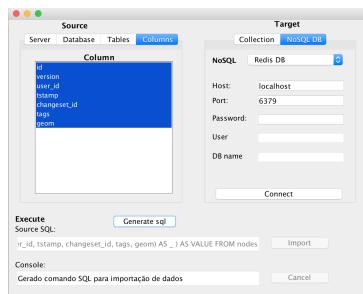


Figura 2. Conexão com o NoSQL

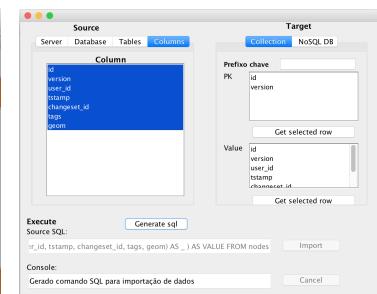


Figura 3. Mapeamento chave-valor

O DINo mapeia dados para os paradigmas NoSQL baseados em chave da seguinte forma:

NoSQL chave-valor : Consiste em definir (Figura 3): *(i)* a chave - que pode ser composta por um prefixo e um ou mais campos da tabela relacional; *(ii)* o valor - que é um documento JSON formado pelos campos selecionados na tabela relacional;

NoSQL documento : Os dados de cada tupla são agrupados em um documento JSON, que é armazenado em uma *collection* que possui o mesmo nome da tabela.

NoSQL colunar : Segue a regra: *(i)* o nome da tabela relacional é usado para criar uma *família de colunas*; *(ii)* cada coluna da tabela relacional corresponde a uma coluna no banco NoSQL; *(iii)* define-se qual é a chave da família de colunas.

Alguns testes preliminares foram realizados utilizando uma base de pontos de interesse do Uruguai disponibilizada pelo *OpenStreetMap*¹. A tabela *nodes* (1842994

¹<http://download.geofabrik.de/south-america/uruguay.html>

tuplas) foi importada para uma instância do banco *Redis* e para o *MongoDB*, ambos os BDs rodando localmente em uma máquina com processador Intel i5-7200U, 8 GB de RAM e Disco rígido de 1 TB. A importação para o *Redis* levou em média 2 minutos e 15 segundos. Já, a importação para o *MongoDB* levou em média 7 min e 53 segundos.

4. Considerações finais

A ferramenta DINO está em sua versão inicial e foi testada com os BDs *PostgreSQL* (*source*) e *MongoDB*, *Redis* e *Cassandra* (*target*), demonstrando bom desempenho por causa do uso de *threads*, além das demais características já citadas. O código fonte está disponível através do link <https://github.com/gbd-ufsc/DINO> e aceita contribuições de terceiros. Como trabalhos futuros, entre outros recursos, pretende-se: melhorar o desempenho da importação; permitir gerar esquemas dos BDs NoSQL criados; adicionar suporte a novos BDs; e, suportar outros tipos de estruturas de entrada, como arquivos CSV (*Comma Separated Values*) e arquivos textos.

Este trabalho foi apoiado por bolsa de iniciação científica (IC) do CNPq.

Referências

- Cattell, R. (2011). Scalable SQL and NoSQL data stores. *ACM SIGMOD Record*, 39(4):12.
- Claudino, M., Souza, D., and Salgado, A. C. (2015). Mapeamentos conceituais entre os modelos Relacional e NoSQL : uma abordagem comparativa. *Revista Principia*, (28):37–50.
- Han, J., Haihong, E., Le, G., and Du, J. (2011). Survey on NoSQL database. *International Conference on Pervasive Computing and Applications, IPCPA 2011*, pages 363–366.
- Mojuprojekty (2017). Código fonte do projeto *sql-to-redis*. Disponível em: <https://github.com/mojuprojekty/sql-to-redis>.
- Murari, M. A., Cunha, G. B. d., and Silveira, S. R. (2016). Desenvolvimento de um software para migração de um banco de dados relacional Firebird, para o não relacional MongoDB. *Revista SETREM*, (28):115–123.
- Poffo, J. P. (2016). *Projeto lógico de bancos de dados NOSQL colunares a partir de esquemas conceituais entidade-relacionamento estendido (EER)*. PhD thesis.
- Ruiz, D. S., Morales, S. F., and Molina, J. G. (2015). Inferring Versioned Schemas from NoSQL Databases and its Applications. *LNCS*, 9381(October):467–480.
- Sadlage, P. J. and Fowler, M. (2012). *NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence*.
- Santos Neto, P. d. A. et al. (2013). Requisitos para ferramentas de migração de dados. In *SBSI - Simpósio Brasileiro de Sistemas de Informação*, pages 887–898. SBC.
- Schreiner, G. A., Duarte, D., and dos Santos Mello, R. (2015). SQLtoKeyNoSQL. In *17th iiWAS*, pages 1–9, New York, New York, USA. ACM Press.
- Vale, F. and Rocha, L. (2011). Nosqlayer: a framework for migrating relational datasets to nosql models. In *REIC*, volume 14. SBC.
- Zhao, G., Lin, Q., Li, L., and Li, Z. (2014). Schema Conversion Model of SQL Database to NoSQL. In *IEEE 3PGCIC*, pages 355–362. IEEE.

Palestras convidadas

Palestras

Nomes, termos, conceitos, significado e outras palavras 140
Renata Vieira - Pontifícia Universidade Católica do Rio Grande do Sul

Introdução à Recuperação de Informações 141
Viviane Moreira - Universidade Federal do Rio Grande do Sul

per:palestra2

Nomes, termos, conceitos, significado e outras palavras

Palestrante: Renata Vieira

Pontifícia Universidade Católica do Rio Grande do Sul

Resumo: Nesta palestra são abordados aspectos da linguagem natural a fim de evidenciar os grandes desafios de se trabalhar nesta área.

Sobre a palestrante: Renata Vieira possui título de PhD em Informática pela Universidade de Edimburgo (1998). É professora da PUC-RS onde atua em pesquisa e ensino na graduação e pós-graduação na área de inteligência artificial, com ênfase em processamento de linguagem natural, representação do conhecimento, ontologias, agentes e web semântica.

per:palestra1

Introdução à Recuperação de Informações

Palestrante: Viviane Moreira

Universidade Federal do Rio Grande do Sul

Resumo: A Recuperação de Informação trata do armazenamento, indexação e busca por informações de natureza não estruturada (texto, imagem, vídeo, etc.). Interagimos diariamente com sistemas de Recuperação de Informação, seja utilizando motores de busca na Web, ou procurando por e-mails em nossos computadores. A palestra tem por objetivo apresentar os principais conceitos da área de Recuperação de Informação para dados textuais. Serão abordadas as etapas de pré-processamento, indexação, consulta, avaliação e coleta de dados na Web.

Sobre a palestrante: Viviane Moreira é Professora do Instituto de Informática da UFRGS, onde desempenha atividades de pesquisa e de ensino tanto na graduação como na pós-graduação. É bolsista de produtividade em pesquisa do CNPq (nível 2). Completou doutorado em Ciência da Computação na Middlesex University em Londres (2004) e mestrado em Ciência da Computação na UFRGS (1999). Suas áreas de pesquisa são Bancos de Dados, Recuperação de Informações e Mineração de Textos. A professora ministra a disciplina de Recuperação de Informações no PPGC da UFRGS há mais de dez anos, tem orientado trabalhos de pesquisa e redigido artigos científicos nesta área.

Minicursos

Minicursos

MongoDB: da teoria ao mundo real	143
<i>Vítor de Ataides - Nodo Digital</i>	
Buscas Semânticas: Fundamentos e Técnicas	144
<i>Giseli Lopes - Universidade Federal do Rio de Janeiro</i>	
Sistemas de detecção de indícios de plágio	145
<i>Solange Pertile - Universidade Federal de Santa Maria</i>	
Firestore: um banco de dados flexível, escalável e em tempo real	146
<i>João Fernandes - Hut8</i>	

MongoDB: da teoria ao mundo real

Vítor de Ataides

Nodo Digital

Resumo: MongoDB é um dos mais utilizados SGBD não relacional. Entre suas características estão indexação, replicação, duplicação de dados e balanceamento de carga. Essas características vêm ao encontro das necessidades das aplicações atuais. Dentre os principais usuários do MongoDB destacam-se Google, Facebook, Ebay e Nokia. Este minicurso será uma jornada, live code, desde a concepção do MongoDB até experiências reais de sua utilização, passando por instalação, operações CRUD, modelagem de dados e indexação.

Sobre o palestrante: Vitor Alano de Ataides é Bacharel em Ciência da Computação, Mestre em Computação com foco em Computação em Nuvem, Doutorando em Computação. Trabalha com desenvolvimento de software há 8 anos. Atualmente é Back-end da Nodo Digital, atuando em projetos como Holonis e Embraer.

Buscas Semânticas: Fundamentos e Técnicas

Giseli Lopes

Universidade Federal do Rio de Janeiro

Resumo: A Recuperação de Informação (RI) trata da representação, armazenamento, organização e acesso a elementos de informação. A Recuperação de Informação, principalmente com o advento da Web, tornou-se parte do dia-a-dia de muitos usuários. Entretanto, a busca através de abordagens tradicionais de RI, partindo de palavras-chave em linguagem natural para recuperar informações não estruturadas, pode se tornar vaga, ambígua e imprecisa. Aproveitando a infraestrutura da Web para permitir o compartilhamento e reuso de dados em uma escala massiva e visando atuar sobre problemas de interoperabilidade semântica, foi criado um conjunto de princípios e tecnologias, conhecido como Dados Interligados (Linked Data). Esse fato impulsionou o desenvolvimento de abordagens para efetuar Buscas Semânticas. Busca semântica é um paradigma de busca que faz uso de semântica explícita para resolver tarefas de busca: interpretar a consulta e os dados; efetuar o casamento entre a consulta e os dados; e ranquear os resultados. Além disso, ao invés de retornar somente documentos (ou páginas Web), possibilitou também prover respostas mais precisas a buscas complexas, capturando informação sobre entidades e seus relacionamentos, através do uso de dados semânticos. Esse minicurso tratará sobre o tema de Buscas Semânticas, focando principalmente em fundamentos e técnicas alinhadas ao uso conjunto de Recuperação de Informação e Dados Interligados.

Sobre a palestrante: Giseli Lopes é Professora Adjunta do Departamento de Ciência da Computação (DCC) da Universidade Federal do Rio de Janeiro (UFRJ). Possui doutorado em Ciência da Computação pela Universidade Federal do Rio Grande do Sul - UFRGS (2012), mestrado em Ciência da Computação pela UFRGS (2007) e graduação em Engenharia de Computação pela Fundação Universidade Federal do Rio Grande - FURG (2004). Já atuou como professora substituta junto ao Departamento de Informática Aplicada da UFRGS e como pós-doutoranda na Pontifícia Universidade Católica do Rio de Janeiro - PUC-Rio junto ao Departamento de Informática - DI. Tem experiência na área de Ciência da Computação, com ênfase em Sistemas de Informação e Banco de Dados, atuando principalmente nos seguintes temas: Sistemas de Recomendação, Dados Interligados, Recuperação de Informação, Redes Sociais e Web Semântica.

Sistemas de detecção de indícios de plágio

Solange Pertile

Universidade Federal de Santa Maria

Resumo: A grande quantidade de documentos disponíveis na Web faz com que seja mais fácil a reutilização de conteúdo de outros autores e torna mais difícil a verificação da originalidade de um determinado texto. Reutilizar texto sem creditar a fonte é considerado plágio. Uma série de estudos relatam a alta prevalência de plágio no meio acadêmico e científico. Neste contexto, muitos pesquisadores e instituições têm se dedicado à elaboração de sistemas para automatizar o processo de verificação de plágio. Este minicurso tem como objetivo apresentar os conceitos, técnicas, e limitações dos Sistemas de Detecção de Plágio, bem como tendências para a área apresentada.

Sobre a palestrante: Solange Pertile é Professora Adjunta do Departamento de Tecnologia da Informação da Universidade Federal de Santa Maria – UFSM, Campus Frederico Westphalen. Possui doutorado em Computação pela Universidade Federal do Rio Grande do Sul - UFRGS (2015), mestrado em Informática pela UFSM (2011) e graduação em Ciência da Computação pela Universidade Regional Integrada do Alto Uruguai e das Missões - URI (2009). Suas áreas de pesquisa são Bancos de Dados, Recuperação de Informações e Mineração de Textos.

er:minicurso4 **Firestore: um banco de dados flexível, escalável e em tempo real**

João Vitor Fernandes

Hut8

Resumo: Neste minicurso teremos uma breve introdução ao Firebase, voltada para a parte de banco de dados, mas sem deixar de lado todas as facilidades providas pela plataforma. Será apresentado o novo banco de dados, atualmente em beta, o Cloud Firestore: um sucessor melhor estruturado do Firebase Realtime Database. Será realizada a construção de um “clone” do Google Keep, realizando desde operações básicas até atualizações em tempo real e sincronização offline.

Sobre o palestrante: João Vitor Fernandes é Graduando em Engenharia de Computação na UFPel. Diretor de Projetos da Hut8. Participou de mais de 30 projetos NoSql e Firebase. Atualmente é sócio e desenvolvedor da startup Donamaid.

Oficinas

Oficinas

Gerenciando dados em fluxos com Apache Storm	148
<i>Guilherme dal Bianco - Universidade Federal da Fronteira Sul</i>	
Modelagem Estatística de Tópicos em Coleções de Documentos.....	149
<i>Denio Duarte - Universidade Federal da Fronteira Sul</i>	
Uma visão sobre Fast-Data: Spark, VoltDB e Elasticsearch.....	150
<i>Luiz Henrique Zambom Santana - Universidade Federal de Santa Catarina</i>	
Gerenciando desenvolvimento de BD: Migrations em PHP	151
<i>Fabio Sperotto - Instituto Federal Sul-rio-grandense</i>	
Emíli@s no país de Banco de Dados.....	152
<i>Nádia Kozievitch e Rita Berardi - Universidade Tecnológica Federal do Paraná</i>	

Gerenciando dados em fluxos com Apache Storm

Guilherme dal Bianco

Universidade Federal da Fronteira Sul

Resumo: Big data tem se tornado uma importante tecnologia capaz de alterar a forma como interagimos. No entanto, lidar com tal volume de informações depende de tecnologias robustas. O processamento em tempo real permite, por exemplo, que seja monitorado o comportamento de grandes volumes de dados a fim de se identificar situações anômalas. Esta oficina irá explorar a plataforma de processamento online Apache Storm. Tal plataforma, utilizada pela Twitter, será explorada durante a oficina a partir de atividades práticas.

Sobre o palestrante: Guilherme Dal Bianco é professor na Universidade Federal da Fronteira Sul (UFFS) onde desempenha atividades de pesquisa e de ensino. Tem doutorado em Ciência da Computação pela Universidade Federal do Rio Grande do Sul (2014) e graduação em Engenharia de computação pela Universidade Federal do Rio Grande (2007). Sua pesquisa concentra-se nas áreas de Banco de dados, Mineração e integração de informações.

Modelagem Estatística de Tópicos em Coleções de Documentos

Denio Duarte

Universidade Federal da Fronteira Sul

Resumo: Modelagem de tópicos é uma importante ferramenta para identificar temas principais (conjunto de palavras) a partir de uma coleção de documentos. Nesta oficina será utilizada a técnica LDA implementada pela biblioteca Gensim do Python.

Sobre o palestrante: Denio Duarte é Doutor em Ciência da Computação pela Université François-Rabelais Tours. Atualmente é professor adjunto da Universidade Federal da Fronteira Sul - UFFS. Atua na área de banco de dados com ênfase em dados semiestruturados e cloud-databases. Participa também em projetos na área de engenharia de software e de aprendizado de máquina.

Uma visão sobre Fast-Data: Spark, VoltDB e Elasticsearch

Luiz Henrique Zambom Santana

Universidade Federal de Santa Catarina

Resumo: Essa oficina apresentará como combinar Spark, VoltDB e Elasticsearch, três tecnologias que materializam os conceitos de Big Data para alcançar velocidade de processamento para um grande volume de dados. Usando um exemplo em informação geográficas, o participante aprenderá como processar dados em tempo real usando Apache Spark, criar visualizações através do Elasticsearch e disponibilizar esses dados de forma escalável em uma ferramenta NewSQL usando o VoltDB.

Sobre o palestrante: Luiz Henrique Zambom Santana possui graduação em Bacharelado em Ciência da Computação pela Universidade Estadual Paulista Júlio de Mesquita Filho (2005), Mestrado em Ciência da Computação pela Universidade Federal de São Carlos (2008) e Mestrado em Negócios Internacionais pela Fachhochschule Mainz (Alemanha) e Universidad de Ciencias Empresariales y Sociales (Argentina). Atualmente cursa doutorado em Ciência da Computação pela Universidade Federal de Santa Catarina. Seus principais interesses de pesquisa são Metodologias para Desenvolvimento de Software, Informática na Educação, Web 2.0, Computação Ubíqua, Big Data, NoSQL e Startups.

Gerenciando desenvolvimento de BD: Migrations em PHP

Fabio Sperotto

Instituto Federal Sul-rio-grandense

Resumo: Esta oficina discute o uso de ORMs (focando em Migrations) para o gerenciamento de mudanças de banco de dados. O foco será dado no desenvolvimento de aplicações em PHP com a aplicação de dois ORMs: Phinx (CakePHP) e Eloquent (Laravel). Estes dois ORMs fazem parte de outros frameworks mas agora estão disponíveis também para uso em outros projetos. Com isso, veremos na prática como podemos utilizá-los e o quanto útil possa ser suas Migrations e Seeds.

Sobre o palestrante: Fabio Sperotto é Bacharel em Sistemas de Informação (Unochapecó) e Mestre em Modelagem Computacional (FURG). Doutorando em Ciência da Computação (UFPEL). Tem 6 anos de experiência em desenvolvimento Web, principalmente com PHP e MySQL. É Professor de Informática no IFSUL – Campus Camaquã.

Emíli@s no país de Banco de Dados. Exclusiva para estudantes de 9º ano do CAIC.

Nádia Kozievitch e Rita Berardi

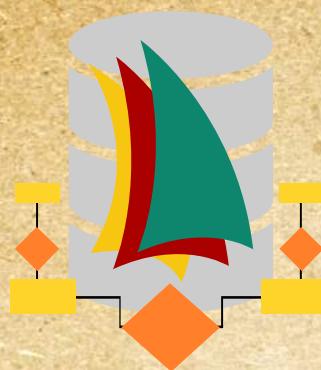
Universidade Tecnológica Federal do Paraná

Resumo: Esta oficina tem como objetivo instigar a curiosidade sobre o tema Banco de Dados. A intenção é divulgar a área de Computação para despertar o interesse de estudantes do ensino médio/tecnológico ou dos anos finais do ensino fundamental, para que conheçam melhor a área e, desta forma, motivá-los a seguir carreira em Computação, que historicamente tem sido predominantemente escolhida pelo público masculino. Dentre o material utilizado, busca-se focar em um impacto visual, em uma integração com temas atuais (como redes sociais, YouTube, etc.), em aplicações atuais, e em possibilidades de continuar o aprendizado (em fontes externas, como banco de dados e aplicações para crianças, tutoriais de SQL, entre outros).

Sobre as palestrantes:

Nádia Puchalski Kozievitch possui graduação em Ciências da Computação pela Universidade Federal do Paraná (2001), mestrado em Informática pela Universidade Federal do Paraná (2005) e doutorado em Ciências da Computação pela Universidade Estadual de Campinas (2011). Trabalhou em projetos de P&D na área de telefonia na IBM (2006-2012); e na Companhia Paranaense de Energia (Copel/Simepar), na área de meteorologia (1999 -2004). Atualmente é professora efetiva da Universidade Tecnológica Federal do Paraná (UTFPR). Seus interesses englobam cidades inteligentes, bibliotecas digitais, GIS e recuperação de informação baseada em conteúdo.

Rita Cristina Galarraga Berardi é Professora (Adjunto) na Universidade Tecnológica Federal do Paraná - UTFPR, Campus Curitiba. Doutora em Ciência da Computação pela Pontifícia Universidade Católica do Rio de Janeiro desenvolvida em colaboração com a Universität Koblenz-Landau, Instituto West de Web Semântica na Alemanha. Completou o mestrado em Ciência da Computação pela Pontifícia Universidade Católica do Rio Grande do Sul, realizado em parceria com Hewlett Packard (HP). Graduada em Ciência da Computação pela Universidade Federal de Pelotas. As áreas de pesquisa que mais interessam estão relacionadas a Banco de Dados, Qualidade dados, Linked data (ou dados conectados), Web Semântica e Ontologias.



XIV
ERBD
Escola Regional de Banco de Dados
2018
Rio Grande-RS



[/erbd.sbc](https://www.facebook.com/erbd.sbc)
<http://www.sbc.org.br/erbd2018>

Realização:



Organização:



Patrocínio:



Apoio:

