



IFSUL 2015
Passo Fundo - RS
20 a 22 de Outubro de 2015

Anais do
II Simpósio de Informática
Big Data

simpinf.passofundo.ifsul.edu.br



II Simpósio de Informática IFSUL Passo Fundo

Passo Fundo - RS - Brasil

ANAIS

Organização Editorial

Alex Sebben da Cunha - IFSUL Passo Fundo

Alexandre Tagliari Lazzaretti - IFSUL Passo Fundo

Coordenação

Élder F. F. Bernardi - IFSUL Passo Fundo

Rafael M. Bertei - IFSUL Passo Fundo

Realização

Instituto Federal Sul-Riograndense - Câmpus Passo Fundo

Apoio

BI do Brasil

International Business Machines - IBM

Sociedade Brasileira de Computação - SBC

Prefácio

Sejam bem-vindos a segunda edição do Simpósio de Informática do IFSUL Câmpus Passo Fundo, ocorrido nos dias 20, 21 e 22 de outubro de 2015. O simpósio é um evento realizado pelos cursos Superior de Tecnologia em Sistemas para Internet e Técnico em Informática.

O evento busca motivar o estudante para exercer a atividade científica e tecnológica, uma vez que proporciona um espaço de divulgação e discussão de temas relevantes e atuais na área de informática desenvolvidos em sua instituição. É nossa aspiração estimular no estudante o interesse pela observação, construção do conhecimento, troca de experiências, destacando assim o comprometimento da instituição com o ensino, pesquisa e a extensão. Além disso, buscamos oferecer, juntamente com o II Simpósio, uma diversificada programação, com palestras, oficinas e minicursos.

O tema do simpósio este ano é "**Big Data - Da infraestrutura à descoberta do conhecimento**", abrangendo o crescimento, a disponibilidade e o uso exponencial das informações obtidas de dados estruturados e não-estruturados.

Em relação aos trabalhos submetidos, foram totalizados 27 trabalhos. Sendo aceitos os 14 trabalhos com melhor média resultante das avaliações dos pares. Dentre os trabalhos os 3 primeiros colocados recebem uma premiação no evento de encerramento.

A realização do simpósio não seria possível sem o trabalho incessante da comissão organizadora do eventos, principalmente dos coordenadores, Prof. Élder F. F. Bernardi e Prof. Rafael M. Bertei, e do auxílio de todos os professores do Instituto Federal Sul-Riograndense Câmpus Passo Fundo.

Cabe um agradecimento aos alunos, alunos bolsistas e técnicos administrativos que auxiliaram na realização das tarefas demandadas pelo simpósio. Além disto, um agradecimento especial aos membros do comitê de avaliação dos trabalhos científicos e a todos que participaram do evento, muito obrigado!

Um ótimo evento a todos!

Alexandre Tagliari Lazzaretti
Roberto Wiest
Vanessa Lago Machado

Comissão de Trabalhos Científicos

Coordenação

Coordenação geral:

Prof. Me. Élder F. F. Bernardi

Prof. Me. Rafael M. Bertei

Comissão de trabalhos científicos:

Prof. Dr. Alexandre Tagliari Lazzaretti

Prof. Me. Roberto Wiest

Prof. Vanesa Lago Machado

Comissão das palestras e minicursos:

Prof. Me. André Fernando Rollwagen

Prof. Me. Lisandro Lemos Machado

Aluno bolsista:

Gustavo Castro

Comitê de Programa

Adilso Souza	IFSUL - Passo Fundo
Alexandre Tagliari Lazzaretti	IFSUL - Passo Fundo
Alexandre Zanatta	Universidade de Passo Fundo
Amilton Martins	IMED - Passo Fundo
Anderson Ritta	IFSUL - Camaqua
André Rollwagen	IFSUL - Passo Fundo
Carlos A. Petry	IFSUL - Passo Fundo
Carmen Scorsatto	IFSUL - Passo Fundo
Denis Carvalho	Instituto Federal de Minas Gerais
IFSUL - Passo Fundo	IFSUL - Passo Fundo
Evandro Kuszera	UTFPR
Fahad Kalil	Faculdade Meridional - IMED
Jair Ferronato	IFSUL - Passo Fundo
Jorge Bavaresco	IFSUL - Passo Fundo
José de Figueiredo	IFSUL - Passo Fundo
José Maurício Cunha Fernandes	Embrapa Trigo
Joseane Amaral	IFSUL - Passo Fundo
Josue Toebe	IFSUL - Passo Fundo
Lisandro Machado	IFSUL - Passo Fundo
Maikon Santos	IFSUL - Passo Fundo
Marcos José Brusso	UPF
Mirkos Ortiz Martins	UNIFRA - Centro Universitário Franciscano
Rafael Bertei	IFSUL - Passo Fundo
Ricardo Dallasen	IFSUL - Passo Fundo
Roberta Ciocari	IFSUL - Passo Fundo
Roberto Wiest	IFSUL - Passo Fundo
Suellen Sotille	IMED
Sylvio Vieira	UNIFRA - Centro Universitário Franciscano
Vanessa Lago Machado	IFSUL - Passo Fundo
Willingthon Pavan	Universidade de Passo Fundo

Comunidade virtual de aprendizagem cooperativa para a geração de ideias inovadoras no setor público	8
Clarissa Felkl Prevedello (Universidade Federal do Rio Grande do Sul - Brazil), Antônio Carlos da Rocha Costa (FURG - Brazil)	
Análise de Desempenho de Arquiteturas de Osciladores Controlados por Tensão (VCOs) para Aplicação em Circuitos Integrados Digitais CMOS	18
Ricardo Dallasen (IFSUL - Passo Fundo - Brazil), Carlos A. Petry (IFSUL - Passo Fundo - Brazil), José de Figueiredo (IFSUL - Passo Fundo - Brazil) Alexsander Carneiro (IFSul - Brazil)	
Método de busca em acervos digitais por feedback de relevância orientado a termos	27
Fernando Hattori (University of São Paulo - Brazil), Edson Gomi (University of São Paulo - Brazil)	
XConversor: Uma Ferramenta de Conversão de Artigos do Formato HTML para XML Aplicada ao Domínio Médico	37
Arthur Carosia (Instituto Federal de São Paulo - IFSP - Brazil)	
Sistema para Auxiliar Administração de Medicamentos em Ambientes de Vivência Assistida	47
Milene Santos Teixeira (UFSM - Brazil), Leandro Oliveira Freitas (Federal University of Santa Maria - Brazil), Un Hee Schiefelbein (Universidade Federal de Santa Maria - Brazil), Juçara Gubiani (Universidade Federal de Santa Maria - Brazil)	
Redes Neurais com os pacotes Neuralnet e Shiny	55
Jorge Bavaresco (IFSUL - Passo Fundo - Brazil), Ronaldo Serpa Rosa (UPF - Brazil)	
Identificação de Situações de Alunos em um Ambiente Educacional Ubíquo: Explorando o Processamento Dinâmico de Regras Semânticas	65
Lidiane Costa da Silva (UFPeI - Brazil), João Ladislau Lopes (IFSUL and UFRGS - Brazil), Ana Marilza Pernas (UFPeI - Brazil)	
Uma Abordagem para Suporte à Decisão Clínica Baseada em Semântica no Domínio da Nefrologia	75
Esau Tavares (UFPB - Brazil), Natasha Lino (UFPB - Brazil)	

Simulação de VANETs utilizando ferramentas Omnet++, SUMO e VEINS	85
Jair Ferronato (IFSul - Instituto Federal Sul-Rio-Grandense - Brazil), Marco Trentin (UPF - Brazil)	
Processo de Engenharia do Conhecimento para o Desenvolvimento de uma Ontologia do Domínio da Doença Renal Crônica para Capacitação de Profissionais de Atenção Primária à Saúde	95
Sidney Campos (Universidade Federal da Paraíba - Brazil), Natasha Lino (Universidade Federal da Paraíba - Brazil)	
GROSS: Um framework para auxiliar na experimentação de escalonadores de Sistemas Operacionais	105
Ádler Gross (Instituto Federal Sul-rio-grandense (IFSUL) - Brazil), Elder Bernardi (Instituto Federal Sulriograndense - Brazil)	
Desenvolvimento e execução de jogos eletrônicos através do recurso Canvas do HTML5	115
Alisson Tres (IFSul - Brazil), Lisandro Machado (Instituto Federal Sul-Rio-Grandense - Brazil)	
Desenvolvimento de Aplicação Interativa para Visualização e Análise de Dados	123
Samantha Gonçalves (Instituto Federal Sul-RioGrandense Campus Passo Fundo - Brazil)	

Comunidade virtual de aprendizagem cooperativa para a geração de ideias inovadoras no setor público

Clarissa Felkl Prevedello¹, Antônio Carlos da Rocha Costa²

¹Programa de Pós-graduação em Informática na Educação – Universidade Federal do Rio Grande do Sul (UFRGS) – Porto Alegre – RS – Brazil

²Programa de Pós-Graduação em Computação – Universidade Federal do Rio Grande (FURG) – Rio Grande – RS – Brazil

clarissaprevedello@gmail.com, ac.rocha.costa@gmail.com

Abstract. *Innovation's culture has made increasingly part of public administration today. Thus, this work presents a proposal to encourage the generation of innovation in the public sector, proposing the use of virtual learning communities to generate new ideas, seeking a change in the way of acting and thinking involved in the process, through the development of a learning environment and openness to change, included with the social aspects. For this paper present a proposal the use for a virtual community as a way to develop cooperation between public servants aiming at innovation, focusing on the process of generating innovation.*

Resumo. *O estímulo a uma cultura de inovação tem feito cada vez mais parte da gestão pública. Nesse contexto este trabalho apresenta uma proposta para a geração de inovação no setor público, propondo a utilização de comunidades virtuais de aprendizagem para a geração de novas ideias, visando uma mudança no modo de agir e pensar dos envolvidos no processo, por meio do desenvolvimento de um ambiente de aprendizagem e a abertura para as mudanças, aliado aos aspectos sociais. Para isso apresenta-se um conjunto de pressupostos para a configuração de uma comunidade virtual como forma de desenvolver a cooperação entre servidores públicos utilizando visando à inovação, focando no processo de geração de inovação.*

1. Introdução

Existe uma necessidade de mudança de cultura na administração pública em direção ao risco e da experimentação, ocasionando uma necessidade de reforma no setor público. Evans (2013) argumenta que o sucesso dessa reforma depende do desenvolvimento de governança centrada no cidadão, sustentada pelo conceito de valor público, mas para o cumprimento desse objetivo é fundamental o estabelecimento de uma cultura de inovação também de valor público. Isso torna, segundo o Design Council (2011), de extrema importância por parte dos governos os estímulos para uma cultura de inovação que esteja em consonância com o que já ocorre no setor privado, a empresa pública também deve se posicionar como empresa inovadora.

Mas ainda existem algumas barreiras para estabelecer uma cultura de inovação na gestão pública, entre elas: a estrutura das organizações, que restringe a criatividade, dificulta a comunicação, a compreensão e, por consequência, a colaboração, impedindo

que as novas ideias se espalhem em torno das organizações ou atravessem as barreiras departamentais; restrições financeiras e políticas impostas pelos governos centrais que perderam a simpatia pelos valores e práticas de empresas públicas; grande quantidade de supervisão que os governos enfrentam em muitos setores; a dificuldade de definir inovação na gestão pública pelos próprios servidores públicos, visto que não existem roteiros para conduzir o caminho e os meios para propor novas formas de fazer ainda não estão claros para a maioria; a falta de estímulo para a maioria dos servidores públicos em assumir riscos, tentar coisas novas e oferecer soluções para problemas criativamente; a existência de poucos programas para ajudar a lidar com a burocracia, evitar armadilhas e implementar ideias; dificuldades em mensurar o sucesso dificultam que o governo justifique um orçamento para pesquisa e desenvolvimento, pois o principal objetivo não é o lucro e se torna difícil medir o impacto; os sistemas de governo não são projetados para promover os servidores inovadores, as recompensas muitas vezes vão para aqueles que cumprem as expectativas e não para aqueles que estabelecem novos níveis de expectativas, o que resulta em servidores que, muitas vezes, preferem ficar com os procedimentos operacionais padrão a tentar algo novo; e diversas outras (Partnership for Public Service e IDEO, 2011, Design Council, 2008, Clarke e Clegg, 1999).

Esses fatores reforçam a ideia de que o governo carece de um ambiente de inovação de base em que novas ideias possam surgir a qualquer momento e em qualquer contexto (Partnership for Public Service e IDEO, 2011), ou seja, um ambiente de aprendizagem e abertura às mudanças, aliado aos aspectos sociais (Clarke e Clegg, 1999) e que possa contribuir para uma mudança de postura em direção a quebra dessas barreiras, abrindo caminho para a inovação de valor público.

Nesse sentido, existem algumas alternativas que se baseiam na utilização da aprendizagem por meio da cooperação em ambientes virtuais. Evans (2013) relata que o paradigma do valor público requer uma mudança de compromisso em que os gestores para trabalhar com redes de discussão e execução motivem o envolvimento em redes, parcerias e relacionamentos, muitas vezes formados no contexto de igualdade e aprendizado mútuo. Johnson (2011) acredita que é necessário investir cada vez mais no trabalho cooperativo com a utilização da Internet, pois seu uso pode alterar de maneira fundamental o modo de cultivar e promover ideias, estimulando o governo a pensar sobre si mesmo como uma plataforma aberta em vez de uma burocracia centralizada. Para Moore (1995) apud Ferrarezzi e Amorin (2007), um estudo sobre inovação no serviço público pode focar o processo que gerou inovação de base e em modos de disseminar ideias, dessa maneira, pode-se partir da geração de ideias inovadoras. Wenger (2006) também defende o compartilhamento de conhecimento aberto, por meio da adoção de comunidades como uma maneira de diminuir as barreiras departamentais, a formalidade e a burocracia do setor público.

Nessa perspectiva, apresenta-se neste trabalho pressupostos para a configuração de uma comunidade virtual de aprendizagem cooperativa que estimule a geração de ideias inovadoras no setor público baseada nos conceitos de Inovação na gestão pública, Design thinking, Aprendizagem Cooperativa e Comunidades Virtuais.

2. Inovação na Gestão Pública e o Design thinking

2.1. Inovação na Gestão Pública

A Escola Nacional de Administração Pública (ENAP) define a inovação no setor público como: mudanças que se processam de forma incremental, sendo comum que as práticas antigas não sejam completamente extintas pelo surgimento das novas, ou que as novas práticas sejam implementadas por meio de estruturas paralelas, reconhecendo o valor das pequenas mudanças na geração de um ambiente institucional de inovações na gestão federal, em longo prazo (Ferrarezzi e Amorin, 2007).

Mas o fundamental a ser considerado em inovação no setor público é não considerá-la como um fim em si, mas por sua capacidade de criar valor público, ou seja, novas ideias devem ser trabalhadas em função da criação de valor público, diferente da inovação no setor privado com foco voltado para o desenvolvimento econômico. Produzir valor público significa levar em consideração os benefícios para os cidadãos e outras partes interessadas; na sua essência, a inovação é a resolução de problemas para as pessoas (Partnership for Public Service e IDEO, 2011, Design Council, 2008).

Devido ao seu caráter de relação direta com a sociedade, a inovação é encarada como um fator crítico para a sobrevivência da empresa pública democrática, pois será a capacidade de mudar e responder às novas demandas sociais que possam surgir que vai determinar a sua longevidade ou sustentabilidade, mas as mudanças necessárias só serão possíveis, segundo Manzini (2008), Clarke e Clegg (1999) e Lundvall (2002) se o setor público estiver disposto a aprender, se tornar aberto e flexível, mas para isso, as pessoas devem reconhecer e compreender a importância da aprendizagem como modo de se adaptar às inovações, em termos dos valores que sustentam os resultados sociais. Dessa maneira, inovar na gestão pública não se trata simplesmente de propor mudanças de uma maneira geral, mas trata-se de um processo que requer aprendizagem e mudança de pensamento, uma abordagem disciplinada e uma forte liderança (Partnership for Public Service e IDEO, 2011).

Nessa perspectiva as iniciativas de inovação na gestão pública têm crescido nos últimos anos (Design Council, 2008). Ocorreu uma liberação de ousadia e inovação onde antes havia conformidade e tradição (Clarke e Clegg, 1999). Liberação que ocorreu por mudanças ocasionadas na gestão pública na virada do século XX para o XXI, com ênfase explícita sobre a gestão da mudança visando uma metamorfose na cultura da organização, melhorando a qualidade do serviço, movendo o *locus* da autoridade gerencial e responsabilidade orçamental do centro da política para mais próximo do ponto de entrega, aproximando-se do público e tentando mudar o equilíbrio de poder em favor de quem a organização se destina a servir (Evans, 2013, Clarke e Clegg, 1999), ou seja, em direção ao valor público.

Desse modo, para este trabalho segundo Clarke e Clegg, 1999, Design Council, 2008, Evans, 2013, Ferrarezzi e Amorin, 2007, Lundvall, 2002, Manzini, 2008, Partnership for Public Service e IDEO, 2011 define-se inovação na gestão pública como; ideias que produzem mudanças, independente da escala, podendo ou não extinguir as anteriores, mas que possuem fundamentalmente a capacidade de criar valor público, devendo levar em consideração se irão beneficiar os cidadãos e outras partes interessadas.

A aprendizagem relacionada ao desenvolvimento pessoal também é considerada como um fator importante para consolidar a inovação na gestão pública, pois proporciona a abertura para a geração de novas ideias e mudanças, podendo ocasionar uma mudança de postura nos servidores públicos, em direção ao reconhecimento e compreensão da importância da aprendizagem como modo de se adaptar às inovações, em termos dos valores que sustentam os resultados sociais.

2.2. O Design thinking como estratégia para promover a cooperação e geração de ideias inovadoras no setor público

Por relacionar-se diretamente como o conceito de inovação na gestão pública o Design thinking representa uma alternativa para promover esse tipo de inovação, pois consiste em uma maneira de compreender as necessidades dos indivíduos e promover seu bem estar por meio do trabalho cooperativo. Trabalhos recentes demonstram que o uso estratégico de técnicas de design colaborativo ajuda inovações a serem implementadas em torno e através das organizações (Design Council, 2008 e Manzini, 2008). Baseado nas semelhanças entre os dois conceitos é que este estudo propõe a utilização do Design thinking como estratégia de geração de ideias inovadoras no setor público, a partir do pensamento dos designers, utilizando-se das comunidades virtuais e da aprendizagem cooperativa.

Assim como o conceito de inovação centrado apenas em avanços e mudanças na esfera pública é menos sustentável, dependendo de novas escolhas que vão em direção ao valor público, o Design thinking se baseia nessa nova abordagem, de são necessárias novas escolhas que equilibrem as necessidades de indivíduos e da sociedade como um todo; novas ideias que lidem como os desafios globais de saúde, pobreza e educação, novas estratégias que resultem em diferenças que importam em um senso de propósito, que inclua todas as pessoas envolvidas (Brown, 2010). Como resultado da busca por novos caminhos para a inovação surgiu o conceito que hoje é de “uma abordagem focada no ser humano que vê na multidisciplinaridade, colaboração e tangibilização de pensamentos e processos, caminhos que levam a soluções inovadoras para negócios” (Vianna et al., 2012, p. 12).

Design thinking é uma metodologia para a prática, a resolução criativa de problemas ou questões que pretende com seu resultado desenvolver um futuro melhor. É uma forma de se concentrar na solução de um problema que se inicia com um objetivo a ser alcançado, em vez de começar com um determinado problema (Cross, 2006).

É a maneira que os designers pensam: os processos mentais que eles usam para projetar objetos, serviços ou sistemas, é diferente do resultado final, produtos elegantes e úteis. Design thinking é a natureza do trabalho de design: um projeto de fluxo de trabalho baseado em torno de problemas. Diferente do trabalho dos gestores, cujo fluxo de trabalho é centrado em torno de atribuições permanentes, designers trabalham em uma base ‘projeto’ (Dunne e Martin, 2006). O designer sabe que para identificar os reais problemas e solucioná-los de maneira mais efetiva, é preciso abordá-los sob diversas perspectivas e ângulos. Assim, prioriza o trabalho colaborativo entre equipes multidisciplinares, que trazem olhares diversificados e oferecem interpretações variadas sobre a questão e, assim, soluções inovadoras (Vianna et al., 2012, p. 13).

Uma característica importante que diferencia o Design thinking de outras formas de resolução de problemas é a abordagem por meio de *wicked problems*. São problemas

pertencentes a classe dos fatores sociais, geralmente mal formulados, em que as informações são confusas, há muitos clientes e tomadores de decisão com valores conflitantes e onde as ramificações em todo o sistema são completamente confusas e não podem ser redefinidas para solução racional, pois são problemas complexos que resistem à definição conclusiva, não havendo uma única maneira adequada de entender o que é o problema (Buchanan, 1992, Cross, 2006, Dunne e Martin, 2006, Rittel e Webber, 1973, Tonkinwise, 2012,).

Isso torna o processo de design um tipo de resolução de problemas em que o solucionador vê o problema ou age como se as metas tivessem sido formuladas de maneira equivocada. Isso permite que se mudem as condições iniciais ou que as transformações sejam permitidas; a implicação é que os designers são projetistas, mesmo quando eles poderiam ser solucionadores de problemas (Cross, 2006).

Devido a sua complexidade, os gestores evitam trabalhar *wicked problems*; em contrapartida, designers abraçam esse tipo de problema como um desafio e desenvolvem uma maneira de resolvê-los baseada em projetos, desenvolvendo também uma forma de pensar distinta do convencional, o Design thinking (Dunne e Martin, 2006).

Com a noção de que todos podem trabalhar pensando como designers, mesmo que ainda resistente, o mito do gênio criativo acaba se desfazendo e juntamente com isso, a noção de que as grandes ideias são formadas por mentes brilhantes, em feitos de imaginação bem além da capacidade de meros mortais (Brown, 2010 e Amabile, 1996). Os projetos de design envolvem um alto grau de criatividade, mas de uma maneira controlada e direcionada pelo processo. Assim, a criatividade é canalizada para a produção de uma solução prática e viável para o problema de design (Ambrose e Harris, 2011). Dessa maneira, ideias inovadoras podem ser resultado de muito trabalho, potencializado por um processo de descoberta criativa centrado no ser humano e seguido por metodologias voltadas para esse fim, como no caso do Design thinking. Portanto, há uma prática de projetar o pensamento (Cross, 2006).

2.3. O Processo de Design thinking

O processo de Design thinking, do ponto de vista de Ambrose e Harris (2011), Brown (2008 e 2010), Cross (2006) e Vianna et al. (2012), de uma maneira geral se baseia em três principais fases: Formulação do problema, Geração de ideias e a Implementação.

A primeira, destinada para a Formulação do problema é baseada em pesquisas e imersão no problema, serve como referência para as demais fases; a segunda, a geração de ideias, destina-se a gerar as ideias, testar os protótipos e selecionar a melhor e mais adequada para a fase seguinte, a de implementação, em que a proposta será colocada em prática e encaminhada para a produção. É importante ressaltar também que os autores falam em avaliação durante todo o processo, pois trata-se de um processo iterativo.

Compreendendo melhor o processo, parte-se para a sua relação com a aprendizagem, tentando aproximar o Design thinking da educação, para entender como pode ser aplicado no desenvolvimento de ideias inovadoras em comunidade virtuais.

2.4. O Design thinking e aprendizagem

Por se tratar de sistematizar a maneira de pensar dos designers enquanto estão desenvolvendo seus projetos (Dunne e Martin, 2006, Tonkinwise, 2012), o Design

thinking tem sido muito utilizado para promover a aprendizagem (Burdick e Willis, 2011, Cross, 2006, Southgate, Reynolds e Howley, 2013). Nessa perspectiva, é importante entender o modo como os designers pensam, ou seja, os aspectos cognitivos que envolvem o pensamento em design, exigidos nas duas primeiras fases do processo de Design thinking: a Formulação do problema e a Geração de ideias.

A primeira fase, a de formulação do problema, recebe uma atenção especial, pois envolve ‘encontrar’ problemas adequados, bem como o modo como serão ‘resolvidos’; isso envolve uma atividade substancial de estruturação e formulação e não apenas a simples aceitação do ‘problema dado’ (Cross, 2006) e a geração de ideias por meio do pensamento abdutivo que podem oferecer as bases para um modo de pensamento voltado para a inovação.

Cross (2006) e Dune e Martin (2006), defendem a utilização do Design thinking na educação não simplesmente como a aplicação de uma metodologia baseada em uma sequência, mas das ‘maneiras de entender/pensar design’, (*designerly ways of knowing*), interpretado como um núcleo de conhecimentos que tem valor educativo intrínseco e não somente de fornecer as condições instrumentais que estão associadas com a educação profissional do design tradicional. Tonkinwise (2012), Cross (2006), Dunne e Martin (2006) e Buchanan (1992) identificam alguns aspectos do ‘*designerly ways of knowing*’ nessas duas fases do processo que são fundamentais para o entendimento do design como parte da educação geral: designers enfrentam *wicked problems* e o modo de resolver problemas é encontrar uma ‘solução focada’, baseada em três aspectos do modo de pensar dos designers: o cognitivo, o afetivo e o interpessoal. Dune e Martin (2006) defendem a ideia de que, para resolver os *wicked problems*, os designers desenvolveram uma maneira de pensamento distinta do pensamento convencional que deve ser estimulada para profissionais de todas as áreas.

3 Aprendizagem Cooperativa, Comunidades Virtuais e a Inovação

3.1. Aprendizagem Cooperativa

A expansão da aprendizagem cooperativa teve seu avanço pela difusão da computação e da Internet, que ocasionou também um crescimento na demanda por sistemas de suporte para a educação *on-line*, propiciando dessa maneira que a aprendizagem baseada na internet se utilize do que há de melhor nas redes de computadores: a possibilidade de comunicação e cooperação entre os pares, (Campos et al., 2003) “resultando no desenvolvimento de aplicações que possibilitam o trabalho e a produção em conjunto” (Behar, 1998, p. 47). Dessa maneira, a partilha de conhecimentos e aprendizagem é ativada pelas tecnologias, que permitem exercer na prática o conceito de cooperação, quando um indivíduo busca o conhecimento e o outro contribui para a sua busca (Wachter, Gupta, Quaddus, 2000).

Aprender por cooperação significa construir coletivamente os conhecimentos, por meio da interação social (Piaget, 1976). As interações entre os sujeitos são o resultado de eles estarem inseridos em um sistema de relações com outros sujeitos, objetos de outra natureza, construindo um mundo de significações (Piaget, 1973).

O elemento motor da interação social é o mecanismo de equilibração que pode solucionar as visões diferentes que os integrantes de um grupo possuem frente à solução de um problema, chamados de conflitos sociocognitivos e que, juntamente com a

equilíbrio, mobilizam e forçam as reestruturações intelectuais e o progresso cognitivo dos sujeitos (Behar, 1998, Piaget, 1976). Dessa maneira, uma situação de cooperação parte das relações estabelecidas entre sujeitos que estão de alguma maneira interligados, uma vez que, para promover a aprendizagem utilizando a cooperação, é importante entender como se estabelecem essas relações entre os sujeitos envolvidos na cooperação.

Por se tratar especificamente de aprendizagem cooperativa, dentre as diferentes teorias que se propõem a estudar e aplicar métodos para promover esse tipo de aprendizagem, foi selecionada a da interdependência social (Johnson e Johnson, 2009), pois se baseia na interdependência que se estabelece entre os membros do grupo por meio de objetivos comuns, permitindo fazer uma relação direta com o método de Design thinking, o qual também se baseia em estabelecer objetivos para a resolução dos *wicked problems*.

3.2. Comunidades Virtuais e a Inovação

As comunidades virtuais são um exemplo de ambiente computacional em que “podem ocorrer situações de troca de informações e ideias, que são base para busca de solução de um problema ou, ainda, do ‘crescimento’, (desenvolvimento cognitivo) de indivíduos e de um grupo” (Behar, 1998, p.50).

Nas comunidades virtuais as pessoas se encontram, mas não simplesmente se encontram, elas se comportam de maneira ativa, se propondo a fazer uma atividade e esse aspecto de fazer uma atividade é o que destaca as comunidades em relação a uma página da internet que desperta nas pessoas uma relação passiva. Por esses motivos, a definição de comunidade *on-line* necessariamente tem que prever a participação (Rheingold, 1996) e por consequência, a cooperação que pressupõe também a relação, por meio de laços. O valor da utilização de uma comunidade virtual está no fato de as pessoas se considerarem seus membros, encontrando valor nas interações. Sendo ativos na construção da comunidade, os membros sentem um senso de propriedade e têm interesse em seu sucesso, por isso, a comunidade deve ser moldada em torno dos interesses dos usuários, que devem ter a capacidade de moldar os ambientes virtuais, criar relacionamentos e aprender. (Wachter, Gupta, Quaddus, 2000).

As características apresentadas acima, como, por exemplo, o local em que as pessoas estão dispostas a cooperar, ou seja, trocar ideias e conhecimentos, de maneira ativa, participativa, incentivadas por um interesse comum, vão ao encontro dos elementos propostos na teoria da interdependência social de Johnson e Johnson (2009) que propõe melhorar a eficácia da cooperação: a interdependência positiva, a responsabilidade individual, a promoção da interação, o uso adequado de habilidades sociais e o processamento de grupo. Todos esses fatores fazem das comunidades virtuais locais propícios para a aprendizagem.

Os elementos acima citados também interferem positivamente no crescimento do interesse em pesquisas que relacionam comunidades virtuais e aprendizagem, originando termos como ‘comunidades de aprendizes’, ‘comunidades de discurso’, ‘comunidades de aprendizagem’, ‘construção do conhecimento comunidades’, ‘comunidades escolares’ e ‘comunidades de prática’ e, independentemente do termo utilizado, trabalhar com comunidades tornou-se obrigatório para inovações educacionais, embora ainda pouco se saiba sobre o valor educacional da contratação de um modelo comunitário de apoio de aprendizagem (Barab, 2003).

Dessa maneira, pretende-se associar os elementos da teoria da interdependência social com o espaço que se mostra ideal para que esse tipo de relação ocorra, as comunidades virtuais, em busca da construção de requisitos para a configuração de uma comunidade virtual que englobe as noções de cooperação e geração de ideias inovadoras para o setor público.

4. Comunidade virtual de aprendizagem cooperativa que estimule a geração de ideias inovadoras no setor público

Com base no referencial teórico adotado na pesquisa elaborou-se a um quadro base para a configuração de uma comunidade virtual com foco na aprendizagem cooperativa entre servidores públicos de diferentes instituições para promover a inovação no setor público.

Partindo-se do conceito de inovação no setor público adotado na pesquisa, abre-se caminho para o desenvolvimento de uma comunidade virtual baseada na aprendizagem cooperativa, tendo como base a Teoria da Interdependência Social (Johnson e Johnson, 2009). Instalado o ambiente para a cooperação, pode-se adotar as técnicas de Design thinking. Primeiramente se propõe para os membros da comunidade *wicked problems*, estabelecendo o estado de tensão e o movimento na direção do cumprimento das metas, com a utilização do *designerly ways for knowing*. Os pressupostos da comunidade virtual delimitados pela pesquisa pode ser visualizados na Figura 1.

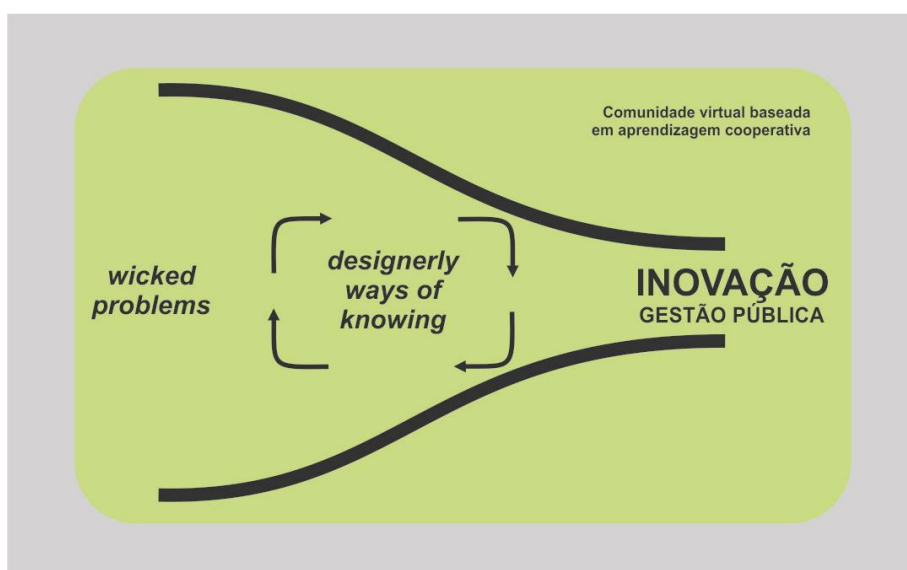


Figura 2. Comunidade virtual de aprendizagem cooperativa que estimula a geração de ideias inovadoras no setor público.

5. Conclusão e perspectivas para trabalhos futuros

Com o estabelecimento do conceito de inovação no setor público baseado em ideias que produzem mudanças, mas que devem fundamentalmente possuir a capacidade de criar valor público, com sua essência na resolução de problemas para as pessoas baseada também na mobilização em direção a aprendizagem relacionada ao desenvolvimento pessoal, podendo ocasionar uma mudança de postura nos servidores públicos, estabelece-se uma estreita relação entre o conceito de Design thinking que

envolve a resolução criativa de problemas sociais, equilibrando as necessidades dos indivíduos com a sociedade como um todo, uma abordagem focada no ser humano que se utiliza da multidisciplinaridade, colaboração e tangibilização de pensamentos e processos para encontrar soluções inovadoras baseada em *wicked problems*.

Entende-se que o ambiente ideal para se estabelecer aprendizagem por meio da cooperação em comunidades virtuais, pois tratam de ambientes de participação ativa e que estimulam a troca. Dessa maneira, para a geração de ideias inovadoras no setor público, foram elaborados os requisitos de comunidade virtual com o objetivo de desenvolver entre servidores públicos um ambiente virtual cooperativo, visando à inovação, construída a partir dos conceitos de inovação no setor público e no estabelecimento de *wicked problems*, baseada no contexto dos profissionais envolvidos, o que vai gerar um movimento em direção à resolução dos problemas através da cooperação, na tentativa de estabelecer a equilibração.

Como trabalhos futuros pretende-se ainda a validação dos pressupostos por meio da configuração de comunidades virtuais com servidores públicos nas quais serão desenvolvidas pesquisas qualitativas das cooperações propiciadas pelo ambiente.

References

- Amabile, T. M. (1996) "Creativity and Innovation in Organizations", In: Harvard Business School Publishing, January, 5, p. 1-15.
- Ambrose, G. e Harris, P. (2011) Design Thinking, Boockmann, Porto Alegre.
- Barab, S. A. (2003) "An Introduction to the Special Issue: Designing for Virtual Communities in the Service of Learning", In: The Information Society, n. 19, p. 197–201.
- Behar, P. A. (1998) "Análise operatória de ferramentas computacionais de uso individual e cooperativo". 140 f. Tese (Doutorado em Informática na Educação) - Curso de Pós-Graduação em Ciência da Computação – CPGCC, Universidade Federal do Rio Grande do Sul - UFRGS, Porto Alegre.
- Brown, T. (2010) Design Thinking, Elsevir, Rio de Janeiro.
- Brown, T. (2008) "Design Thinking", In: Harvard Business Review, jun., p.1-11.
- Buchanan, R. (1992) "Whicked problems in Design Thinking", In: Design Issues. v. 8, n. 2, p. 5-21.
- Burdick, A.; Willis, H. (2011) "Digital learning, digital scholarship and design thinking", In: Design Studies, v. 32, n. 6, nov., p. 546-556.
- Campos, F. C. A. et al. (2003) "Cooperação e aprendizagem on-line", DP&A, Rio de Janeiro.
- Clarke, T. e Clegg, S. (1999). "Changing Paradigms in Public Service Management", In: Administrative Theory & Praxis, v. 21, n. 4, dec., p. 485-490, <http://www.jstor.org/stable/25611388>.
- Cross, N. (2006) Designerly ways of knowing, Springer-Verlag, London.
- Design Council (2008). "Briefing 2: The role of design in public services", www.designcouncil.org.uk/briefing02.

- Design Council (2011). "Design for innovation: Facts, figures and practical plans for growth", <http://www.designcouncil.org.uk>.
- Dunne, D.; Martin, R. (2006) "Design Thinking and How It Will Change Management Education: An Interview and Discussion", In: Academy of Management Learning & Education, v. 5, n. 4, p. 512–523.
- Evans, M. (2013) "Participação social: lições aprendidas da Europa", In: Evans, M., Ferrarezzi, E., Oliveira, C. G., Grau, N. C. (autores), Antero, S. A., Salgado, V. A. B. (Organização). Participação Social: textos para discussão. Fundação Instituto para o Fortalecimento das Capacidades Institucionais – IFCI, AECID, Editora IABS, Brasília-DF, Brasil.
- Ferrarezi, E. e Amorin, S. (2007) "Concurso inovação na gestão pública federal no Brasil: análise de uma trajetória". In: Cadernos ENAP, 32. Brasília.
- Johnson, S. (2011) De onde vêm as boas ideias. Tradução Maria Luiza Borges. Zahar, Rio de Janeiro, e-book.
- Johnson, D. W. e Johnson, R. T. (2009) "An Educational Psychology Success Story: Social Interdependence Theory and Cooperative Learning". In: Educational Researcher, v. 3, n. 5, p.365-379.
- Lundvall, B. et al. (2002) "National systems of production, innovation and competence building", In: Research Policy, v. 31, is. 2, feb., p. 213-231.
- Manzini, E. (2008) Design para a inovação social e sustentabilidade: comunidades criativas, organizações colaborativas e novas redes projetuais. e-papers, Rio de Janeiro, e-book.
- Partnership for Public Service e IDEO (2011). "Innovation in Government", <http://ourpublicservice.org/OPS/publications/searchresults.php>.
- Piaget, J. (1973) Estudos Sociológicos. : Forense, Rio de Janeiro.
- Piaget, J. (1976) A equilibração das estruturas cognitivas, Zahar, Rio de Janeiro.
- Rheingold, H. (1996) A comunidade virtual", Gradiva, Lisboa.
- Rittel, H. W. J. e Webber, M. M. (1973) "Dilemmas in a General Theory of Planning", In: Policy Sciences, 4, p. 155-169, Amsterdam.
- Southgate, E., Reynolds, R. e Howley, P. (2013) "Professional experience as a wicked problem in initial teacher education", In: Teaching and Teacher Education, v. 31, apr., p. 13-22.
- Tonkinwise, C. (2012) "The Grammar of Design Thinking: The Histories and Promises of Socio-Material Practice Remaking". In: DRAFT. p. 1-35.
- Vianna, M. et al. (2012) Design thinking: inovação em negócios. Rio de Janeiro: MJV Press, 2012.
- Wachter, R. M., Gupta, J. N.D. e Quaddus, M. A. (2000) "IT takes a village: Virtual communities in support of education", In: International Journal of Information Management, v. 20, is. 6, dec., p. 473-489.
- Wenger, E. (2006) "Communities of practice: a brief introduction", <http://www.ewenger.com/theory/index.htm>.

Análise de Desempenho de Arquiteturas de Osciladores Controlados por Tensão (VCOs) para Aplicação em Circuitos Integrados Digitais CMOS

Ricardo Vanni Dallsen, Carlos Alberto Petry, José Antônio Oliveira de Figueiredo, Alexsander Furtado Carneiro

Instituto Federal Sul-rio-grandense (IFSul)

Estrada Perimetral Leste, 150 – 99064-440 – Passo Fundo – RS – Brazil

{ricardo.dallsen, carlos.petry, jose.figueiredo, alexsander.carneiro}
@passofundo.ifsul.edu.br

Abstract. *This work presents an analysis of 3 different single ended ring VCOs, suitable for clock generation in CMOS digital circuits. Three different VCOs were designed for a 45nm technology and simulated using NGSPICE tool. Current starved VCOs attained highest operating frequency and lowest electrical consumption, suitable for circuits such as high performance microprocessors and low power applications. Series resistance VCO achieved lowest gain, suitable for low jitter applications, for instance data converters.*

Resumo. *Este trabalho realiza a análise de 3 arquiteturas de VCOs em anel com saída simples, voltados para a geração de clock em circuitos integrados digitais CMOS. Foram projetados 3 VCOs em tecnologia de processo de 45nm e simulados na ferramenta NGSPICE. Os VCOs do tipo current starved obtiveram as maiores frequências de operação e menor consumo elétrico, sendo aptos a serem integrados em processadores de alto desempenho e para aplicações de baixo consumo. A arquitetura de VCO com resistência em série obteve o menor ganho, sendo mais apta a aplicações que necessitem de baixo jitter, como conversores de dados.*

1. Introdução

A maioria dos circuitos integrados digitais modernos são síncronos, o que significa que é necessária a existência de um sinal de *clock* para cadenciar suas operações, controlando o fluxo de dados e sinais de controle. Para realizar o ajuste de frequência e fase de PLLs (*Phase Locked Loops*) é necessário um oscilador controlável (VCO). Para a geração de sinais de *clock* em circuitos integrados digitais são geralmente utilizados PLLs com VCO em anel. Este tipo de VCO é empregado devido à sua grande faixa de ajuste de frequência, sua dimensão reduzida e capacidade de implementação diretamente no processo de fabricação de circuitos integrados digitais, não requerendo etapas adicionais no processo [Dallsen 2012; Fahim 2005; Miyazaki et al 2004].

O artigo está organizado da seguinte maneira: a seção 2 mostra uma breve fundamentação teórica de osciladores controlados por tensão, na seção 3 são abordadas as arquiteturas analisadas; na seção 4 é realizada a análise das arquiteturas abordadas em 3 aspectos, frequência de operação, ganho e dissipação de potência e na seção 5 são apresentadas as conclusões.

2. Operação de Osciladores Controlados por Tensão (VCOs)

Um oscilador em anel consiste em N estágios de ganho em um laço. Isto pode ser visto com mais detalhes na Figura 1, que mostra a arquitetura de um oscilador em anel de 3 estágios. Devido à realimentação positiva da malha é gerada uma defasagem entre os sinais de entrada e de saída da malha, o que altera constantemente a tensão de saída do laço, fornecendo a frequência de saída [Baker 2010];

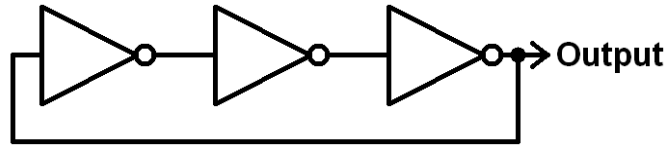


Figura 1: Oscilador em anel de 3 estágios.

Os VCOs possuem uma entrada de uma tensão de controle e uma saída, cuja frequência é proporcional a esta tensão de controle. O comportamento de um VCO pode ser modelado a partir da equação (1) [Fahim 2005; Baker, 2010].

$$f_{out} = f_0 + K_{VCO} \cdot V_{ctrl} \quad (1)$$

onde V_{ctrl} é o sinal de controle, f_0 é a frequência a vazio, isto é, sem sinal de controle ($V_{ctrl} = 0$), e K_{VCO} é o ganho ou sensibilidade do VCO. O parâmetro K_{VCO} representa o ganho do oscilador, que é a taxa em que a frequência de saída varia em relação à tensão aplicada a sua entrada. Para obter o valor de K_{VCO} , pode-se derivar a sua função de transferência, ou calcular através da equação (2).

$$K_{VCO} = \frac{f_2 - f_1}{V_{ctrl2} - V_{ctrl1}} \quad (2)$$

em que f_2 é a frequência obtida usando uma tensão de controle V_{ctrl2} , e idem para f_1 e V_{ctrl1} . Quanto maior a faixa de frequência que o VCO conseguir operar, maior será a faixa de retenção e a faixa de captura do PLL. Entretanto, quanto maior o ganho do oscilador, mais predisposto a *jitter* (e a ruído de fase) ele será, pois pequenas variações na tensão de controle produzirão variações consideráveis da frequência de saída [Mandal and Sarkar 2010; Aranda et al 2013; McNeill and Ricketts 2009].

3. Arquiteturas de VCOs analisadas

Uma abordagem para implementação de osciladores em anel é através da utilização de inversores lógicos. Este tipo de implementação é simples, uma vez que não utiliza componentes passivos nem circuitos de partida externos. Entre as vantagens deste tipo de VCO destacam-se: (i) baixo ruído intrínseco devido à pequena quantidade de componentes necessários, (ii) simplicidade de implementação e (iii) baixo consumo elétrico.

3.1 Arquitetura Current Starved

Para ajustar a frequência de oscilação, podemos limitar o fornecimento de energia para os inversores do anel, o que aumenta o tempo de carga da capacitância equivalente da saída. Para realizar a limitação da corrente são inseridas fontes de corrente na alimentação de cada estágio do VCO. A Figura 2 mostra uma célula de um oscilador

current starved. Os transistores M2 e M3 compõem o inversor e M1 e M4 limitam a corrente de alimentação. Os sinais V_{ctrl_p} e V_{ctrl_n} ajustam a corrente fornecida para o inversor. Com menor capacidade de corrente, a carga dos *gates* do inversor do estágio seguinte ocorre mais lentamente, provocando alterações na frequência de oscilação do anel.

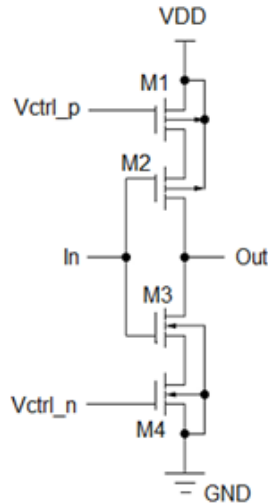


Figura 2: Célula de um VCO *current starved*.

A Figura 3 mostra a representação esquemática de um oscilador de 5 estágios composto por células *current starved*. Os dois transistores com as setas (à esquerda) são utilizados para gerar uma réplica complementar da tensão de controle (V_c nesta figura), utilizada para controlar as fontes de corrente do ramo superior.

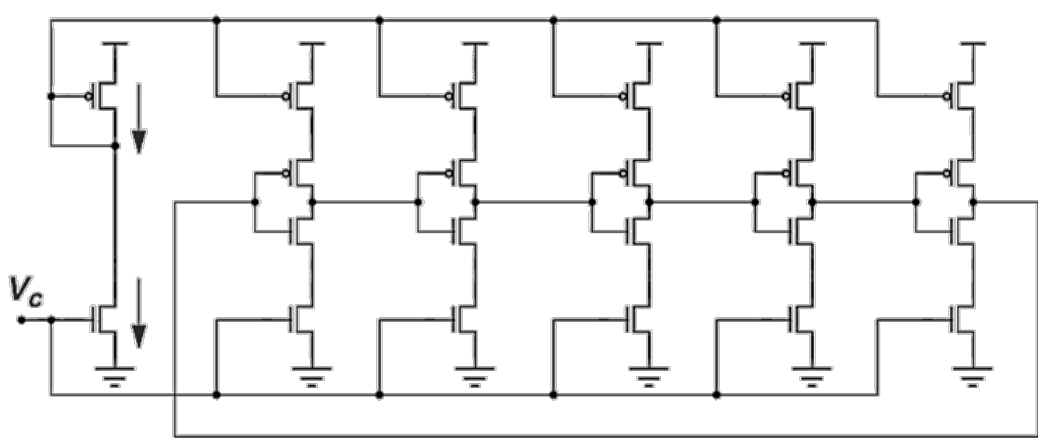


Figura 3: Circuito completo de um VCO *current starved*.

3.2 Arquitetura Current Starved Simple

Esta arquitetura é uma variação da *current starved* apresentada anteriormente. A diferença consiste na limitação da corrente do oscilador. Ao invés de ser realizada com uma fonte de corrente para cada estágio, como mostrado anteriormente na Figura 3, esta variação emprega fontes de corrente que limitam o oscilador como um todo. Isto é mostrado em detalhes na Figura 4. Neste tipo de VCO também é necessário utilizar um circuito para a geração da tensão de controle complementar, mostrado anteriormente na Figura 3. Na Figura 4 este item foi omitido para simplificar o diagrama.

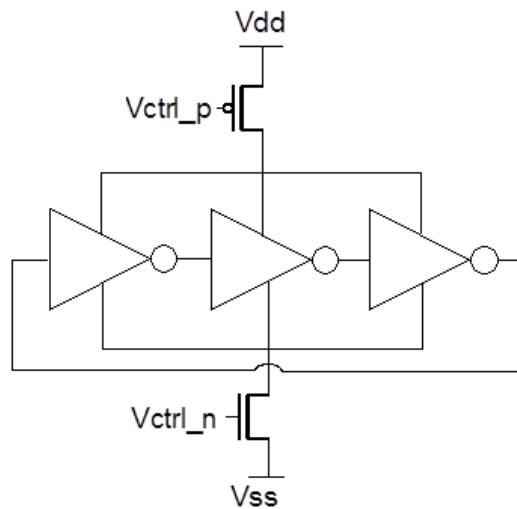


Figura 4: VCO *current starved* com fonte de corrente simples.

3.3 Arquitetura de VCO com resistência em série

Outra maneira de ajustar a frequência é através da inserção de um resistor controlado entre os estágios do anel de inversores, que provoca o aumento do tempo de carga da capacitância do estágio seguinte. Com o aumento do tempo de carga o tempo de transição de nível lógico também aumenta, diminuindo a frequência de oscilação. O circuito da célula utilizada para esta arquitetura é mostrado na Figura 5. Os transistores M1 e M2 formam o inversor e M3 e M4 são responsáveis pela resistência colocada em série na saída dos inversores. Esta resistência é similar a uma porta de transmissão (*transmission gate*).

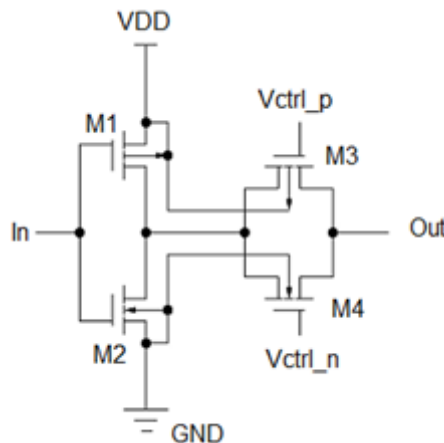


Figura 5: Célula de VCO com resistência em série.

Na Figura 6 é mostrado um VCO utilizando 3 destas células. Para gerar a tensão de controle dos transistores PMOS é utilizado o mesmo esquema mostrado anteriormente na Figura 3, omitido na figura para simplificar o diagrama.

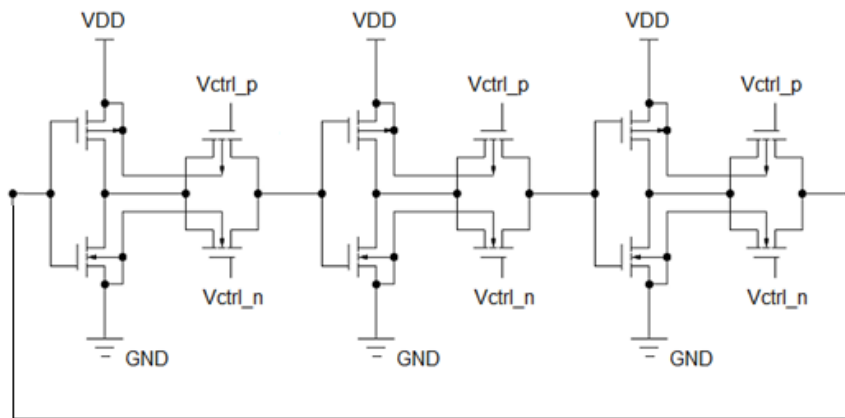


Figura 6: Circuito de VCO com resistência em série.

4. Comparativo de desempenho

Para realizar a comparação entre as arquiteturas de VCOs, foram implementados anéis com 7 estágios para cada uma das arquiteturas. O modelo de tecnologia empregado foi o PTM (*Predictive Technology Model*) de 45nm desenvolvido pela Universidade do Estado do Arizona [ASU]. As dimensões utilizadas para os transistores foram de $W_{\text{NMOS}}=90\text{nm}$ e $W_{\text{PMOS}}=135\text{nm}$, com $L_{\text{NMOS}}=L_{\text{PMOS}}=50\text{nm}$ para todos os transistores. Na saída dos VCOs, foi adicionado um buffer com o mesmo dimensionamento, e como carga do buffer foi utilizado um capacitor de 1fF. Nos transistores responsáveis pela geração da réplica complementar da tensão de controle foi empregado $W_{\text{PMOS}}=135\text{nm}$ e $W_{\text{NMOS}}=180\text{nm}$. A utilização deste tamanho de W_{NMOS} visa aumentar a linearidade do VCO (BAKER, 2010).

A comparação dos VCOs foi realizada sob 3 aspectos: (i) frequência e faixa de operação, (ii) ganho do VCO e (iii) consumo elétrico. A tensão foi variada em passos de 0,5V na faixa superior de operação (entre 1V até 0,65V). Abaixo disto, o passo foi reduzido para 0,25V (entre 0,625 até o mínimo suportado pelo VCO em análise).

4.1 Frequência de operação

Frequência de operação é a principal característica de um oscilador. Para os VCOs temos, além disto, sua faixa de operação que compreende o intervalo de frequências em que o VCO consegue fornecer uma saída. Na Figura 7 é mostrado o gráfico da frequência de operação em função da tensão de controle aplicada. O *current starved* multi foi assim chamado por empregar uma fonte de corrente independente para cada terminal de alimentação de cada estágio, e o *current starved* simples por utilizar apenas uma fonte coletiva para todo o oscilador, servindo para todos os estágios.

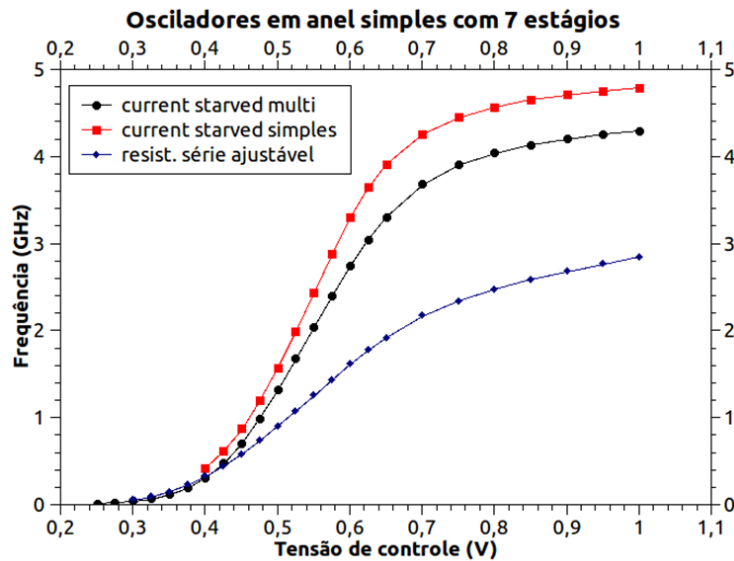


Figura 7: Frequência em função da tensão de controle.

Como mostrado na Figura 7, o *current starved* simples (CSS) obteve a maior frequência de operação, 4,76GHz ($V_{ctrl}=1V$). O *current starved* multi (CSM) com a mesma tensão de controle operou a 4,3GHz e o com resistência em série (RS) apenas 2,847GHz. A frequência mínima de operação foi de 424MHz para o *current starved* simples, 260MHz para o *current starved* multi e 60MHz para o com resistência em série.

A faixa de operação dos VCOs é de 4,28GHz para o *current starved* multi, 4,36GHz para o *current starved* simples e 2,79GHz para o com resistência em série. Este parâmetro influencia, por exemplo, na faixa de atuação de um PLL. Com faixas de operação altas podem ser implementadas técnicas de redução da frequência de operação visando redução do consumo elétrico [Fahim 2005] ou compensação de degradação de desempenho [Dallasen 2012].

4.2 Ganho do VCO (K_{VCO})

Este fator mostra a sensibilidade do oscilador, ou seja, o quanto ele altera sua frequência para uma dada variação na sua tensão de controle. Quanto maior o ganho do oscilador, mais sensível a variações na sua tensão de controle. Variações na tensão de controle podem ocorrer, por exemplo, por variações na temperatura do circuito integrado, na tensão de alimentação, entre outros fatores, resultando em *jitter* (ruído de fase em domínio frequência) na saída do VCO [Mandal and Sarkar 2010; Aranda et al 2013]. Assim quanto maior for o ganho, mais propenso a *jitter* será o VCO.

Para obter os parâmetros de ganho dos VCOs, foi realizada a derivada das curvas da Figura 7. O resultado obtido é mostrado na Figura 8.

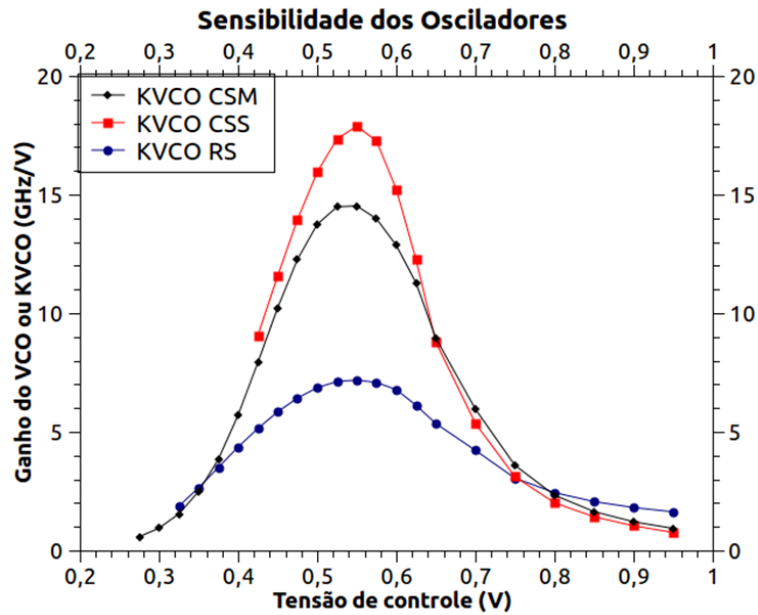


Figura 8: Ganho KVC em função da tensão de controle.

O VCO *current starved* simples obteve o maior ganho entre as 3 arquiteturas de VCOs. Este comportamento era esperado, visto que para uma mesma tensão de controle, esta arquitetura foi a que obteve a maior frequência e faixa de operação entre todas. Os VCOs *current starved* apresentam um pico de ganho elevado em relação à opção com resistência em série, que obteve o maior ganho linear entre as arquiteturas analisadas. Esta característica do ganho do VCO com resistência em série torna-se útil em circuitos que necessitam de baixo *jitter*, como por exemplo, conversores de dados [Aranda 2013; McNeill and Ricketts 2009].

4.3 Dissipação de potência

O consumo elétrico é um fator chave em aplicações cuja alimentação se dá por baterias. A redução do consumo elétrico é sempre bem vinda em dispositivos móveis e sistemas embarcados, aumentando a autonomia sem a necessidade do emprego de baterias com maior capacidade.

Na Figura 10 são mostradas as curvas da potência em função da frequência de operação para cada uma das 3 arquiteturas abordadas.

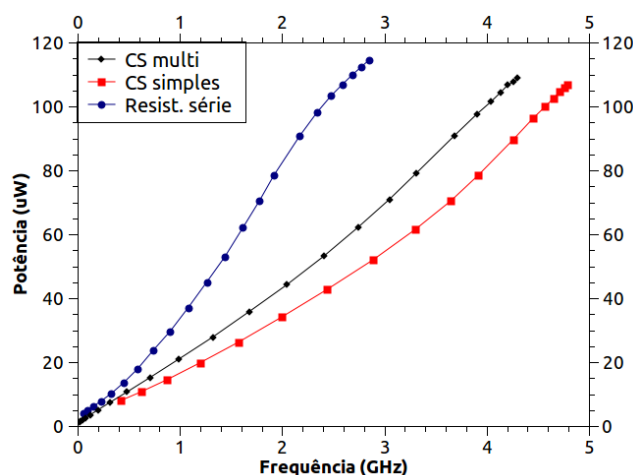


Figura 9: Comparativo da potência dissipada das 3 arquiteturas de VCO

Observando o gráfico, nota-se que para frequências menores que 500MHz a potência dissipada pelas 3 arquiteturas é similar. Para gerar um sinal de 2GHz, o *current starved* multi dissipou $44\mu\text{W}$ enquanto o *current starved* simples dissipou $34\mu\text{W}$. O VCO com resistência em série dissipou $80\mu\text{W}$, 57% mais que o melhor caso (*current starved* simples). Com base nos resultados obtidos é possível concluir que aplicações de baixo consumo são beneficiadas com a utilização de VCOs do tipo *current starved*.

5. Conclusões

Dentre as arquiteturas de VCO simuladas, a *current starved* simples obteve a maior frequência de operação, o maior ganho e bem como o menor consumo, seguido de perto da *current starved* multi. Devido a estas características, tais arquiteturas são mais adequadas para circuitos integrados que operam em altas frequências, como por exemplo, processadores. Aplicações que requeiram baixo consumo de energia e/ou que sejam alimentadas por baterias, como por exemplo, processadores de smartphones, também são atendidas por estas arquiteturas.

A arquitetura de VCO com resistência em série apresentou o menor ganho sendo adequada para aplicações que necessitam de baixo *jitter*, como por exemplo conversores de dados. Entretanto, a maior dissipação de potência em relação aos VCOs *current starved* pode ser fator limitante para sua aplicação.

Referências

- Aranda, M. L.; Díaz, O. G.; Álvarez, C. R. (2013) "A Performance Comparison of CMOS Voltage-Controlled Ring Oscillators for its Application to Generation and Distribution Clock Networks", Science Journal of Circuits, Systems and Signal Processing. Vol. 2, No. 2, 2013, pp. 56-66.
- ASU. Arisona State University. Predictive Technology Model (PTM). Disponível em: <<http://ptm.asu.edu/>>. Acesso em: 06 Abril 2015.
- Baker, J.; (2010) CMOS Circuit Design, Layout, and Simulation, Wiley 3^a. ed.
- Dallasen, R.V.; Wirth, G.I.; Both, T.H., (2012) "A PLL for clock generation with automatic frequency control under TID effects," Integrated Circuits and Systems

Design (SBCCI), 2012 25th Symposium on , vol., no., pp.1,5, Ago. 30 2012-Set. 2 2012

Fahim, A. M.; (2005) Clock Generators for SOC Processors: Circuits and Architectures, Springer, 2005 edition.

Mandal, M K.; Sarkar, B C.; (2010) Ring oscillators: Characteristics and applications In Indian Journal of Pure & Applied Physics (IJPAP) Vol. 48, pp136-145

McNeill, J., Ricketts, D.; (2009) The Designer's Guide to Jitter in Ring Oscillators, Springer 1ed ed.

Miyazaki, T.; Hashimoto, M.; Onodera, H., (2004) "A performance comparison of PLLs for clock generation using ring oscillator VCO and LC oscillator in a digital CMOS process," Design Automation Conference, 2004. Proceedings of the ASP-DAC 2004. Asia and South Pacific, vol., no., pp.545 - 546, 27-30 Jan. 2004

Método de busca em acervos digitais por feedback de relevância orientado a termos

Fernando Hattori¹, Edson Satoshi Gomi¹

¹Departamento de Engenharia da Computação – Escola Politécnica
Universidade de São Paulo (USP) – São Paulo – SP – Brazil

hattori@usp.br, gomi@usp.br

Abstract. *The Vector Space Model is the most widely used information retrieval model within digital libraries' systems. However, the relevance feedback algorithms developed to be used with VSM model are not global nor permanent. This paper presents a novel method that uses of relevance feedback named terms oriented. In this method, users' feedbacks lead to modifications in the terms' vectors representations. These modifications are global and permanent, influencing further searches. An experiment was conducted using the ClueWeb09 dataset, giving evidence that this method improves the quality of search results when compared with Vector Space Model.*

Resumo. *O modelo de recuperação de informação mais utilizado no contexto de acervos digitais é o Vector Space Model, porém os mecanismos de feedbacks de relevância implementados para este modelo não possuem uma estratégia global e permanente. Este trabalho apresenta um novo método de feedbacks de relevância denominado orientado a termos, permitindo que as modificações realizadas por influência dos feedbacks dos usuários sejam globais e permanentes. Foram realizados experimentos utilizando o dataset ClueWeb09 que dão evidências de que o novo método melhora a qualidade dos resultados da busca em relação ao modelo tradicional Vector Space Model.*

1. Introdução

O advento da *World Wide Web* (WWW) e a diminuição dos custos das tecnologias de informação e comunicação tornaram possível a criação e a disponibilização de grandes acervos digitais na internet. Diante do grande volume de informações disponíveis e da necessidade de permitir o acesso a documentos ou itens rapidamente, o motor de busca é considerado um importante mecanismo para os acervos digitais [Schwartz 2000].

Entretanto, com a utilização dos motores de busca, o usuário recebe como resultado de uma consulta muitos documentos que podem ou não ser revelantes para a sua busca. Além disso, as regras que definem a ordenação dos documentos pode não refletir a necessidade do usuário, obrigando o usuário a avaliar vários itens para encontrar aqueles que são de seu interesse. Adicionalmente, os usuários se concentram principalmente nos primeiros resultados, sendo que dificilmente analisam todos os documentos entregues pelo motor de busca [Pass et al. 2006]. Uma forma de minimizar este problema é o uso de algoritmos que melhorem a ordenação dos resultados da busca, facilitando o trabalho do usuário e aumentando a probabilidade de que este encontre os documentos que busca.

Os modelos de recuperação da informação representam o modo com que os objetos digitais são selecionados num acervo digital e ordenados de acordo com o critério de relevância, para comporem as respostas às consultas feitas pelos usuários. O modelo de recuperação de informação mais utilizado no contexto de acervos digitais é o *Vector Space Model* (VSM). Este define um espaço vetorial como arcabouço para descrever os documentos (d_i) e as consultas (q_j). Cada documento e cada consulta são representados no modelo como um vetor com n dimensões, onde n é a quantidade de termos encontrados no vocabulário. Cada termo é equivalente a uma palavra que ocorre em pelo menos um documento [Salton et al. 1975].

$$\begin{aligned} d_i &= \{d_{i1}, d_{i2}, d_{i3}, \dots, d_{in}\} \\ q_j &= \{q_{j1}, q_{j2}, q_{j3}, \dots, q_{jn}\} \end{aligned}$$

Cada uma das dimensões deste espaço vetorial representa um dos termos do vocabulário. Os valores d_{ik} e q_{jk} correspondem ao “peso” (ou importância) que o termo k possui dentro do documento i e da consulta q_j , respectivamente. A similaridade entre documento e consulta ($sim(d_i, q_j)$) é calculada através de um produto vetorial entre os vetores que representam o documento e a consulta (Equação 1). Desse modo, o mecanismo de busca pode ordenar os documentos em ordem decrescente de similaridade [Salton et al. 1975].

$$sim(d_i, q_j) = d_i \cdot q_j = |d_i||q_j| \cos(\theta) \quad (1)$$

Uma das deficiências desse modelo é o fato do modelo não aproveitar o *feedback* dos usuários na melhoria dos resultados da busca [Carpineto and Romano 2012, Buckley et al. 1994]. Como solução para contornar este problema, existe um grande conjunto de implementações de diferentes mecanismos que utilizam *feedbacks* de relevância recebidos dos usuários na tentativa de melhorar a qualidade dos resultados da busca. Mas essas implementações possuem algumas deficiências, tornando as modificações produzidas através dos *feedbacks* locais ou temporárias [Bodoff et al. 2001].

Diante dessas deficiências, o objetivo deste trabalho de pesquisa é apresentar um novo método de *feedback* de relevância denominado orientado a termos.

Na Seção 2 são descritas as duas estratégias dos algoritmos de *feedback* de relevância existentes e suas deficiências. Na Seção 3 é descrita a estratégia de *feedback* de relevância orientado a termos e na Seção 4 é descrito o algoritmo que foi desenvolvido para a realização dos experimentos comparando esta estratégia e o tradicional VSM. Os dados utilizados no experimento são descritos na Seção 5 e as etapas desenvolvidas para o experimento, na Seção 6. Na Seção 7 são apresentados os resultados do experimento e na Seção 8 são apresentadas as conclusões obtidas dos resultados e alguns trabalhos futuros.

2. Trabalhos relacionados

A maior parte dos algoritmos de *feedback* de relevância são orientados a consulta, ou seja, os *feedbacks* obtidos de um usuário são utilizados para modificar a consulta do próprio usuário e refazê-la, utilizando, por exemplo, o algoritmo de Rocchio [Rocchio 1971], ilus-

trado na Equação 2 (modificado de modo a apresentar somente os *feedbacks* positivos).

$$q_j^{modificada} = \alpha * q_j^{inicial} + \beta * \frac{1}{\|D_r\|} \sum_{i=1}^r d_i \quad (2)$$

tal que, $d_i \in D_r$

Onde $q_j^{inicial}$ e $q_j^{modificada}$ são as consultas inicial e modificada, respectivamente, α e β são constantes que determinam o quanto os julgamentos influenciarão a consulta modificada e D_r é o conjunto de documentos julgados relevantes pelo usuário através de *feedback*. Alguns exemplos da utilização da estratégia de *feedback* de relevância orientada a consultas são os algoritmos de Limbu et. al. [Limbu et al. 2006] e Ma e Lin [Ma and Lin 2014].

Porém, a estratégia de *feedbacks* orientados a consulta possui uma deficiência no fato de que os *feedbacks* de relevância para uma determinada consulta são utilizados apenas para aquelas consultas realizadas em uma mesma sessão do usuário. Ou seja, mesmo que um usuário em uma sessão realize todo o processo de envio de *feedbacks* diante de uma consulta, as informações dos *feedbacks* recebidos não são utilizados para melhorar as próximas consultas em sessões posteriores dele mesmo ou de outros usuários. Além disso, já que este tipo de mecanismo exige mais interações do usuário com o sistema, muitos deixam de utilizá-lo.

Uma segunda estratégia de utilização das informações obtidas a partir dos *feedbacks* de relevância é a orientada a documentos, na qual os *feedbacks* são utilizados para modificar a representação dos documentos. Esta estratégia é apresentada principalmente como uma solução para a deficiência da perspectiva orientada a consultas, pois a modificação sobre a representação dos documentos é permanente, influenciando as consultas realizadas em sessões seguintes [Bot and Wu 2004, Brauen et al. 1968]. Partindo da consulta inicial e dos julgamentos devolvidos pelo usuário, o sistema modifica as representações dos documentos, tentando minimizar a distância entre os documentos considerados relevantes e a consulta, utilizando por exemplo o algoritmo de Brauen [Brauen et al. 1968], ilustrado na Equação 3.

$$d_i^{modificado} = d_i^{inicial} + \alpha * (q_j - d_i^{inicial}) \quad (3)$$

Onde $d_i^{inicial}$ e $d_i^{modificado}$ são as representações do documento i antes e depois das modificações motivadas por *feedback* do usuário e q_j é a consulta realizada. Alguns exemplos da utilização da estratégia de *feedback* de relevância orientada a documentos são os algoritmos de Bot e Wu [Bot and Wu 2004], Fuhr e Buckley [Fuhr and Buckley 1990] e Belew [Belew 1989]. Porém, esta estratégia também possui suas próprias deficiências, sendo que a principal delas está no fato de os *feedbacks* modificarem os documentos de forma local, ou seja, somente aqueles documentos explicitamente citados pelos usuários como relevantes são modificados. Como os usuários normalmente julgam ou visualizam somente os primeiros documentos devolvidos pela busca, vários dos documentos potencialmente relevantes são ignorados e podem ser preteridos em consultas futuras.

Desse modo, uma alternativa para estas duas estratégias na utilização dos *feedbacks* de relevância é a utilização do algoritmo proposto neste trabalho, denominado "Feedback Orientado a Termos".

3. Feedback de relevância orientado a termos

No método proposto, os *feedbacks* de relevância recebidos são utilizados para melhorar a compreensão do modelo sobre a relação entre os termos do vocabulário, criando relações entre os termos utilizados na consulta e os termos dos documentos apontados como relevantes pelo usuário. As alterações realizadas por esta nova estratégia serão permanentes, porque as informações obtidas do processamento dos *feedbacks* alteram as representações dos termos e essas representações modificadas são utilizadas nas próximas sessões deste e de outros usuários, ao contrário das modificações sobre consultas. Além disso, as alterações também são globais, porque as representações modificadas dos termos influenciam potencialmente o cálculo de similaridade entre a consulta do usuário e qualquer documento do acervo, mesmo aqueles documentos não julgados previamente pelo usuário.

4. Algoritmo de feedback de relevância orientado a termos

O algoritmo de *feedback* de relevância orientado a termos é baseado no modelo VSM e também representa documentos e consultas como vetores de n dimensões, onde n é o tamanho do vocabulário. No modelo VSM, cada um dos termos é representado por um dos vetores unitários e ortogonais entre si que formam a base vetorial do espaço de representação dos documentos [Wong et al. 1985, Salton 1989].

Este algoritmo foi desenvolvido com a capacidade de receber e processar os *feedbacks* devolvidos pelos usuários de modo que eles sejam utilizados para alterar a representação dos termos do vocabulário no modelo. No modelo de *feedback* orientado a termos, os vetores que representam os termos (v_k) não são sempre unitários e ortogonais entre si como no modelo VSM e sofrem alterações de acordo com os *feedbacks* recebidos, influenciando diretamente no cálculo de similaridade entre documentos e consultas.

O funcionamento do algoritmo é dividido em 4 etapas sendo elas, indexação, coleta de *feedback*, transformação dos vetores de termos e busca.

4.1. Indexação

A indexação funciona como no modelo VSM apresentado por Salton [Salton et al. 1975], no qual o peso d_{ij} de cada termo sobre os documentos é medido utilizando o *tf-idf*. tf_{ij} se refere à frequência do termo j dentro do documento i e idf_j se refere ao inverso da frequência de documentos que possuem o termo j . O peso d_{ij} recebe o valor de $tfidf_{ij}$ e é calculado da seguinte maneira:

$$tfidf_{ij} = (1 + \log f_{ij}) * \log \frac{|D|}{|\{d_i \in D : f_{ij} > 0\}|} \quad (4)$$

Onde f_{ij} é a frequência (ou contagem de ocorrências) bruta do termo j no documento i e $|\{d_i \in D : f_{ij} > 0\}|$ é a quantidade de documentos do acervo que possuem pelo menos uma ocorrência do termo j .

As funções logarítmicas são utilizadas para diminuir o quanto a frequência do termo no documento influencia no valor do “peso”, pois é exagerado imaginar que um termo que ocorre 20 vezes em um documento seja 20 vezes mais relevante que um termo que ocorre somente uma vez. Além disso, as funções logarítmicas também são usadas

para diminuir o fator *idf*, pois os valores de *idf* são potencialmente muito maiores que os valores de *tf* [Manning et al. 2009].

Nesta etapa, além da indexação padrão do VSM, também é realizada a inicialização dos vetores de termos de acordo com a Equação 5.

$$v_{km} = \begin{cases} 0 & \text{se } k \neq m \\ 1 & \text{se } k = m \end{cases} \quad (5)$$

4.2. Coleta de feedbacks

Esta etapa é responsável pela coleta ou recuperação dos *feedbacks* fornecidos pelos usuários. Os *feedbacks* dos usuários informam quais documentos foram considerados relevantes diante de quais consultas. A partir dos *feedbacks* são formados dois conjuntos de termos: um dos conjuntos é formado pelos termos da consulta realizada e o outro é formado pelos termos mais representativos dos documentos considerados relevantes nos *feedbacks* dos usuários. O segundo conjunto é formado pelos 20 termos mais representativos dos documentos considerados relevantes, sendo que são considerados representativos aqueles termos que possuem os maiores valores de *tf-idf* para cada documento relevante.

No passo seguinte, são gerados pares ordenados (t_k, t_m) que relacionam um dos termos da consulta e um dos termos dos documentos, a partir do produto cartesiano entre os dois conjuntos. O conjunto de todos os pares ordenados gerados a partir de todos os *feedbacks* recebidos forma o conjunto F . Cada um dos pares ordenados do conjunto F será analisado e utilizado em transformações sobre os vetores de termos.

4.3. Transformação dos vetores de termos

Cada um dos pares ordenados gerados pela etapa anterior afetarão os vetores de termos nos pontos que representam a relação entre os dois termos do par ordenado. Ou seja, para cada par $(t_k, t_m) \in F$, serão modificados os valores v_{km} e v_{mk} nos vetores de termos de acordo com a Equação 6. Se o valor já presente na matriz seja maior que zero, o novo valor é a multiplicação do valor antigo por $(1+\beta)$, caso contrário, o novo valor é igual β .

$$v_{km}^{novo} = v_{km}^{novo} = \begin{cases} v_{km} + \alpha & \text{se } 0 < v_{km} < \beta \\ \alpha & \text{se } v_{km} = 0 \\ \beta & \text{se } v_{km} \geq \beta \\ 1 & \text{se } k = m \end{cases} \quad (6)$$

Os experimentos realizados durante esta pesquisa indicaram empiricamente que bons valores de α e β são $\alpha = 0,15$ e $\beta = 0,75$.

4.4. Busca no acervo

A busca é realizada seguindo o modelo VSM modificado [Salton 1989]. Essas modificações incluem a relação entre os termos do vocabulário no cálculo da similaridade. A similaridade entre o documento d_i e a consulta q_j é calculado através da Equação 7.

$$sim(d_i, q_j) = d_i \cdot q_j = \sum_{k,m=1}^n d_{ik} q_{jm} (v_k \cdot v_m) \quad (7)$$

Onde v_k e v_m são os vetores que representam os termos t_k e t_m do vocabulário.

5. Descrição dos dados utilizados no experimento

Para avaliar o funcionamento do algoritmo de *feedback* de relevância orientado a termos, o algoritmo proposto teve seu desempenho comparado com o modelo VSM padrão através de um experimento que usou um conjunto de dados (*dataset*) formado por um subconjunto de documentos do *ClueWeb09* [Callan et al. 2009] e pelo conjunto de consultas, *feedbacks* e julgamentos disponibilizados pelo *TREC NIST* no eixo temático de *feedback* de relevância de 2010 (*Relevance Feedback Track* [Buckley et al. 2010]).

O *dataset ClueWeb09* é formado por um conjunto de documentos, um conjunto de consultas e um conjunto de julgamentos de relevância. Os julgamentos de relevância são indicações de quais documentos são relevantes ou não diante de cada uma das consultas. Geralmente, são criados manualmente por especialistas da área, que julgam um a um os documentos diante de cada consulta. Como o processo de criação desses julgamentos é muito caro e trabalhoso, este conjunto de julgamentos é geralmente parcial (nem todos os documentos do acervo são analisados diante de cada consulta) [Zhang and Kamps 2010].

As consultas e os julgamentos disponibilizados pelo *TREC NIST* como parte de um projeto de incentivo à pesquisa em recuperação de informação, provendo a infraestrutura necessária para avaliação de novos mecanismos de recuperação de informação em larga escala.

A avaliação de experimentos em sistemas de recuperação de informações é tradicionalmente realizada através da comparação dos resultados produzidos pelo sistema e os resultados sugeridos por usuários ou especialistas diante das mesmas consultas. Assim, o desempenho de vários sistemas é testado diante dos mesmos acervos e das mesmas consultas, e eles podem ser comparados estatisticamente utilizando as métricas de precisão e cobertura [Baeza-Yates and Ribeiro-Neto 2011, Manning et al. 2009]. Como os julgamentos para este *dataset* são parciais, a avaliação não é feita considerando simplesmente os resultados de cada métrica para comparar diferentes algoritmos. Então, a avaliação comparativa é feita levando-se em consideração os valores de precisão por cobertura [Baeza-Yates and Ribeiro-Neto 2011, Ruthven and Lalmas 2003].

6. Experimento

A Figura 1 ilustra as etapas executadas para a realização do experimento, desde a criação do conjunto de documentos que formam o acervo até o cálculo dos valores de precisão média.

O conjunto de documentos *ClueWeb09* possui 1.040.809.705 de documentos (páginas *web*) e esta quantidade torna o processo de indexação muito custoso trazendo poucas vantagens para a análise comparativa dos algoritmos de recuperação de informação. As poucas consultas e julgamentos disponibilizados pelo *TREC NIST* acabam cobrindo somente uma parte muito pequena desse conjunto, por isso, o experimento é realizado sobre um subconjunto do *dataset*. O subconjunto utilizado no experimento, é formado pelos primeiros 1.000 documentos devolvidos por um mecanismo de busca simples (VSM) diante de cada uma das 100 consultas disponibilizadas pelo *TREC NIST*, mais todos os documentos que foram julgados por um especialista, totalizando 101.479 documentos.

Etapas do experimento realizado

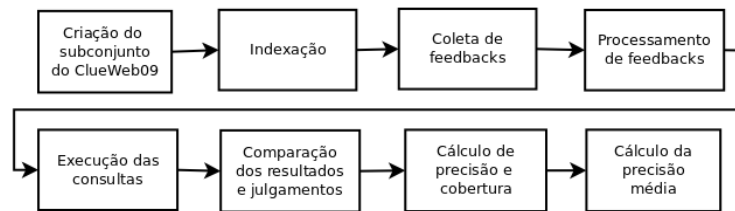


Figura 1. Fluxograma das etapas executadas para a realização do experimento proposto. Fonte: elaborado pelos autores.

A etapa de indexação é realizada da mesma maneira que o VSM padrão, com a criação do índice invertido, construção dos vetores de documentos e cálculo dos valores de *tf-idf*. Durante essa etapa, já são criados também os vetores de termos. As etapas de coleta e processamento de *feedbacks* incluem o processo de agrupar os documentos considerados relevantes para cada consulta e realizar o processamento deles seguindo o algoritmo de *feedback* de relevância orientado a termos. As informações obtidas através dos *feedbacks* não são utilizadas pelo modelo VSM.

Na etapa seguinte, as 100 consultas disponíveis são executadas utilizando tanto o algoritmo de *feedback* orientado a termos quanto o VSM. Os resultados desses dois algoritmos para cada uma das consultas formam uma lista ordenada de documentos de acordo com a relevância percebida pelo algoritmo. A etapa seguinte é a comparação desta lista de resultados com os julgamentos humanos, que gera uma lista ordenada de “acertos” (caso o documento da lista de resultados é considerado relevante de acordo com os julgamentos humanos) e “erros” (caso contrário).

Algoritmo 1: Cálculo de precisão por cobertura.

início

variável total = quantidade de documentos relevantes para a consulta;

variável contador = 0;

variável contador_acertos = 0;

repita

leia a próxima entrada da lista;

contador = contador + 1;

if “acerto” then

contador_acertos = contador_acertos + 1;

precisão = contador_acertos/contador;

cobertura = contador_acertos/total;

armazenar(precisão, cobertura);

até fim da lista de erros e acertos;

fin

Esta lista de erros e acertos é usada para o cálculo dos valores de precisão por cobertura. Este cálculo é realizado seguindo o Algoritmo 1 para cada uma das 100 con-

sultas realizadas, que resulta em 100 listas que possuem valores de precisão por nível de cobertura para cada um dos algoritmos testados.

Depois da execução do Algoritmo 1 para todas as consultas, são geradas duas listas com valores de precisão média por nível de cobertura (uma lista para cada algoritmo testado), sendo que os valores de precisão média são calculada a partir dos valores de precisão resultantes do Algoritmo 1.

7. Resultados obtidos

O gráfico da Figura 2 mostra os resultados do experimento, sendo eles, os valores de precisão média por nível de cobertura.

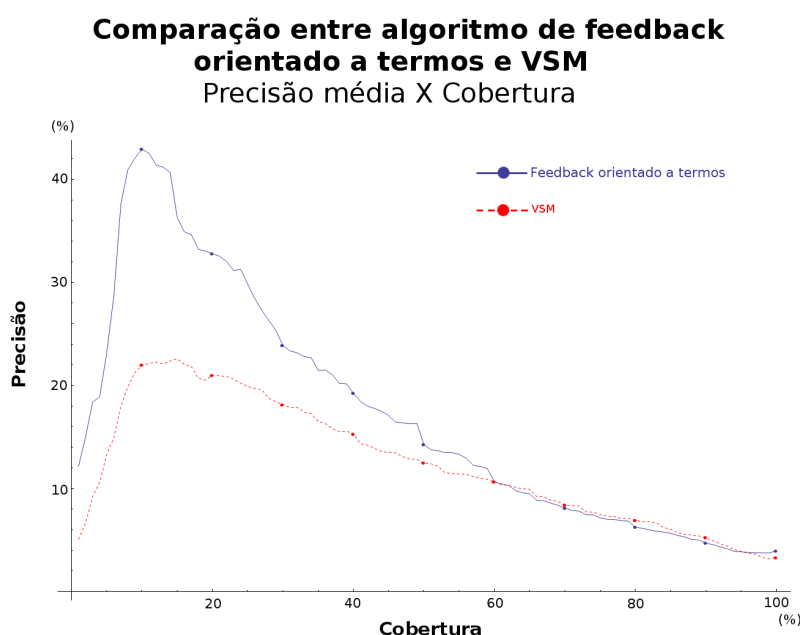


Figura 2. Gráfico comparativo entre o algoritmo de *feedback* orientado a termos e VSM em relação aos valores de precisão média por cobertura. Fonte: elaborado pelos autores.

O algoritmo que utiliza os *feedbacks* de relevância orientados aos termos apresenta uma melhora significativa para os valores de níveis de cobertura de até 40%. Esta porção inicial dos resultados da busca é a mais relevante quando comparando diferentes modelos de recuperação de informação, porque a maioria dos usuários normalmente se concentram nos primeiros resultados e ignoram o restante dos resultados [Pass et al. 2006].

Tabela 1. Valores de precisão por cobertura (até 40%).

Algoritmos	10%	20%	30%	40%
FROT	0,4294	0,3280	0,2390	0,1923
VSM	0,2196	0,2099	0,1815	0,1530

A Tabela 1 permite observar mais precisamente alguns os valores de precisão média por nível de cobertura (até 40%) obtidos pelos algoritmo de Feedback de Relevância Orientado a Termos (FROT) e VSM padrão.

8. Conclusão

A Figura 2 e a Tabela 1 evidenciam a capacidade do algoritmo de Feedback de Relevância Orientado a Termos em melhorar a qualidade dos resultados de uma busca em relação ao modelo mais amplamente utilizado (VSM), utilizando *feedbacks* de relevância orientado a termos. Quando o nível de cobertura atinge o valor de 10%, onde se concentram a maioria dos acessos e visualizações dos usuários de acordo com Pass, Chowdhury e Torgeson [Pass et al. 2006], a melhoria é de 95% em relação ao VSM padrão.

A utilização de *feedbacks* de relevância na alteração dos vetores que representam os termos do vocabulário pode melhorar a qualidade dos resultados obtidos. A estratégia de uso dos *feedbacks* permite que as informações obtidas dos usuários sejam armazenadas permanentemente e possam ser utilizadas em sessões de outros usuários, além de serem globais, ou seja, os *feedbacks* influenciam o cálculo de similaridade para todo o acervo e não somente para os documentos julgados pelos usuários.

Os trabalhos futuros incluem a investigação de modos alternativos para a realização dos cálculos de *tf-idf* e das transformações sobre os vetores de termos (incluindo a adaptação da equação de Brauen [Brauen et al. 1968]). Além da análise de modelos alternativos para a representação da relação entre termos, como por exemplo o uso do GVSM [Wong et al. 1985].

Referências

- Baeza-Yates, R. and Ribeiro-Neto, B. (2011). *Modern Information Retrieval*. Addison-Wesley Publishing Company, USA, 2nd edition.
- Belew, R. K. (1989). Adaptive information retrieval: Using a connectionist representation to retrieve and learn about documents. *SIGIR Forum*, 23(SI):11–20.
- Bodoff, D., Enache, D., Kambil, A., Simon, G., and Yukhimets, A. (2001). A unified maximum likelihood approach to document retrieval. *Journal of the American Society for Information Science and Technology*, 52(10):785–796.
- Bot, R. S. and Wu, Y.-f. B. (2004). Improving document representations using relevance feedback: The rfa algorithm. In *Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management, CIKM '04*, pages 270–278, New York, NY, USA. ACM.
- Brauen, T. L., Holt, R. C., and Wilcox, T. R. (1968). Document indexing based on relevance feedback. Report isr-14 to the national science foundation, section xi, Department of Computer Science, Cornell University, Ithaca, NY.
- Buckley, C., Lease, M., Smucker, M. D., Jung, H. J., Grady, C., Buckley, C., Lease, M., Smucker, M. D., Grady, C., Lease, M., et al. (2010). Overview of the trec 2010 relevance feedback track (notebook). In *The Nineteenth Text Retrieval Conference (TREC) Notebook*.
- Buckley, C., Salton, G., and Allan, J. (1994). The effect of adding relevance information in a relevance feedback environment. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '94*, pages 292–300, New York, NY, USA. Springer-Verlag New York, Inc.
- Callan, J., Hoy, M., Yoo, C., and Zhao, L. (2009). Clueweb09 data set.

- Carpineto, C. and Romano, G. (2012). A survey of automatic query expansion in information retrieval. *ACM Comput. Surv.*, 44(1):1:1–1:50.
- Fuhr, N. and Buckley, C. (1990). Probabilistic document indexing from relevance feedback data. In *Proceedings of the 13th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '90, pages 45–61, New York, NY, USA. ACM.
- Limbu, D. K., Connor, A., Pears, R., and MacDonell, S. (2006). Contextual relevance feedback in web information retrieval. In *Proceedings of the 1st International Conference on Information Interaction in Context*, IiX, pages 138–143, New York, NY, USA. ACM.
- Ma, Y. and Lin, H. (2014). A Multiple Relevance Feedback Strategy with Positive and Negative Models. *PLOS ONE*, 9(8).
- Manning, C. D., Raghavan, P., and Schütze, H. (2009). *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA.
- Pass, G., Chowdhury, A., and Torgeson, C. (2006). A picture of search. In *Proceedings of the 1st International Conference on Scalable Information Systems*, InfoScale '06, New York, NY, USA. ACM.
- Rocchio, J. J. (1971). *Relevance feedback in information retrieval*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.
- Ruthven, I. and Lalmas, M. (2003). A survey on the use of relevance feedback for information access systems. *Knowl. Eng. Rev.*, 18(2):95–145.
- Salton, G. (1989). *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.
- Salton, G., Wong, A., and Yang, C. S. (1975). A vector space model for automatic indexing. *Commun. ACM*, 18(11):613–620.
- Schwartz, C. (2000). Digital libraries: an overview. *The Journal of Academic Librarianship*, 26(6):385–393.
- Wong, S. K. M., Ziarko, W., and Wong, P. C. N. (1985). Generalized vector spaces model in information retrieval. In *Proceedings of the 8th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '85, pages 18–25, New York, NY, USA. ACM.
- Zhang, J. and Kamps, J. (2010). A search log-based approach to evaluation. In Lalmas, M., Jose, J., Rauber, A., Sebastiani, F., and Frommholz, I., editors, *Research and Advanced Technology for Digital Libraries*, volume 6273 of *Lecture Notes in Computer Science*, pages 248–260. Springer Berlin Heidelberg.

XConversor: Uma Ferramenta de Conversão de Artigos do Formato HTML para XML Aplicada ao Domínio Médico

Arthur E. de O. Carosia¹, Diego de Castro Cunha, Cristina Dutra de Aguiar Ciferri²

¹Instituto Federal de São Paulo (IFSP)
São João da Boa Vista, SP, Brasil

²Universidade de São Paulo
São Carlos, SP, Brasil

arthuremanuel.carosia@ifsp.edu.br, diego.aax@gmail.com, cdac@icmc.usp.br

Abstract. *Medical articles are available in online repositories with millions of medical publications. From the immense amount of information available, humans are not able to read and assimilate all this information. This paper presents the XConversor tool, which performs the conversion of the medical articles from HTML to XML in order to structure the article, aiming in the Text Mining processes or export to a database. The generated XML file has tags that identify various elements of interest in a medical article: authors, sections, paragraphs, sentences and pages.*

Resumo. *Artigos científicos da área médica estão disponíveis em repositórios on-line, que possuem milhões de publicações. A partir da imensa quantidade de informação disponível, os seres humanos não são capazes de ler e assimilar toda essa informação, sendo a Mineração de Textos uma atividade que favorece o processamento automático de textos. Este artigo apresenta a ferramenta XConversor, que converte de artigos da área médica do formato HTML para o formato XML, estruturando o artigo e auxiliando nos processos de Mineração de Textos ou de povoamento de banco de dados. O arquivo XML gerado possui etiquetas que identificam diversos elementos de interesse em um artigo da área médica: autores, seções, parágrafos, sentenças e páginas.*

1. Introdução

Artigos científicos da área médica estão disponíveis em repositórios *on-line* como *PubMed* (<http://www.ncbi.nlm.nih.gov/pubmed>), que possui mais de 18 milhões de publicações médicas. O conhecimento armazenado nesses artigos consiste em uma importante fonte de informação, que pode ser explorada visando auxiliar o desenvolvimento de novas pesquisas ou ainda poder proporcionar uma possível adequação das experiências realizadas ao cenário brasileiro. Nesse contexto, a Mineração de Textos [Tan 1999] é uma tendência que favorece o processamento de textos, viabilizando o acesso a uma grande quantidade de dados e o objetivo de extração de conhecimento. No entanto, as tarefas de Mineração de Textos são realizadas predominantemente sobre documentos não-estruturados, como páginas HTML [HTML 2015] e documentos textuais [Aranha and Passos 2006]. Além disso, trabalhos como o de [Corney et al. 2004] destacam o benefício em extrair informação em artigos completos, afinal mais da metade da informação extraída é do corpo do artigo.

Um dos problemas desta abordagem é a localização das informações textuais extraídas nos documentos originais. Por exemplo, um especialista da área médica pode julgar adequado examinar um determinado parágrafo relacionado a uma sentença extraída automaticamente em busca de maiores esclarecimentos em relação a um experimento. Neste sentido, um desafio enfrentado por sistemas de Mineração de Textos é encontrar uma fonte de informações na qual seja possível o acesso posterior aos níveis estruturais dos artigos como, por exemplo, número de página, parágrafo ou sentença.

Dessa forma, este artigo introduz a ferramenta *XConversor* (XML Conversor). O objetivo da ferramenta *XConversor* é converter artigos científicos da área médica do formato HTML, que é o formato predominante dos artigos médicos disponíveis, para o formato XML [XML 2015], apresentando etiquetas que identificam porções relevantes à estrutura do artigo. Dessa forma, o artigo no formato *XML* estaria preparado para posteriores atividades de Mineração de Textos. Além disso, o documento convertido no formato *XML* pode ser utilizado também no povoamento de banco de dados. Portanto, este documento convertido deve ser legível e deve ser composto por marcadores *XML* estruturados e posicionados adequadamente no documento para, por exemplo, separar diferentes conteúdos do arquivo original e evidenciar partes relevantes no texto. Por fim, o documento *XML* deve manter uma correspondência com o artigo original, permitindo inclusive identificar a frase do parágrafo e a página que contém um dado de interesse.

Este artigo é estruturado como descrito a seguir. A Seção 2 descreve a ferramenta proposta *XConversor*, enquanto que a Seção 3 ilustra exemplos de conversão de artigos da área médica do formato *HTML* para o formato *XML*. A Seção 4 descreve trabalhos correlatos. Por fim, a Seção 5 conclui este artigo.

2. A Ferramenta *XConversor*

O formato dos dados textuais pode ser estruturado, semi-estruturado ou não-estruturado. Segundo o trabalho [Elmasri and Navathe 1999], as informações armazenadas em banco de dados são dados estruturados, documentos XML são dados semi-estruturados, enquanto as páginas da Web em formato HTML são não-estruturados.

A maior parte dos artigos disponíveis em *PubMed* são obtidos no formato *HTML*. Assim, a função da ferramenta *XConversor* é estruturar no formato *XML* cada artigo convertido com etiquetas que identificam as seguintes informações: seções, subseções, números de páginas, parágrafos, frases, nomes dos autores, nome da revista e título do artigo. O documento convertido no formato XML pode ser posteriormente utilizado em atividades de Mineração de Textos ou para promover o povoamento de um banco de dados relacional.

Esta seção está organizada como descrito a seguir. A Seção 2.1 apresenta a arquitetura da ferramenta *XConversor* e seus módulos principais, enquanto que a Seção 2.2 ilustra a interface da ferramenta. A Seção 2.3 apresenta as heurísticas utilizadas para identificar as etiquetas *XML* em um artigo. Por fim, a Seção 2.4 detalha o uso de tradutores na conversão realizada pela ferramenta *XConversor*.

2.1. Arquitetura

A arquitetura da ferramenta *XConversor* possui os seguintes módulos, ilustrados na Figura 1:

- **Interface principal:** mostra visualmente as opções oferecidas pela ferramenta para converter artigos, editar artigos e alterar configurações.
- **Conversor de artigos:** consiste no núcleo da ferramenta e, portanto, tem como funcionalidade fazer a conversão de documentos do formato *HTML* para o formato *XML* por meio de scripts elaborados na linguagem *Perl* [Perl 2015]. Desde que os artigos *HTML* a serem convertidos são oriundos de diferentes revistas da área médica, as quais possuem características de formatação heterogêneas, há diversos tradutores (*Tradutor 0*, *Tradutor 1*, ..., *Tradutor n*), os quais são específicos para cada formato de artigo e têm por objetivo adaptar os formatos heterogêneos desses artigos para o formato provido pelo processador de conversão. A definição de tradutores garante flexibilidade à ferramenta *XConversor*, permitindo que a ferramenta converta artigos médicos de diferentes formatos, criando-se novos tradutores quando necessário.
- **Editor de artigos:** exibe artigos convertidos no formato *XML* e oferece ao usuário opções de edição desses artigos, de comparação com os artigos originais no formato *HTML* e de validação dos artigos *XML* convertidos.
- **Configurações:** permite a definição das configurações para o funcionamento da ferramenta, tais como a configuração de pastas de artigos a serem convertidos (i.e., arquivos no formato *HTML*) e de artigos já convertidos (i.e., arquivos no formato *XML*).

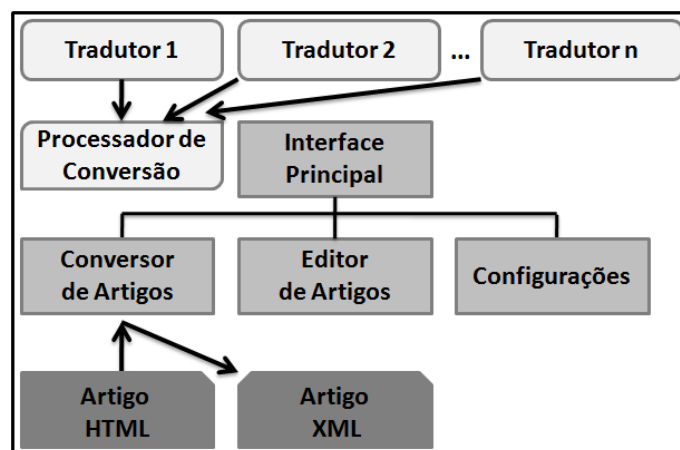


Figura 1. Arquitetura da Ferramenta *XConversor*

A conversão de artigos na ferramenta *XConversor* é ilustrada na Figura 2 e feita da seguinte forma. O documento *HTML* é alterado para incluir alguns marcadores *XML* propostos, como aqueles que identificam o nome da revista, o título e os subtítulos do artigo (Figura 2 a). Em seguida, há a remoção das tags *HTML* presentes no documento e a inclusão dos marcadores *XML* propostos para a identificação de parágrafos (Figura 2 b), gerando um documento no formato *TXT* que contém todas as marcações dos elementos já identificados. Por fim, são inseridos no documento *TXT* os marcadores *XML* propostos para identificação de sentenças, números de página e autores, gerando o documento *XML* convertido final (Figura 2 c).

A estrutura geral de um arquivo *XML* gerado pela ferramenta é mostrado na Figura 3. Ao fim do processamento, os seguintes elementos são identificados em um arquivo *XML* convertido: nome da revista de publicação do artigo (*magazine*), título do artigo (*title*), autores (*author*), parágrafos *paragraph*, páginas (*page*), sentenças (*sentence*).

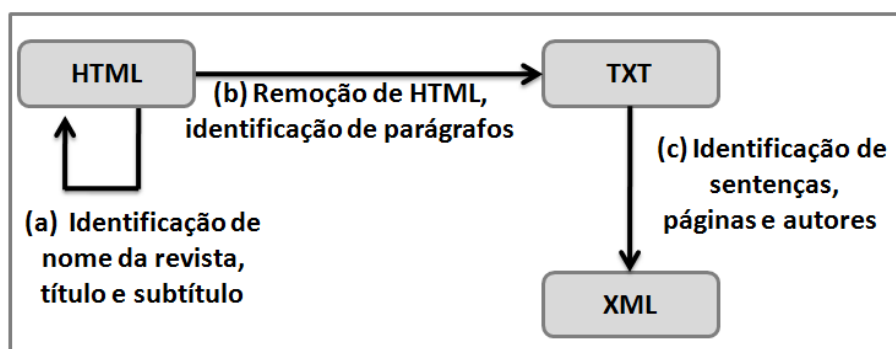


Figura 2. Processo de Conversão Realizado pela Ferramenta *XConverter*

```

<document>
  <magazine></magazine>
  <title></title>
  <author></author>
  <paragraph page="" number="">
    <sentence number=""></sentence>
  </paragraph>
</document>
  
```

Figura 3. Estrutura XML Gerada pela Ferramenta *XConverter*

2.2. Interface com o Usuário

A ferramenta *XConverter* possui um módulo de interface com o usuário especialista. Nela é possível acompanhar todo o processo de conversão e inclusive editar o arquivo *XML* convertido.

Para a identificação de alguns termos no arquivo *XML* resultante, a ferramenta *XConverter* necessita de algumas informações fornecidas pelo usuário na aba *HTML-XML Converter*, ilustrada na Figura 4. Esta aba apresenta em seu canto superior esquerdo o nome do artigo no formato *HTML* a ser carregado, em seu canto inferior esquerdo apresenta o artigo carregado, bem como a caixa de seleção para o nome do journal que este artigo pertence. Nesta versão da ferramenta está presente a capacidade de converter artigos das fontes *Blood Journal* [Blood Journal 2015] e *New England Journal of Medicine* [N. E. J. Medicine 2015], que possuem uma vasta gama de artigos da área médica. Já no canto superior direito, esta aba apresenta informações necessárias para a conversão do artigo: primeiro autor, último autor e ano de publicação do artigo. Estas informações também podem ser carregadas a partir de um módulo que retira estes dados de um arquivo *BibTex*, ao clicar no botão *Load BibTex* no canto superior direito. Ainda, estas mesmas informações podem ser retiradas do próprio artigo, ao selecionar estes itens no artigo exibido no canto inferior esquerdo e clicando no botão correspondente para armazenar as informações (o botão *First Author* refere-se a primeiro autor, o botão *Last Author* refere-se a último autor e o botão *Year* refere-se a ano). Por fim, a parte inferior direita apresenta a escolha de se o artigo apresenta ou não uma primeira página a ser removida na conversão (em caso de ser capa do artigo e não ter dados relevantes) e também botões que permitem a conversão imediata do artigo (*Convert Article*) ou que o adicionará para conversão em lote (*Add Lote*), que se trata da conversão de vários artigos simultaneamente.

A Figura 5 ilustra a interface da aba *Editor*. Nela, pode-se visualizar o documento

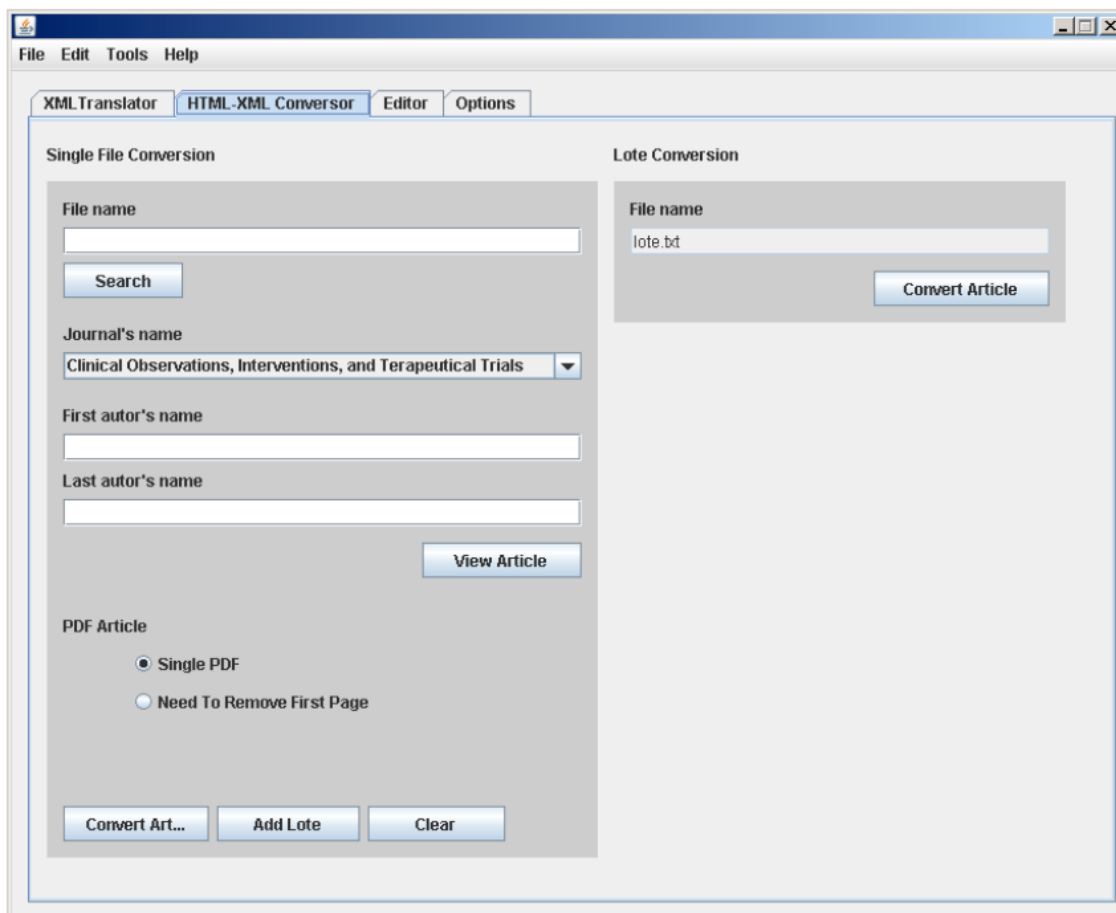


Figura 4. Interface de Conversão da Ferramenta XConversor

HTML à esquerda e o documento *XML* convertido pelo processador de conversão à direita. Essa aba ainda mostra lado a lado os documentos e assim facilita a comparação entre ambos e a visualização de inconsistências. Além disso, há uma barra de ferramentas na parte superior com opções de edição (novo, salvar, abrir, fechar, desfazer e refazer). Na lado de visualização do documento *XML*, no canto superior direito, existem as opções de: (i) inserir novas tags, para o caso em que o usuário deseje alterar o documento gerado caso esse apresente alguma incorreção na conversão; e (ii) alterar o tamanho da fonte para facilitar a visualização do texto.

A interface da ferramenta *XConversor* está implementada na linguagem de programação *Java* [Java 2015].

2.3. Heurísticas de Conversão

De acordo com a proposta de marcadores *XML* descrita na Seção 2.1, os seguintes elementos relevantes devem ser identificados em um artigo: revista que publicou o artigo, título do artigo, nomes dos autores, parágrafos e sentenças, além de números de páginas, e subtítulos. Para cada um destes elementos, o processador de conversão define uma heurística específica, a qual é definida com base no formato padrão. Cada uma dessas heurística é descrita a seguir.

- **Revista:** identificado de acordo com o fornecimento dessa informação pelo usuário. Existe uma lista de revistas possíveis de serem convertidas e o usuário

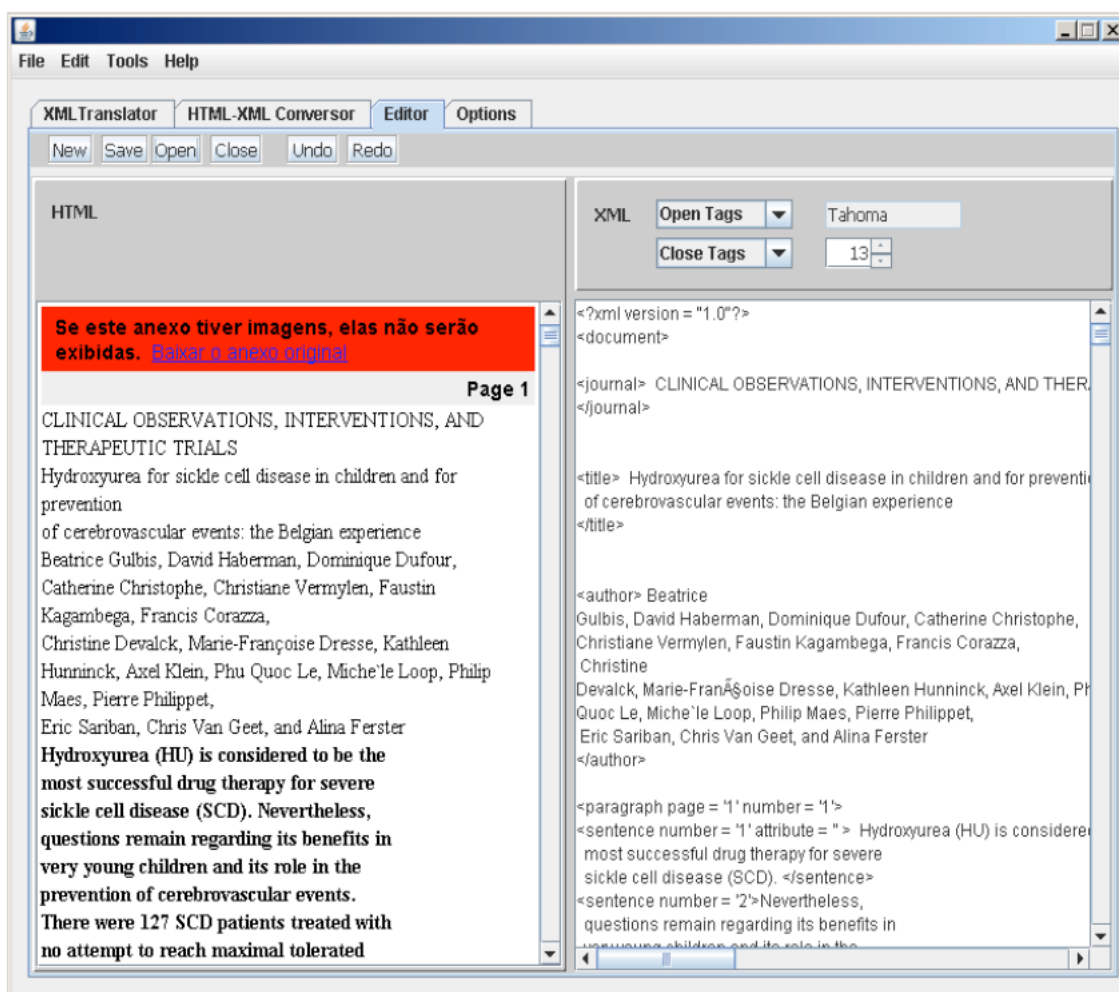


Figura 5. Interface de Edição da Ferramenta XConverter

deve indicar qual delas é a que se deseja converter. Uma vez fornecido o nome da revista, ele é identificado por meio da busca desse nome no artigo.

- **Autor:** identificado de acordo com o fornecimento dos nomes dos primeiro e último autores pelo usuário. Ele também é identificado por meio da busca desses nomes no artigo.
- **Título:** identificado tendo como base a maior fonte do texto. Para tanto, usa-se o atributo *font-size* da etiqueta *span* do documento *HTML*.
- **Parágrafo:** identificado levando-se em consideração a análise do atributo *left*, da etiqueta *div*, que indica o posicionamento horizontal do texto no documento *HTML* em relação à margem. Se a linha atual possui valor *left* maior que o anterior e o posterior, então o próximo texto é definido dentro de um novo parágrafo. Esse é um dos elementos mais dependentes da estrutura do documento *HTML*.
- **Subtítulo:** identificado com a seguinte regra: deve estar em negrito (identificado por meio do uso da etiqueta *b*), não deve conter ponto final e deve ocupar apenas uma linha. Essa última condição foi acrescida para que um subtítulo não seja confundido com o resumo do artigo, o qual na área médica é geralmente escrito em negrito e possui várias linhas.
- **Número de página:** obtido por meio do próprio documento *HTML*. O número de cada página do artigo é marcado com uma tag *HTML*.

2.4. Tradutores

A ferramenta *XConversor* adotada um formato padrão de artigos que podem ser convertidos. Entretanto, revistas da área médica possuem, em geral, características de formatação heterogêneas. Assim, seus artigos podem possuir um formato muito diferente do formato padrão de conversão adotado no processador de conversão, impossibilitando que eles sejam convertidos corretamente pela ferramenta. Para resolver esse problema, a ferramenta *XConversor* é baseada no uso de tradutores.

Um tradutor é um *script* implementado na linguagem de programação *Perl* que tem a funcionalidade de adaptar o formato de um artigo para o formato padrão implementado na ferramenta. Para cada revista que contém artigos médicos de interesse, é possível criar um tradutor específico, baseado nas características de formatação dessa revista.

A definição de tradutores garante flexibilidade à ferramenta *XConversor*, permitindo que a ferramenta converta arquivos médicos de diferentes formatos, criando-se novos tradutores quando necessário. Foram desenvolvidos até o momento dois tradutores, cada um relativo a uma determinada revista. Cada tradutor possui diferenças com relação ao formato padrão implementado na ferramenta. As revistas consideradas pelos tradutores, que possuem suas características adaptadas às funcionalidades da ferramenta são: *Blood Journal* e *New England Journal of Medicine*.

3. Conversão de Artigos com a ferramenta *XConversor*

Para validar a conversão provida pela ferramenta *XConversor*, foram convertidos até o momento 20 artigos da área médica, pertencentes às revistas *Blood Journal* e *New England Journal of Medicine*, que são revistas que possuem tradutores implementados na ferramenta.

Todos os documentos *XML* gerados pela ferramenta mostraram-se consistentes com o documento original, ou seja, o documento *XML* convertido especificou, corretamente, o nome da revista, o título do artigo, os autores, as seções, os parágrafos, as sentenças e as páginas nas quais esses parágrafos aparecem. Isso mostra que, para qualquer artigo que siga o formato dessas revistas, a conversão é realizada com sucesso.

Como exemplo, a Figura 6 demonstra um artigo no seu formato *HTML* e a Figura 7 demonstra o mesmo artigo convertido para o formato *XML*. Note as características do artigo no formato *HTML*: título com fonte maior que as demais, lista de autores e texto em várias colunas. Tais elementos são determinantes para a construção dos tradutores descritos na Seção 2.4. Cada formato de artigo que possua características diferentes deverá ter seu próprio tradutor desenvolvido.

Outro ponto a se considerar é que o arquivo resultante *XML* procura atender fielmente a cada um dos elementos do artigo original. Dessa forma, o especialista sempre possuirá a vantagem de poder retornar ao artigo original para mais detalhes das informações extraídas.

4. Trabalho Correlatos

A ferramenta correlata, chamada de GetItFull [Natarajan et al. 2006], realiza o download de artigos científicos no formato *HTML* e os processa gerando um arquivo no formato *XML* com etiquetas identificando cada seção do artigo. Assim, os arquivos *XML* gerados

Hydroxyurea for sickle cell disease in children and for prevention of cerebrovascular events: the Belgian experience

Béatrice Gulbis, David Haberman, Dominique Dufour, Catherine Christophe, Christiane Vermeylen, Faustin Kagambega, Francis Corazza, Christine Devalck, Marie-Françoise Dresse, Kathleen Hunninch, Axel Klein, Phu Quoc Le, Michèle Loop, Philip Maes, Pierre Philippet, Eric Sariban, Chris Van Geet, and Alina Ferster

Hydroxyurea (HU) is considered to be the most successful drug therapy for severe sickle cell disease (SCD). Nevertheless, questions remain regarding its benefits in very young children and its role in the prevention of cerebrovascular events.

There were 127 SCD patients treated with no attempt to reach maximal tolerated doses who entered the Belgian Registry: 109 for standard criteria and 18 who were at risk of stroke only. During 426 patient-years of follow-up for patients with stan-

dard criteria, 3.3 acute chest syndromes, 1.3 cerebrovascular events, and 1.1 osteonecrosis per 100 patient-years were observed. A subgroup of 32 patients followed for 6 years experienced significant benefit over this period. In each subgroup of children (younger than 2 years, 2-5, 6-9, and 10-19 years) followed for 2 years, clinical and biologic changes were similar, except for children younger than 2 years who had no total hemoglobin increase and remained at risk of severe

anemia. In 72 patients evaluated by transcranial Doppler studies (TCD), 34 patients were at risk of primary stroke and only 1 had a cerebrovascular event after a follow-up of 96 patient-years. These results confirm the benefit of HU, even in very young children, and its possible role in primary stroke prevention. (Blood. 2005; 105:2685-2690)

© 2005 by The American Society of Hematology

Figura 6. Artigo HTML para Conversão Por Meio da Ferramenta XConversor

podem ser utilizados em aplicações de mineração de texto ou serem exportados para um banco de dados.

O processo elaborado para a ferramenta *GetItFull* é descrito como segue. O primeiro módulo realiza o *download* de documentos. O segundo módulo processa artigos no formato *HTML*, removendo caracteres especiais, e etiquetas *HTML*, inserindo etiquetas de seção, e gerando o documento no formato *XML*. O formato do arquivo *XML* gerado pela ferramenta *GetItFull* contém as etiquetas *title* (título), *abstract* (resumo), *introduction* (introdução), *methods* (métodos), *results* (resultados) e *discussion* (discussão).

Como pode ser observado, a ferramenta *GetItFull* apresenta semelhanças com a ferramenta *XConversor*, principalmente quanto ao processo de conversão. No entanto, as ferramentas diferem em vários pontos. O primeiro ponto é o seguinte: como o *XML* proposto para a ferramenta *GetItFull* já inclui nomes das seções, ela se torna limitada a artigos que possuam todas as seções (i.e., título, resumo, introdução, métodos, resultados e discussão), ao contrário da ferramenta *XConversor*, que usa regras para encontrar as seções e seus nomes, que podem ser variados. Além disso, a ferramenta *XConversor* é mais flexível e pode oferecer suporte a uma maior quantidade de documentos convertidos, desde que novos tradutores podem ser facilmente incorporados a ela.

Outra diferença refere-se ao fato de que o documento *XML* gerado pela ferramenta *GetItFull* não incorpora vários marcadores que a ferramenta *XConversor* introduz: autor, frase, parágrafo e página em uma seção. Com isto, a ferramenta *GetItFull* apresenta um nível de detalhamento menor em relação à ferramenta *XConversor*, que ao introduzir esses marcadores adicionais garante a possibilidade de encontrar no artigo original a posição exata em que um trecho de texto aparece. Além disso, em algum caso de incorreção na conversão, a ferramenta *XConversor* possui o módulo editor, capaz de ajudar ao usuário a identificar e corrigir o erro, comparando o artigo convertido ao original. Por fim, a ferramenta *GetItFull* não oferece funcionalidades relacionadas à interação com o usuário.


```

<?xml version="1.0" ?>
<document>
  <magazine>CLINICAL OBSERVATIONS, INTERVENTIONS, AND THERAPEUTIC
    TRIALS CLINICAL OBSERVATIONS, INTERVENTIONS, AND THERAPEUTIC
    TRIALS</magazine>
  <title>Hydroxyurea for sickle cell disease in children and for prevention of
    cerebrovascular events: the Belgian experience</title>
  <autor>Beatrice Gulbis, David Haberman, Dominique Dufour, Catherine
    Christophe, Christiane Vermylen, Faustin Kagambega, Francis Corazza,
    Christine Devalck, Marie-Françoise Dresse, Kathleen Hunninck, Axel Klein,
    Phu Quoc Le, Miche`le Loop, Philip Maes, Pierre Philippet, Eric Sariban,
    Chris Van Geet, and Alina Ferster</autor>
  <paragraph page="1" paragraph="1">
    <sentence number="1" attribute="">Hydroxyurea (HU) is considered to
      be the most successful drug therapy for severe sickle cell disease
      (SCD).</ sentence >
    < sentence number="2">Nevertheless, questions remain regarding its
      benefits in very young children and its role in the prevention of
      cerebrovascular events.</ sentence >

```

Figura 7. Artigo XML Convertido pela Ferramenta XConversor

5. Conclusão

Este artigo apresentou a ferramenta *XConversor*, cujo objetivo é converter artigos da área médica do formato *HTML* para o formato *XML* com o objetivo de estruturar artigos para posterior Mineração de Textos ou armazenagem em um banco de dados relacional. Essa ferramenta foi descrita em termos: (i) dos marcadores por ela utilizados; (ii) da proposta de sua arquitetura; (iii) de sua interface com o especialista; (iv) do detalhamento de seus módulos voltados à conversão e à tradução de artigos de diferentes revistas.

A ferramenta *XConversor* foi testada com artigos médicos das revistas *Blood Journal* e *New England Journal of Medicine* e apresentou resultados de conversões satisfatórios, ou seja, o documento XML convertido especificou, corretamente, o nome da revista, o título do artigo, os autores, as seções, os parágrafos, as sentenças e as páginas nas quais esses parágrafos aparecem.

Além disso, uma comparação com o trabalho relacionado *GetItFull* foi realizada, sendo apresentados os pontos em que ambas ferramentas se diferem e também aqueles pontos em que a ferramenta *XConversor* supera as limitações da ferramenta correlata.

Trabalhos futuros incluem o desenvolvimento de um módulo de conversão de artigos que estão do formato *PDF* para *XML*, além de ampliar o número de revistas capazes de serem convertidas pela ferramenta e aplicar a ferramenta para artigos científicos pertencentes a outros domínios.

Referências

- Aranha, C. N. and Passos, E. P. L. (2006). A tecnologia de mineração de textos. *Revista Eletrônica de Sistemas de Informação (RESI)*, 8(2):1–8.
- Blood Journal (2015). Blood Journal. <http://bloodjournal.hematologylibrary.org>. Último acesso em 19-04-2015.

- Corney, D. P. A., Buxton, B. F., Langdon, W. B., and Jones, D. T. (2004). Biorat: extracting biological information from full-length papers. *Bioinformatics*, 20(17):3206–3213.
- Elmasri, R. A. and Navathe, S. B. (1999). *Fundamentals of Database Systems*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 3rd edition.
- HTML (2015). HTML - Hipertext Markup Language. <http://www.w3.org/HTML/>. Ultimo acesso em 19-04-2015.
- Java (2015). Java Programming Language. <http://www.java.com/>. Ultimo acesso em 19-04-2015.
- N. E. J. Medicine (2015). New England Journal of Medicine. <http://www.nejm.org/>. Ultimo acesso em 19-04-2015.
- Natarajan, J., Haines, C., Berglund, B., DeSesa, C., Hack, C. J., Dubitzky, W., and Bremer, E. G. (2006). Getitfull - A tool for downloading and pre-processing full-text journal articles. In *Knowledge Discovery in Life Science Literature, PAKDD 2006 International Workshop, KDLL 2006, Singapore, April 9, 2006, Proceedings*, pages 139–145.
- Perl (2015). The Perl Programming Language. <http://www.perl.org/>. Ultimo acesso em 19-04-2015.
- Tan, A. H. (1999). Text mining: the state of the art and the challenges. *KDAD, Beijing, China, PAKDD*, pages 71–76.
- XML (2015). XML - Extensible Markup Language. <http://www.w3.org/XML/>. Ultimo acesso em 19-04-2015.

Sistema para Auxiliar Administração de Medicamentos em Ambientes de Vivência Assistida

Milene Santos Teixeira¹, Leandro Oliveira Freitas¹, Un Hee Schiefelbein¹, Juçara Salete Gubiani¹

¹Curso de Sistemas para Internet – Universidade Federal de Santa Maria (UFSM)
Santa Maria – RS - Brasil

{milene.tsi,peace.unhee,jsgubiani}@gmail.com,
leandro@politecnico.ufsm.br

Abstract. *Due to the large increase in the elderly population occurred in recent years, new concerns about their health and well being are raised. In this context, comes up the Ambient Assisted Living, an Ambient Intelligence's field that makes use of pervasive and ubiquitous technology aiming to provide products and services, usually, to elderly. This paper proposes a pervasive and context sensitive system that aims to assist elderly in the use of medication. The system makes use of a physical device and alerts that help the elderly to use the correct medicine at the correct time. Furthermore, it is also able to alert a responsible under adverse conditions and to keep a track record of the use of medication. For a better definition of this proposal, works in this field were analyzed and, for its validation, a case study is presented. The system is in phase of development and it is intended to be deployed in real environments.*

Resumo. *Diante do grande aumento da população idosa ocorrido nos últimos anos, surgem novas preocupações com relação a saúde e bem estar destes. Neste contexto, destacam-se os Ambientes de Vivência Assistida, uma área de Ambientes Inteligentes que faz uso de tecnologia ubíqua ou pervasiva visando prover produtos e serviços, em geral, à idosos. Este trabalho propõe um sistema pervasivo sensível ao contexto que visa auxiliar idosos na administração da medicação. O sistema faz uso de um dispositivo físico e emite alertas que auxiliam o idoso a usar o medicamento correto no horário correto. Além disso, ele também é capaz de alertar um responsável em situações adversas e manter um registro de acompanhamento do uso da medicação. Para melhor definição da proposta foram analisados trabalhos realizados na área e para sua validação, um estudo de caso é apresentado. O sistema encontra-se em fase de desenvolvimento e visa-se implantá-lo em ambientes reais.*

1. Introdução

De acordo com dados do Instituto Brasileiro de Geografia e Estatística (IBGE), o número de pessoas idosas no Brasil vem crescendo consideravelmente. Estima-se que em 30 anos o número de idosos ultrapasse o número de crianças [IBGE 2008]. Em [ERCIM 2011] verifica-se que este fenômeno tem ocorrido de maneira similar em diversos outros países. Como consequência disto, surge uma nova demanda por produtos e serviços que proporcionem o bem estar deste grupo da população. Com o objetivo de suprir esta demanda, destacam-se pesquisas que envolvem Ambientes de

Vivência Assistida, uma área de Ambientes Inteligentes que foca no uso de tecnologia pervasiva ou ubíqua na residência de idosos [Aarts e Wichert 2009] e tem sido amplamente explorada na Europa [ERCIM 2011].

Considerando os conceitos destas tecnologias, este trabalho tem por objetivo propor um sistema para auxiliar na administração de medicamentos a pacientes que recebem acompanhamento clínico em Ambientes de Vivência Assistida. Este sistema deve trabalhar em conjunto com um dispositivo físico onde serão dispostos os medicamentos do paciente, sendo este capaz de: localizar o paciente na residência e alertá-lo para usar o medicamento correto; garantir que o medicamento correto foi utilizado na hora correta e, em caso de erro, alertar um responsável pelo paciente. Para validação da proposta, apresenta-se um estudo de caso englobando as funcionalidades do sistema.

Este trabalho possui a seguinte estrutura: primeiramente são apresentados os conceitos e diferenças de computação pervasiva e ubíqua; a seguir serão descritos conceitos de Ambientes Inteligentes e Ambientes de Vivência Assistida. Na sequência apresenta-se a proposta do trabalho seguida por um estudo de caso. Após isto, são citados alguns trabalhos publicados na área e uma breve conclusão resume o trabalho e fala sobre trabalhos futuros.

2. Computação Ubíqua e Pervasiva

A computação pervasiva pode ser descrita como uma tecnologia 'invisível' [Jones e Jo 2004]. De acordo com o previsto pelo cientista Mark Weiser [Weiser e Brown, 1996] na década de 90, esta invisibilidade ocorre devido ao fato da tecnologia passar a existir no ambiente a nossa volta de uma forma tão natural que, muitas vezes, nem a percebemos. Dessa forma, [Satyanarayana 2001] refere-se à computação pervasiva em um ambiente como uma 'tecnologia que desaparece', devido ao fato de ser altamente integrada ao usuário sem exigir muita interação com este. Em outras palavras, aplicações usando computação pervasiva precisam ser cada vez mais sensíveis ao contexto a fim de operar nos mais diferentes ambientes e reduzir a necessidade de atenção do usuário [Henricksen, Indulska e Rakotonirainy 2002].

O termo 'computação ubíqua' (UbiComp) foi criado pelo cientista Mark Weiser em 1991 [Weiser 1991]. Weiser previa que em um futuro breve, o usuário não precisaria mais focar no dispositivo usado para a execução de uma tarefa, sendo possível, muitas vezes, nem perceber que havia um dispositivo sendo utilizado [Weiser 1996]. Segundo Weiser, computação ubíqua seria um tipo de computação na qual não seria mais apenas um computador capaz de realizar diversas atividades, e sim diversos dispositivos embarcados conectados entre si capazes de serem usados no dia a dia das pessoas, tanto em seu trabalho, escritório, como em seus lares [Weiser 1991].

Em [Friedewald e Raabe 2011] destaca-se que o termo computação ubíqua é refletido em outros conceitos muito similares a este, como por exemplo, o conceito de 'Computação móvel' e 'Computação Pervasiva'. Entretanto, estes termos não possuem o mesmo significado. De acordo com [Lyytinen e Yoo 2002], o nível de embarcamento e de mobilidade é o que os diferencia. A computação móvel, por exemplo, apresenta um alto grau de mobilidade, porém um baixo grau de embarcamento. Sistemas tradicionais de computação se caracterizam por um baixo grau nestas duas categorias. Em síntese, sabendo que ambas computação pervasiva ou ubíqua apresentam um alto grau de

embarcamento, o que as difere é o grau de mobilidade que, em geral, é baixo na pervasiva, porém consideravelmente alto na ubíqua. O uso de sensores e microprocessadores integrados a ambientes, visando o bem estar e otimização de tarefas rotineiras, é o grande ponto em comum destas duas tecnologias [Friedewald e Raabe 2011].

3. Ambientes Inteligentes e Ambientes de Vivência Assistida

De acordo com [Alcaniz e Rey 2005], o objetivo de Ambientes Inteligentes (AmI) é ampliar a interação entre seres humanos e tecnologias da informação através do uso de dispositivos com características da computação pervasiva [ISTAG 2001]. Processar fatos para detectar futuras situações em um ambiente, prever situações indesejadas podendo agir antecipadamente a elas e ter ampla capacidade para manipular diferentes situações são apenas três características citadas por [Machado et al. 2014] das quais sistemas AmI devem apresentar. Diferentemente da computação convencional que se restringe ao uso de interfaces que exigem um contato direto do usuário, sistemas para AmI visam uma melhor utilização do ambiente que cerca o usuário. Para isto, a AmI utiliza sensores capazes de captar elementos como som, cheiro, formas e movimento, por exemplo. Seguindo esta linha de raciocínio, [Augusto, Nakashima, e Aghajan 2009] alegam que sistemas AmI, a fim de operar eficientemente, devem conhecer o usuário bem como o ambiente que o cerca possibilitando, assim, que o usuário interaja com o sistema sem precisar demandar grande esforço e atenção. Tal interação deve ser invisível, através de dispositivos que interpretam emoções e ações da pessoa através do uso de sensores.

Relacionado a Ambientes Inteligentes, surgem os Ambientes de Vivência Assistida (*Ambient Assisted Living* – AAL) [ERCIM 2011], que consistem na assistência a domicílio e têm como objetivo prover uma melhor qualidade e uma vida mais segura e saudável [ERCIM 2011], em geral à pacientes idosos. Isto pode ser obtido por meio da criação de um ambiente que possibilite maior autonomia e assistência a atividades rotineiras e que, ao mesmo tempo, garanta uma independência ao usuário, prolongando assim o tempo que ele poderá viver em sua residência [Sun et al. 2009]. Assim, AAL envolve conceitos, produtos e serviços que conectam novas tecnologias no ambiente do próprio paciente [ERCIM 2011], podendo ser usada na prevenção, cura e melhora nas condições de saúde deste, assim como em seu bem estar [Rashidi e Mihailidis 2013].

4. Proposta de Trabalho

De acordo com [Ruscin e Linnebur 2014], o uso incorreto de medicação é um fator agravante no estado de saúde de idosos. Considerando este fator, este trabalho busca propor uma solução em AAL através da definição de um sistema *web* para o gerenciamento de serviços capazes de ajudar o paciente no uso de medicações.

O sistema proposto deve ser utilizado juntamente com um *dispenser* com características pervasivas onde serão colocados os medicamentos do paciente. Tal dispositivo não é tratado neste artigo, porém sugere-se que possua *containers*, cada qual com um tipo de remédio diferente. Além disso, deve ser capaz de manter uma comunicação estável com o sistema proposto neste trabalho. A Figura 1 apresenta um *workflow* com os processos que compõem as funcionalidades do sistema.

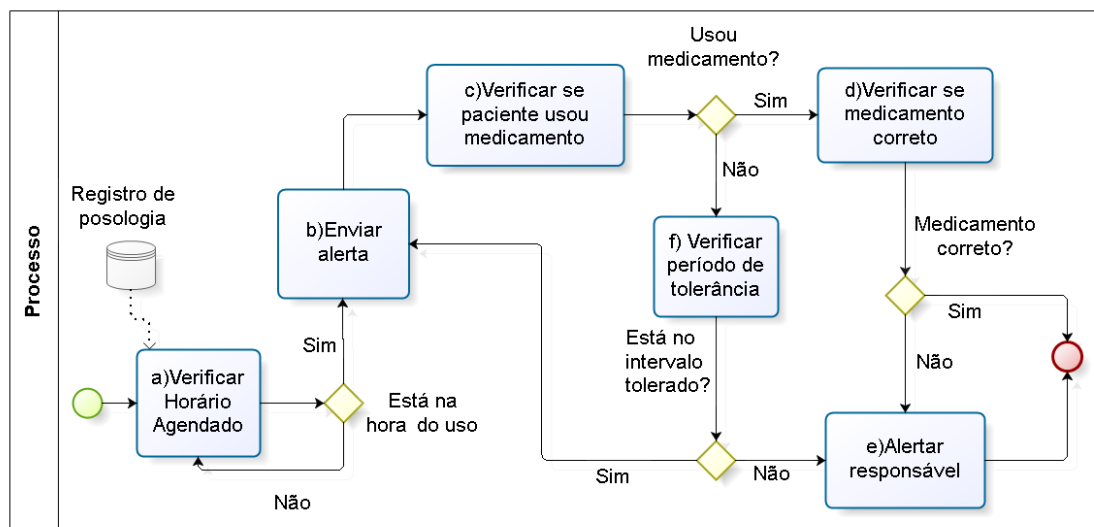


Figura 1. Fluxo de funcionamento do sistema

Com base nas informações recebidas do *dispenser*, o sistema será capaz de: alertar o paciente a tomar o medicamento na hora correta (a,b,c,f); instruir o paciente a usar o medicamento correto (d); alertar um responsável caso o paciente não tenha usado o medicamento corretamente (e); manter um registro de acompanhamento do uso da medicação. A seguir, estas funcionalidades são descritas detalhadamente.

4.1. Alerta Sobre Horário Correto de Medicação

Para cada medicamento deve-se armazenar a posologia, ou seja, a dose e a frequência diária que o paciente deve usá-lo, bem como a hora em que ele deve ser administrado. Com base nestes registros, o sistema verifica periodicamente o horário e alerta o paciente para usar a medicação. Visando o entendimento correto do significado do alerta, sugere-se a utilização de sinais sonoros [Alencar e Neris 2013] e visuais [García-Velázquez et al. 2011] como forma de interação do *dispenser* com o paciente. Ou seja, quando um determinado medicamento deve ser administrado seu *container* emitirá sinais de luz e som. O *container* deverá permanecer em alerta até que seja detectado que o paciente usou a medicação ou de acordo com uma restrição definida no sistema. Esta restrição refere-se a informações específicas de cada medicamento como, por exemplo, um intervalo limite de tolerância entre a hora correta de usar a medicação e o horário exato em que o paciente usou.(por exemplo: 20 minutos).

Ao verificar que o paciente usou a medicação, um alerta é enviado ao sistema notificando esta ação. Caso o sistema não receba esta entrada (de acordo com a restrição definida), assume-se que o paciente não tomou a medicação e um novo alerta deve ser emitido.

4.2. Uso de medicação correta

O *dispenser* poderá comportar mais de um medicamento, logo, é importante auxiliar o paciente a usá-los corretamente. Alertas exibidos no *container* ajudam a evitar enganos, mas o sistema não controla as ações finais do paciente. Caso os sensores detectem que o paciente pegou um *container* com o medicamento errado (informação que não corresponde à esperada), um alerta é enviado ao sistema do *dispenser* que, por sua vez,

deve sinalizar o erro da maneira mais adequada. Visando uma maior segurança do paciente, um responsável também é informado do possível engano.

4.3. Alerta à cuidadores em situações adversas

Caso o paciente não use a medicação no tempo correto ou se o sistema detectar o uso de um medicamento errado, um alerta deve ser enviado a um cuidador (familiar ou responsável). Por exemplo, ao atingir o limite de tolerância pré-estabelecido sem a detecção de uso do medicamento, o sistema notifica o responsável a respeito da situação, especificando o medicamento em questão. Cabe a este responsável verificar o motivo e tomar as devidas providências.

4.4. Registro de acompanhamento de uso de medicação

Visando um melhor acompanhamento do tratamento do paciente, o sistema possui um registro diário do uso da medicação contendo a hora correta da administração, a hora em que de fato foi administrada e uma informação indicando se um alerta de erro foi emitido. Este registro possibilita um melhor acompanhamento da saúde do idoso por seu cuidador. Além disso, estes dados poderão ser exportados, permitindo uma integração com outros sistemas como, por exemplo, um sistema usado pelo médico responsável a fim de acompanhar o tratamento. Os dados do paciente e dos medicamentos são gerenciados por uma página *web* que poderá ser acessada também por responsáveis pelo paciente.

Por se tratar de um ambiente de Vivência Assistida com características pervasivas, sugere-se que o *dispenser* mantenha uma comunicação com outros dispositivos presentes na casa do paciente. Assim os alertas poderão ser emitidos por mais de um dispositivo, dependendo da localização do usuário. Para atingir este objetivo, podem ser usados sensores de localização e uma conexão *web* será responsável pela transmissão do alerta. Destaca-se o fato de que, caso o paciente não seja localizado por nenhum sensor, um alerta deve ser enviado para seu responsável. Esta funcionalidade é importante, pois o paciente pode não se encontrar em casa ou então pode ter sofrido algum tipo de piora em seu estado de saúde.

Para evitar falhas na emissão dos alertas devido a possíveis perdas de conexão com a internet, será mantido um arquivo local com os dados registrados no sistema. Assim, ocorrendo uma perda de conexão, o alerta ainda poderá ser emitido no dispositivo. Caso seja detectado que o paciente não utilizou o medicamento, o responsável será alertado assim que a conexão for restabelecida.

5. Estudo de Caso

Para validação do sistema e considerando o *workflow* proposto, definiu-se um estudo de caso que será descrito a seguir:

Dona Maria¹ é uma aposentada de 64 anos e usa medicamento contínuo para controle de pressão sanguínea três vezes ao dia e diabetes duas vezes ao dia. Ela vive sozinha em sua residência que possui um sistema embarcado, a qual pode ser caracterizada, então, como um AAL. Devido a sua condição, Dona Maria frequentemente esquece o horário correto de sua medicação. A paciente utiliza o

¹ Paciente fictício

sistema proposto neste trabalho com as posologias de seus remédios previamente configuradas por uma segunda pessoa, responsável pelo gerenciamento do sistema.

O sistema é utilizado a partir do registro dos dados da paciente e de uma rede de internet disponível na residência. Dona Maria opta por manter o dispositivo na sua cozinha, mas possui dispositivos computacionais em todos os cômodos da casa capazes de receber alertas do sistema.

Em uma primeira situação, o sistema alerta sobre o horário do medicamento para pressão. Dona Maria encontra-se na cozinha e pode ouvir e visualizar o alerta, fazendo então o uso correto do medicamento. Em um segundo cenário, Dona Maria encontra-se em seu quarto. Sensores dispostos neste cômodo detectam sua localização e, com isso, o alerta é emitido na TV. Devido a uma distração, Dona Maria pega um *container* com medicamento errado. Este fato é detectado pelo sistema e um novo alerta é imediatamente emitido, fazendo com que a paciente troque pelo correto. Joana, filha de Dona Maria, que é a responsável pela paciente, também foi informada deste equívoco e contata sua mãe para verificar a situação.

Em uma terceira situação, o sistema detecta que, apesar de ser notificada para usar o remédio para controle de pressão sanguínea, Dona Maria não o faz. Neste caso, ao atingir o limite do tempo de espera previamente estabelecido, um alerta é enviado para Joana. Ao receber este alerta, Joana contatou Dona Maria e verificou que esta não percebeu o alerta, pois havia adormecido. A agilidade do contato permitiu que Dona Maria ainda pudesse tomar sua medicação sem grande atraso. Os demais alertas para este dia foram respeitados pela paciente.

Assim, pode-se concluir que, com a utilização do sistema para auxiliar pacientes em ambientes AAL, a quantidade de ocorrência de problemas devido ao uso incorreto de medicação pode diminuir consideravelmente.

6. Trabalhos Relacionados

Diversas pesquisas vêm sendo desenvolvidas na área de AAL, principalmente na Europa. Com base nisso e visando aprimorar o serviço proposto neste trabalho, foram analisados alguns trabalhos relacionados às áreas envolvidas.

Em [Ha e Byun 2012], desenvolveu-se um dispositivo ubíquo (bracelete) que, através do uso de um acelerômetro, é capaz de detectar possíveis quedas do paciente. Ao detectar uma queda, uma mensagem é enviada a um familiar.

Em [García-Velázquez, Rodriguez e Andrade 2009], apresenta-se AMDD (*Ambient Medicine Dispenser Display*), que consiste em um *dispenser* de medicamentos que alerta o paciente para tomar sua medicação. Este alerta é feito através da iluminação do *container* onde está o medicamento que deve ser utilizado.

GUIDE ME [García-Velázquez et al. 2011] refere-se a um dispositivo que auxilia no uso de medicação. Consiste em interfaces geométricas que são amarradas à medicação do paciente, onde o número de lados que a forma geométrica possui equivale ao número de vezes que a pessoa deve usar aquele medicamento diariamente. A interface possui luzes que se acenderão na hora do uso do medicamento e se apagarão após o paciente ter usado este medicamento, ou após um tempo pré-definido.

Baseando-se na análise destes trabalhos e demais fatores, o trabalho aqui proposto visa o uso de computação pervasiva através de um sistema de fácil uso, sem a necessidade de uma interface que exija esforço do usuário. O fato de alertar um responsável e o registro de acompanhamento do uso da medicação são diferenças importantes, que não foram identificados em nenhum dos trabalhos relacionados a alertas de medicação. É importante ressaltar que neste projeto não será desenvolvido o dispositivo físico, entretanto este trabalho pode ser utilizado por outras pesquisas já existentes no mercado como, por exemplo, em [García-Velázquez et al. 2011] e [García-Velázquez, Rodriguez e Andrade 2009].

7. Conclusão

O grande aumento da população idosa tem demandado novos serviços que visam prover mais segurança e conforto para esta faixa etária. Este trabalho propôs um sistema *web* a fim de ajudar idosos na administração de sua medicação em um Ambiente de Vivência Assistida. O sistema encontra-se em fase de desenvolvimento e, a partir de sua definição, pretende-se realizar testes em ambientes simulados e posteriormente, se possível, em ambientes reais. Como resultado, almeja-se disponibilizar o serviço a usuários reais visando auxiliá-los durante o tratamento médico e proporcionando um estilo de vida mais independente.

Referências

- Aarts, E. e Wichert, R. (2009) Ambient intelligence. In: *Technology Guide: principles, applications, trends*, p. 244-249. Springer Berlin Heidelberg.
- Alcaniz, M. e Rey, B. (2005) New technologies for ambient intelligence; In: *Ambient Intelligence*. IOS Press.
- Alencar, T.S. e Neris, V.P.A. (2013) Sistemas ubíquos para todos: conhecendo e mapeando os diferentes perfis de interação. In: *IHC '13 Proceedings of the 12th Brazilian Symposium on Human Factors in Computing Systems*, p. 178-187.
- Augusto, J.C., Nakashima, H. e Aghajan, H. (2009) Ambient intelligence and smart environments: a state of the art. In: *Handbook of Ambient Intelligence and Smart Environments: part 1*, p. 3–31 New York: Springer.
- ERCIM News (2011), Special theme: Ambient Assisted Living. Ed. 87. Disponível em: <http://ercim-news.ercim.eu/images/stories/EN87/EN87-web.pdf>
- Friedewald, M. e Raabe, O. (2011). Ubiquitous computing: an overview of technology impacts. *Telematics and Informatics*, v.28, n.2, p.55-65.
- García-Velázquez, J.P. et al. (2011) Supporting the strategies to improve elders' medication compliance by providing ambient aids. *Personal and Ubiquitous Computing*, v. 15, n. 4, p. 389-397.
- García-Velázquez, J. P., Rodriguez, M. D. e Andrade, A. J. (2009) Ambient information systems for supporting elder's independent living at home. In: *Proceedings of the 10th International Work-Conference on Artificial Neural Networks: Part II: Distributed Computing, Artificial Intelligence, Bioinformatics, Soft Computing, and Ambient Assisted Living Table of Contents*. Espanha.

- Ha, Y.G. e Byun, Y.C. (2012) A ubiquitous homecare service system using a wearable user interface device. In: *Computer and Information Science (ICIS), IEEE/ACIS 11th International Conference On*, p. 649-650.
- Henricksen, K., Indulska, J. e Rakotonirainy, A. (2002) Modeling context information in pervasive computing systems; In: *Lecture notes in computer science*, p. 167-180. Alemanha: Springer.
- IBGE - Instituto Brasileiro de Geografia e Estatística (2008), Projeção da população do Brasil por sexo e idade 1980-2050: revisão 2008. Rio de Janeiro: IBGE.
- ISTAG (2001), Scenarios for ambient intelligence in 2010, European Commission Report.
- Jones, V. e Jo, J.H. (2004) Ubiquitous learning environment: an adaptive teaching system using ubiquitous technology. In: *Ascilite Conference, 21*. Austrália. Disponível em: <<http://ascilite.org.au/conferences/perth04/procs/jones.html>>. Acesso em: 2 de abr. 2015.
- Lyytinen, K. e Yoo, Y. (2002) Issues and challenges in ubiquitous computing. *Communications of the ACM*, v.45, n.12, p. 64.
- Machado, A. et al. (2014) A reactive and proactive approach for ambient intelligence. *ICEIS*, 2: 501-512.
- Rashidi, P. e Mihailidis, A. (2013) A survey on ambient-assisted living tools for older adults. *IEEE Journal of Biomedical and Health Informatics*. 17, 3.
- Ruscin, J.M. e Linnebur, S.A. (2014) Drug-related problems in the elderly. Disponível em: <http://www.merckmanuals.com/professional/geriatrics/drug-therapy-in-the-elderly/drug-related-problems-in-the-elderly>. Acesso em: 14 de agosto de 2015.
- Satyanarayana, M. (2001) Pervasive computing: vision and challenges. *Personal Communications IEEE*, v.8, n.4, p. 10 – 17.
- Sun H. et al. (2009) Promises and challenges of ambient assisted living systems. In: *Information Technology: New generations (ITNG), Sixth International Conference On*, p.1201-1207.
- Weiser, M. (1991) The computer for the 21st century. *Scientific American*, v. 265, n. 3, p. 94–104.
- Weiser, M. e Brown, J.S. (1996) Designing calm technology. *Power Grid Journal* 1, v.1, n.1, p.94-100.

Redes Neurais com Neuralnet e Shiny

Jorge Luis Boeira Bavaresco¹, Ronaldo Serpa da Rosa²

¹Instituto Federal de Educação, Ciência e Tecnologia Sul-Rio-Grandense (IFSUL)
Passo Fundo - RS - Brasil

²Universidade de Passo Fundo (UPF)
Passo Fundo - RS - Brasil

jorge.bavaresco@passofundo.ifsul.edu.br, ronaldoserpa84@gmail.com

Abstract. *This article aims to present an approach to implementing neural networks with GUI for use by end users. For this was used open source tools such as the R language and its Neuralnet and Shiny packages. Thus expected to guide the use of neural networks with these technologies for academic and professional purposes.*

Resumo. *Este artigo tem como objetivo apresentar uma abordagem para implementação de redes neurais com interface gráfica para utilização por usuários finais. Para tanto foi utilizado ferramentas de software livre como a linguagem R e seus pacotes NeuralNet e Shiny. Desta forma espera-se nortear a utilização de redes neurais com estas tecnologias para fins acadêmicos e profissionais.*

1. Introdução

A área da inteligência artificial evoluiu nas últimas décadas e atualmente o uso de sistemas inteligentes pode auxiliar em processos de tomadas de decisão das mais diversas áreas do conhecimento. As atividades como o processamento de imagens, aplicações financeiras ou aplicações da área agrícola para a prevenção de doenças podem fazer uso de redes neurais artificiais para auxiliar na tomada de decisão. As redes neurais utilizam variáveis de entrada (covariáveis) esperando obter uma resposta para um determinado problema. Desta forma, pode-se com um determinado conjunto de entradas prever um resultado. Nas redes neurais artificiais, os dados observados são usados para treinar a rede neural, sendo que esta aprende por uma aproximação do relacionamento por interatividade e adaptação dos parâmetros [Hornik et al. 1989].

Existem diversas ferramentas para a utilização de redes neurais artificiais, e neste universo de ferramentas, pode-se encontrar alternativas proprietárias e de software livre. Um dos objetivos deste artigo é apresentar uma solução utilizando software livre. Entre as alternativas, encontra-se a linguagem R e seus pacotes *Neuralnet* e *Shiny*. O R [R Core Team 2014] é ao mesmo tempo uma linguagem de programação e um ambiente para computação estatística e gráficos. Trata-se de uma linguagem de programação especializada em computação com dados. O *neuralnet* [neuralNet 2014] é um pacote do R, desenvolvido por Fritsch e Günther em 2008, que contempla uma função muito flexível para treinar redes neurais *feedforward*. Já o *Shiny* [RStudio and Inc. 2014] é um pacote do R para criação de interfaces gráficas para a web. Ele abstrai do desenvolvedor a habilidade de conhecer linguagens de programação para a web trabalhando assim em um nível alto

de abstração, que possibilita a construção de uma interface gráfica utilizando somente a linguagem R.

Neste artigo abordaremos a utilização de redes neurais artificiais com interface gráfica para o usuário, utilizando o pacote *neuralnet* e *Shiny* da linguagem R. Na seção 2 são apresentados alguns conceitos de rede neural e uma breve introdução sobre os pacotes *neuralnet* e *Shiny*. Após, na seção 3 é apresentado o estudo de caso implementado na prática, com as atividades de criação e treinamento da rede neural artificial, e a criação da interface gráfica. Por fim, na seção 4 os resultados obtidos são apresentados.

2. Referencial teórico

2.1. Redes neurais

O cérebro humano processa as informações de uma forma diferente que o computador digital convencional, esta diferenciação é atribuída principalmente por este ser altamente complexo, não-linear e paralelo. Esta diferença na forma tem motivado os estudos de redes neurais [Simon 2001]. Segundo Haykin, uma rede neural é uma máquina composta de componentes eletrônicos ou simulada por programação em um computador digital, a qual é projetada para modelar a maneira como o cérebro desempenha as suas funções. Uma rede neural, assemelha-se com o cérebro humano em dois aspectos: na aquisição do conhecimento, onde o mesmo é adquirido por meio de um processo de aprendizagem; no armazenamento do conhecimento adquirido, sendo este armazenado nos pesos sinápticos, ou também conhecidos como forças de conexão [Simon 2001].

2.1.1. Perceptrons multicamadas

Existem diversas classes de redes neurais. Conforme [Simon 2001], a Perceptron Multicamadas é uma das mais importantes classes de redes neurais. Este modelo foi criado para suprir a deficiência do modelo perceptron simples, adicionando camadas ocultas que objetivam melhorar o desempenho da rede. As redes da classe *Perceptron* Multicamadas, são compostas de uma camada de entrada, uma ou mais camadas ocultas de nós computacionais e uma camada de saída, conforme Figura 1.

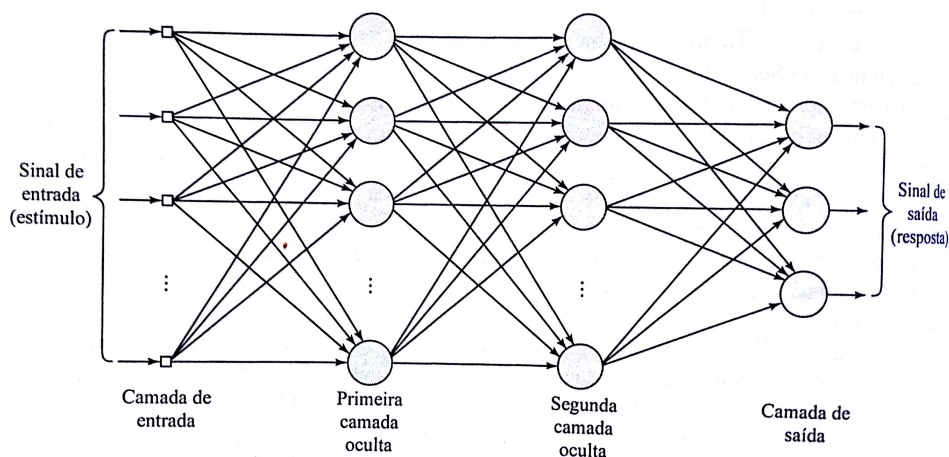


Figura 1. Rede neural multicamada. Fonte: [Simon 2001]

2.1.2. Aprendizado Supervisionado e Backpropagation

As redes *Perceptron* Multicamada utilizam o algoritmo de retropropagação de erro para realizar o treinamento da rede de forma supervisionada. O algoritmo de retropropagação é baseado na regra de aprendizagem por correção de erro, o qual resume-se em dois passos por meio das diferentes camadas da rede: um passo para frente, a propagação, e um passo para trás, a retropropagação [Simon 2001]. Ao realizar o passo para frente, os pesos sinápticos da rede são mantidos inalterados, então o valor de entrada é aplicado nos nós da rede, camada por camada, e um conjunto de saídas é gerado. Durante o passo para trás, os pesos sinápticos são ajustados de acordo com a regra de correção de erro, com o objetivo de alcançar a resposta real desejada para a rede.

2.2. Pacote neuralnet

O *Neuralnet* é um pacote do R para o treinamento de redes neurais. Pode-se utilizar algoritmos como *backpropagation* ou *resilient backpropagation* com ou sem *backtracking* [Fritsch et al. 2012]. Com ele é possível treinar redes neurais de maneira flexível e com um mínimo de esforço. A função do pacote para o treinamento de redes neurais fornece opções para definir o número necessário de camadas ocultas e neurônios ocultos de acordo com a complexidade necessária, a qual é calculada com a adição de camadas ocultas ou neurônios ocultos [Günther and Fritsch 2010].

Entre os argumentos da função para o treinamento da rede neural, pode-se destacar alguns como “*formula*”, que é uma descrição do modelo a ser ajustado, “*data*” que é o conjunto de dados a ser utilizado, “*hidden*” que especifica o número de camadas ocultas e neurônio ocultos de cada camada. Outros parâmetros podem ser utilizados para customizar a rede neural.

2.3. Pacote Shiny

O *Shiny* é um pacote do R criando para facilitar o desenvolvimento de aplicações web. Com ele é possível construir aplicações web interativas de maneira simples, sem a necessidade do desenvolvedor possuir conhecimentos de tecnologias de desenvolvimento para a web. Isto é possível por meio de uma ligação automática entre a entrada do usuário e o processamento do lado do servidor, e de diversos componentes pré construídos que facilitam a elaboração da interface com o mínimo de esforço [RStudio and Inc. 2014].

Desta forma o *Shiny* pode ser considerado um *framework* de desenvolvimento de aplicações web para o R. No site oficial ¹ pode-se encontrar tutoriais, artigos e exemplos de aplicações, o que pode ser utilizado como para para o desenvolvimento.

3. Material e métodos

Nesta seção será descrito o estudo de caso realizado, que consiste no uso de redes neurais para prever se um homem possui a fertilidade alterada ou não. A tecnologia utilizada para criação da rede neural artificial será o pacote *neuralnet* [neuralNet 2014], e para interface gráfica do usuário será utilizado o pacote *Shiny* [RStudio and Inc. 2014], ambos da linguagem R. Primeiramente será descrito de forma detalhada como foi realizado o treinamento da rede neural, e posteriormente como foi criada a interface do usuário

¹<http://shiny.rstudio.com/>

3.1. Dados para treinamento

Para treinamento de uma rede neural, é necessário um conjunto de dados para o treinamento e um conjunto para validação da rede neural. O repositório de dados para treinamento [UIC 2014] foi criado em 1987 por David Aha e colegas de pós-graduação na Universidade da Califórnia em Irvine e disponibiliza conjuntos de dados para treinamento de redes neurais. Este repositório visa auxiliar a análise empírica de algoritmos de aprendizado de máquina. É amplamente utilizado por estudantes, educadores e pesquisadores de todos o mundo como a principal fonte de dados de aprendizagem de máquina.

O conjunto de dados utilizado no treinamento é um conjunto de dados de fertilidade ², onde 100 voluntários forneceram amostras de sêmen analisadas segundo os critérios do WHO 2010 [Organization 2010]. A concentração de esperma é relacionada com dados sociodemográficos, fatores do ambiente, status de saúde e hábitos de vida.

A Tabela 1 apresenta os atributos que compõe o conjunto de dados, como a descrição dos atributos, nome utilizado na coluna dos atributos, significado e valores possíveis para entrada. É realizado um ajuste no conjunto de dados original, fornecendo nomes para os atributos, para facilitar a utilização na linguagem R, e o valor do atributo diagnosis, que possui valores 'O' ou 'N' é transformado em 0 ou 1.

Atributo	Nome Coluna	Entrada	Valores
Estação do ano em que a coleta foi realizada	season	1) inverno 2) Primavera 3) Verão 4) Outono	(-1, -0.33, 0.33, 1)
Idade em que a coleta foi realizada	age	18-36	(0 a 1)
Acidente ou traumas graves	diseases	1) Sim 2) Não	(0, 1)
Acidente ou traumas graves	accident	1) Sim 2) Não	(0, 1)
Intervenções cirúrgicas	surgical	1) Sim 2) Não	(0, 1)
Febre alta no ultimo ano	fevers	1) Menos de três meses atrás 2) Mais de três meses atrás 3) Não	(-1, 0, 1)
Frequência de consumo de álcool	alcohol	1) Várias vezes por dia 2) Todo dia 3) Várias vezes por semana 4) Uma vez por semana 5) Dificilmente ou nunca	(0 a 1)
Habito de fumar	smoking	1) Nunca 2) Ocasionalmente 3) Diariamente	(-1, 0, 1)
Numero de horas sentado durante o dia	hours_sitting	16	(0 a 1)
Resultado	diagnosis	Diagnóstico normal (N), Alterado (O)	(N, O)

Tabela 1: Descrição do conjunto de dados. Fonte: do autor

²Disponível em <http://archive.ics.uci.edu/ml/datasets/Fertility>

3.2. Treinamento da rede neural

É necessário o treinamento da rede neural utilizando o conjunto de dados descrito anteriormente para que seja possível prever o resultado de diagnóstico a partir de um determinado conjunto de entradas. Após a rede ser treinada, ela pode ser salva para posterior utilização pela interface gráfica. Para treinamento da rede usando o pacote *neuralnet* e a linguagem R, o código executa os seguintes passos:

1. Leitura do arquivo com o conjunto de dados
2. Transformação do valor do atributo diagnosis, que possui valores 'O' ou 'N' em 0 ou 1
3. Separação de linhas para o treinamento e linhas para validação
4. Criação e treinamento da rede neural
5. Validação da rede neural
6. Verificação do percentual de acerto da rede

Os passos acima são responsáveis por ler os dados, criar a rede neural e validar a saída da rede neural. Caso o percentual de acerto da rede tenha sido satisfatório, a *workspace* utilizada pelo R contendo a rede neural treinada pode ser salva em um arquivo para que seja utilizada posteriormente.

No código fonte abaixo, são realizados os passos 1 a 3. Na linha 1, o comando *rm()* limpa qualquer dado que estava na memória do R, evitando que informações que haviam sido carregadas anteriormente influenciem na execução do código. A linha 2 com o comando *setwd* define o diretório de trabalho, procurando os arquivos informados no restante do código neste diretório. Após isto, na linha 3 é realizado o carregamento em memória da biblioteca a ser utilizada pela rede neural, com o comando *library("neuralnet")*. Na linha 4 é realizada a leitura de um arquivo, que contém os dados para treinamento e validação da rede. São informados três parâmetros, o primeiro é o nome do arquivo, o segundo indica se temos uma linha que é o cabeçalho com os nomes das colunas, e o terceiro é o caractere que separa as colunas. Na linha 6, como a coluna *diagnosis* vem com valores *O* ou *N*, altera-se o valor para 0 ou 1 respectivamente. Na linha 8 separa-se uma parte dos dados para treinamento, dos registros 1 a 70, e na linha 9 os dados para validação, dos registros 71 a 100.

```
1. rm()
2. setwd("/home/user/fertilityDiagnosis")
3. library("neuralnet")
4. dados<-read.table("fertility_Diagnosis.txt", header=T, sep=",")
5. ## N (normal) = 0 altered (O) = 1
6. dados$diagnosis <- ifelse(dados$diagnosis == "N", 0, 1)
7. ## criando os conjuntos para treinamento e validação
8. dadosTreinamento<-dados[1:70,]
9. dadosValidacao<-dados[71:100,]
```

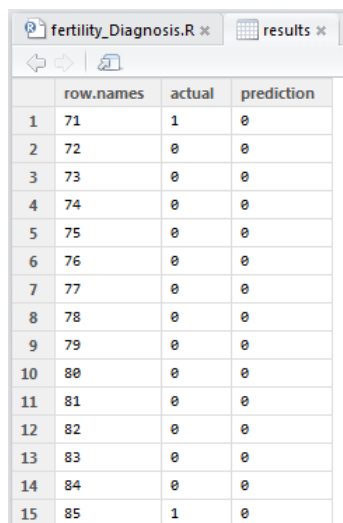
A criação da rede neural pode ser observada no código fonte abaixo, que demonstra o uso do comando *neuralnet*, nas linhas 11 a 15. O primeiro parâmetro passado ao método é a fórmula utilizada para treinamento da rede e o segundo os dados utilizados para o treinamento. O parâmetro *hidden* indica o número de neurônios utilizado na rede. Neste caso foram utilizados 10 neurônios na camada interna. Caso seja necessário a configuração com mais de uma camada interna, no lugar do valor 10 informa-se *c(10,5)*, neste caso criando suas camadas com 10 e 5 neurônios respectivamente. Pode-se configurar o algoritmo usado no aprendizado (*algorithm*), o *threshold* e a taxa de aprendizado

(*learningrate*), que é obrigatório quando utilizado o algoritmo *backpropagation* (*backprop*). Na linha 17 é realizada a plotagem da rede, tornando possível a sua visualização de forma gráfica.

```
10. ## criando a rede neural (NN)
11. redeNeural <- neuralnet(
12.   diagnosis ~ season + age + diseases + accident + surgical + fevers +
13.     alcohol + smoking + hours_sitting, dadosTreinamento, hidden = 10,
14.     lifesign = "minimal", linear.output = FALSE, threshold = 0.01,
15.     algorithm = "backprop", learningrate=0.01)
16. ## plot a rede neural
17. plot(redeNeural, rep = "best")
```

Após a criação da rede, o próximo passo é a sua validação. Para tal, é utilizado o conjunto de dados separado para validação. Com ele poderemos verificar se a rede neural é capaz de prever o resultado do diagnóstico. No próximo código fonte, na linha 19, cria-se um subconjunto dos dados de validação, retirando somente a coluna com o resultado do diagnóstico, armazenando na variável *temp_test*. Depois, na linha 23, usa-se o comando *compute*, passando-se como parâmetros a rede neural criada e os dados temporários para validação (variável *temp_test*), armazenando o resultado na variável *diagnosis.results*. Após, na linha 25 e 26, cria-se um novo conjunto de dados, com o valor do diagnóstico real e o valor do diagnóstico predito pela rede, para que comparações possam ser realizadas. Tal conjunto de dados pode ser visualizado na Figura 2. Das linhas 31 a 35 se calcula o percentual de acerto da rede.

```
18. ## dados temporários para testar o resultado da saida
19. temp_test <- subset(dadosValidacao, select =
20.   c("season" , "age" , "diseases" , "accident" , "surgical" ,
21.     "fevers" , "alcohol" , "smoking" , "hours_sitting"))
22. ## calculando o resultado
23. diagnosis.results <- compute(redeNeural, temp_test)
24. ## criando um dataframe com os dados observados e a predição da rede
25. results <- data.frame(actual = dadosValidacao$diagnosis,
26.   prediction = diagnosis.results$net.result)
27. results
28. # arredondando a saida do resultado
29. results$prediction <- round(results$prediction)
30. results
31. totalCerto <- sum(results$prediction == results$actual)
32. #totalErrado <- nrow(results) - totalCerto
33. ##percentual de acerto
34. percentualCerto <- totalCerto * 100 / nrow(results)
35. percentualCerto
```

	row.names	actual	prediction
1	71	1	0
2	72	0	0
3	73	0	0
4	74	0	0
5	75	0	0
6	76	0	0
7	77	0	0
8	78	0	0
9	79	0	0
10	80	0	0
11	81	0	0
12	82	0	0
13	83	0	0
14	84	0	0
15	85	1	0

Figura 2. Resultado obtido pela rede neural. Fonte: do autor

O tempo para o treinamento da rede neural varia conforme o tamanho e complexidade do conjunto de dados. Desta forma conjuntos maiores e mais complexos demandam mais tempo para a rede neural ser treinada. Sendo assim, é importante salvar a rede neural após o treinamento, possibilitando a sua utilização sem a necessidade de novo treinamento. No próximo código fonte, na linha 37, é realizada a persistência em um arquivo da *workspace* do R, que no momento contém a rede neural treinada.

```
36. ## salvando a workspace
37. save.image(file="redeNeuralTreinada.RData")
```

3.3. Interface Gráfica

Para a criação de uma interface gráfica para a utilização da rede neural treinado por usuário final, foi utilizado o pacote Shiny da linguagem R. A aplicação shiny consiste de três arquivos, *global.R*, *ui.R* e *server.R*.

No próximo código fonte é descrito o arquivo *global.R*. Ele define os recursos que serão utilizados pelos outros componentes da aplicação (*ui.R* e *server.R*). Na linha 1 e 2, são carregados os pacotes necessários, e na linha 3 é definido o diretório de trabalho. Já na linha 4, é efetuado o carregamento da *workspace* do R, que contém a rede neural treinada.

```
1. library("shiny")
2. library("neuralnet")
3. setwd("/home/user/fertilityDiagnosis/")
4. load("redeNeuralTreinada.RData")
```

O arquivo *ui.R* é responsável por gerar a interface gráfica, onde são definidos os componentes que a compõe, e que serão utilizados pelo arquivo *server.R*. O próximo código fonte apresenta um trecho deste arquivo, onde se define um campo de entrada de valores, que recebe três parâmetros, o primeiro é um identificador do componente, o segundo é o texto que aparece na tela, e o terceiro o conjunto de valores para seleção do usuário.

```
selectInput("season", "Estação de coleta da amostra:",
c("Inverno" = -1, "Primavera" = -0.33, "Verão" = 0.33, "Outono" = 1))
```

O arquivo `server.R` é responsável pelo processamento. O trecho que realizou o processamento da entrada do usuário e calcula o resultado é descrito no próximo código fonte. No código fonte anterior observa-se um componente chamado *season*. O valor selecionado no componente pode ser acessado por meio do comando `input$season`. Desta forma se obtêm os valores selecionados pelo usuário. Sendo assim é necessário se obter os valores digitados pelo usuário, e armazenar esta entrada em um conjunto de dados que contém somente uma linha, tornando assim possível a predição da entrada do usuário com a rede neural. Nas linha 4 a 15 são obtidos os valores selecionados pelo usuário e é criada uma estrutura de dados com esta entrada. Na linha 17 por meio do comando `compute`, utilizando a rede neural criada e a entrada do usuário, é predito o resultado. Nas linhas 18 a 20, se captura o resultado da rede neural, transformado a saída (0 ou 1) no texto Normal ou Alterado que será exibido para o usuário.

```
1. output$resultadoDiagnostico <- renderText({
2.   withProgress(message = 'Processando..', value = 0, {
3.     ## obtendo o valor da idade
4.     vlrIdade <- 1 / 36
5.     idade <- round(input$age * vlrIdade,2)
6.     vlrHoraSentado <- 1 / 16
7.     horasSentado <- round( input$hours_sitting * vlrHoraSentado,2)
8.     entradaUsuario <- data.frame(season = as.numeric(input$season) , age = as.numeric(idade) ,
9.     diseases = as.numeric(input$diseases) ,
10.    accident = as.numeric(input$accident) ,
11.    surgical = as.numeric(input$surgical) ,
12.    fevers = as.numeric(input$fevers) ,
13.    alcohol = as.numeric(input$alcohol) ,
14.    smoking = as.numeric(input$smoking) ,
15.    hours_sitting = as.numeric(horasSentado))
16.    ## calculando o resultado da rede com a entrada do usuário
17.    resultado <- compute(redeNeural, entradaUsuario)
18.    diagnostico <- round(resultado$net.result[1,])
19.    saidaDiagnostico <- ifelse(diagnostico == 0, "Normal", "Alterado")
20.    paste("Diagnóstico: ",saidaDiagnostico) }) })
```

A Figura 3 apresenta a interface gráfica criada.

Diagnóstico de fertilidade

Figura 3. Interface gráfica. Fonte: do autor

4. Resultados

A implementação da solução é descrita na seção 3. O objetivo original era comprovar a viabilidade de uma rede neural eficiente com o pacote *neuralnet*, e a utilização do pacote *Shiny* para a construção de uma interface gráfica.

Para a escolha da rede neural mais eficiente, foram realizadas várias tentativas de configuração e se observou o resultado com cada uma delas. Estes resultados podem ser visualizados na Tabela 2. Nem sempre a criação da rede resultava num mesmo percentual de acerto, sempre observando-se variações para cima e para baixo. Na primeira configuração com uma camada de 10 neurônios, a configuração do algoritmo '*backprop*' e '*sag*' apresentaram os melhores resultados, com 83.33% e 86.66% para os menores e maiores percentuais de acerto. O melhor resultado observado foi com o algoritmo '*backprop*' e '*slr*', na configuração da rede com 4 camadas com 10, 7, 6 e 5 neurônios, que apresentou até 90% de acerto.

Com o pacote *Shiny* foi possível construir uma interface gráfica para a utilização da rede neural de forma simples e prática. O resultado pode ser observado na Figura 3.

Algoritmo	Camadas	Neuronios camadas internas	Threshold	Acerto Menor	Acerto Maior
backprop	1	10	0.01	83.33%	86.66%
backprop	1	10	0.001	70.00%	86.66%
rprop+	1	10	0.01	76.66%	86.66%
rprop-	1	10	0.01	73.33%	86.66%
sag	1	10	0.01	83.33%	86.66%
slr	1	10	0.01	73.33%	86.66%
backprop	4	10 , 7 , 6 , 5	0.01	86.66%	90.00%
backprop	4	10 , 7 , 6 , 5	0.001	83.33	86.66%
rprop+	4	10 , 7 , 6 , 5	0.01	76.66%	90.00%
rprop-	4	10 , 7 , 6 , 5	0.01	83.33%	86.66%
sag	4	10 , 7 , 6 , 5	0.01	83.33%	86.66%
slr	4	10 , 7 , 6 , 5	0.01	86.66%	90.00%

Tabela 2: Resultados. Fonte: do autor.

5. Conclusão

Este trabalho se propôs a apresentar a utilização de redes neurais artificiais utilizando o pacote *neuralnet*, e a criação da interface gráfica utilizando o pacote *Shiny*, ambos da linguagem R, fazendo uso de um conjunto de dados para treinamento da rede.

No estudo de caso foi utilizado um conjunto de dados de fertilidade, para a criação e validação da rede neural. Comprovou-se que é viável a utilização de redes neurais com o pacote *neuralnet*, com um bom percentual de acerto da rede.

Foram realizadas medições com as diferentes configurações do *neuralnet*, visando a busca de uma arquitetura de rede com percentual de acerto maior, conseguindo-se em algumas configurações percentuais de acerto de até 90%.

A criação da interface gráfica com o pacote *Shiny* comprovou a viabilidade do uso desta tecnologia para a utilização de redes neurais, com um baixo nível de complexidade na implementação.

Pode-se concluir que o pacote *neuralnet* da linguagem R é uma alternativa para a criação de redes neurais artificiais, e o pacote *Shiny* uma alternativa para a criação de interfaces, com as vantagens de serem ferramentas livres de custos, com baixo nível de complexidade de implementação e bom desempenho.

Referências

- Fritsch, S., Guenther, F., and following earlier work by Marc Suling (2012). *neuralnet: Training of neural networks*. R package version 1.32.
- Günther, F. and Fritsch, S. (2010). *neuralnet: Training of neural networks*. *The R journal*, 2(1):30–38.
- Hornik, K., Stinchcombe, M., and White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366.
- neuralNet (2014). Package *neuralnet* - training of neural networks. Disponível em: <http://cran.r-project.org/web/packages/neuralnet/neuralnet.pdf>. Acesso em 06 Julho 2014.
- Organization, W. H. (2010). *World health statistics 2010*. World Health Organization.
- R Core Team (2014). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.
- RStudio and Inc. (2014). *shiny: Web Application Framework for R*. R package version 0.10.1.
- Simon, H. (2001). *Redes neurais—princípios e prática*.
- UIC (2014). *Uc irvine machine learning repository*. Disponível em: <http://archive.ics.uci.edu/ml/index.html>. Acesso em 06 Julho 2014.

Identificação de Situações de Alunos em um Ambiente Educacional Ubíquo: Explorando o Processamento Dinâmico de Regras Semânticas

Lidiane Costa da Silva¹, João Ladislau B. Lopes^{2,3}, Ana Marilza Pernas¹

¹Centro de Desenvolvimento Tecnológico – Universidade Federal de Pelotas (UFPel)
Pelotas – RS – Brasil

²Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Porto Alegre – RS – Brasil

³Instituto Federal Sul Rio-Grandense – Campus Visconde da Graça (IFSul-CAVG)
Pelotas – RS – Brasil

{lcdsilva, marilza}@inf.ufpel.edu.br, jlblopes@inf.ufrgs.br

Abstract. *One of the main research challenges of Ubiquitous Computing is the situation awareness, which can be considered as particularization of context awareness. One strategy for identifying situations is the use of semantic rules defined based on a context model specified with ontologies. Thus, it becomes possible reasoning about these valid information for certain situations of interest. In this sense, the central contribution of this paper is the possibility of a dynamic processing of semantic rules for identifying situations in an adaptative educational environment ubiquitous.*

Resumo. *Um dos principais desafios de pesquisa da Computação Ubíqua é a consciência de situação, a qual pode ser considerada como uma particularização da consciência de contexto. Uma das estratégias para identificação de situações consiste no emprego de regras semânticas definidas com base em um modelo de contexto especificado com ontologias. Com isso, torna-se possível o raciocínio a respeito destas informações válidas para determinadas situações de interesse. Assim, o presente trabalho apresenta, como contribuição central, a viabilização de um processamento dinâmico de regras semânticas visando à identificação de situações de alunos que utilizam um ambiente educacional ubíquo.*

1. Introdução

Mark Weiser, em seu artigo histórico, definiu a Computação Ubíqua (UbiComp) como a área de pesquisa que estuda a integração da tecnologia às atividades cotidianas. Nesse sentido, a UbiComp tem como premissa fundamental o acesso ao ambiente computacional do usuário o tempo todo, em qualquer lugar e com qualquer dispositivo [Weiser 1991].

Para contemplar essa premissa, um desafio central é a consciência de situação. Esta pode ser considerada como uma particularização da consciência de contexto, onde

situações são vistas como contextos logicamente ligados [Anagnostopoulos 2006]. Uma situação consiste da interpretação de elementos de contextos, relacionando cada um de forma a prover alguma informação válida em um intervalo de tempo específico [Lopes et al. 2014].

Para possibilitar o processamento computacional dos dados contextuais e a consequente identificação de situações, estes devem ser representados de uma forma que seja processável por máquina, definindo um modelo que seja capaz de representar e viabilizar o processamento dos dados relativos aos contextos. Nesse sentido, o raciocínio sobre o contexto permite a identificação de situações, sendo que uma das possibilidades de implementação do raciocínio é o emprego de regras [Pernas 2012].

O presente trabalho utiliza como cenário de aplicação um ambiente educacional ubíquo, empregando neste cenário um modelo de contexto do aluno baseado em ontologias. Com isso, o trabalho tem como objetivo viabilizar a identificação da situação atual dos alunos, utilizando regras semânticas no processamento das informações contextuais.

Este artigo está estruturado da seguinte forma: a seção 2 apresenta os trabalhos relacionados. Na seção 3 é descrito o foco de desenvolvimento, onde o cenário educacional é exposto juntamente com a descrição da rede de ontologias que compõem este trabalho. Na seção 4 é apresentada a prototipação, descrevendo seu desenvolvimento e as tecnologias empregadas no trabalho. A seção 5 traz as considerações finais e os trabalhos futuros.

2. Trabalhos Correlatos

Existem vários trabalhos correlatos que utilizam ontologias tanto para modelagem quanto para raciocínio sobre o contexto. Destes, destacam-se três que tratam especificamente sobre ontologias para modelagem de situação: (i) O'Brien (2009), que utiliza ontologias específicas a aplicações sensíveis à localização; (ii) Baumgartner et al. (2010), onde a ontologia tem foco nos sistemas de trânsito, onde é necessário detectar o momento específico em que os eventos ocorrem no sistema, sendo assim, conceitos de alto nível são especializados para modelagem da dimensão tempo e (iii) Matheus et al. (2005), define uma ontologia genérica para modelagem de situação e a partir dela outros conceitos podem ser definidos, possibilitando a extensão para cenários específicos.

A ontologia utilizada nesse trabalho reusa alguns conceitos vindos das propostas de Baumgartner et al. (2010) e Matheus et al. (2005), principalmente por apresentarem ontologias de fácil reuso e conceitos bem fundamentados.

3. Foco do Desenvolvimento

Os ambientes educacionais adaptativos constroem um modelo dos objetivos dos usuários, suas preferências e conhecimento e utilizam estas informações, juntamente com informações relacionadas à interação com o ambiente para adaptar o conteúdo de acordo com o contexto de interesse dos usuários.

O AdaptWeb[®] (Ambiente de Ensino-Aprendizagem Adaptativo na Web) [Oliveira et al. 2003] foi empregado como cenário de aplicação deste trabalho. Este ambiente é um sistema de hipermídia adaptativo de educação à distância baseado na

web, opensource disponível no SourceForge (<http://adaptweb.sourceforge.net/>), que possui a finalidade de adaptar o conteúdo, a apresentação e a navegação de acordo com a situação dos alunos. A sua adaptação é suportada pela criação de um modelo flexível do estudante (modelo do usuário), onde, para cada estudante, são armazenadas informações pessoais tais como seu background, conhecimento, preferências, histórico navegacional e recursos tecnológicos [Palazzo et al. 2008].

3.1. Rede de Ontologias

Neste trabalho foi utilizado um modelo para gerência de situações de aprendizagem, na forma de uma rede de ontologias [Pernas 2012], construída a partir do mapeamento de três domínios: do aluno, educacional e tecnológico.

Assim, esta rede de ontologias modela os elementos contextuais que descrevem o dispositivo computacional utilizado pelo aluno, sua localização física, perfil e particularidades do aluno e o domínio educacional. Os grupos de ontologias que constituem cada domínio são apresentados na Figura 1.

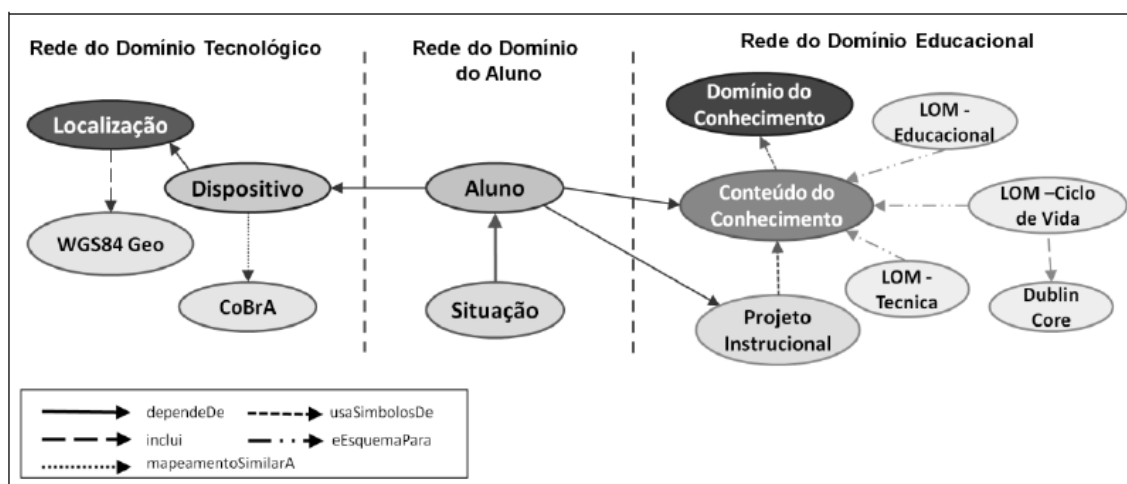


Figura 1 - Rede de Ontologias

Domínio Educacional

A rede de ontologias do domínio educacional tem por objetivo estruturar o conteúdo usado para aprendizagem e conceitos relativos a cursos e disciplinas. As seguintes ontologias foram reusadas: (i) LOM-Ciclo de Vida [IEEE 2002]; (ii) LOM-Técnica [IEEE 2002]; (iii) LOM-Educacional [IEEE 2002]; (iv) Dublin Core [DCMI 2011]; (v) Conteúdo do Conhecimento [Muñoz 2004] e (vi) Domínio do Conhecimento [Muñoz 2004].

Domínio Tecnológico

A rede de ontologias do domínio tecnológico é responsável pela representação e gerência dos dados de contexto relacionados ao ambiente tecnológico e físico do aluno. Este agrupamento foi feito devido ao sensoriamento da localização ser executado por meio do dispositivo computacional usado e se tratar de dados de identificação da latitude e longitude do ponto onde o dispositivo computacional sendo usado pelo aluno está localizado.

Assim, a rede de ontologias do Domínio Tecnológico é composta pelas seguintes ontologias: (i) WGS84 Geoposicionamento [W3C 2003]; (ii) CoBrA (*Context Broker Architecture*) [Chen 2003]; (iii) Dispositivo [Pernas 2012] e Localização [Pernas 2012].

Domínio do Aluno

A ontologia Aluno conceitua as preferências (estilo cognitivo de aprendizagem e estilo navegacional), trajetória de aprendizado, competências e objetivos do aluno [Muños; Palazzo 2004], o dispositivo computacional usado para navegação no ambiente educacional e o local onde se encontra no momento.

A ontologia de situação é responsável por acessar os elementos contextuais e desempenhar alguma forma de raciocínio sobre estes dados, de forma a definir a situação corrente.

3.2. Estudo de Situações

Cada elemento de contexto instanciado é avaliado para determinação da situação, apontando restrições que devem ser consideradas para a recomendação de objetos de aprendizagem. Na continuidade são apresentadas as possíveis instâncias do contexto, bem como a descrição das situações definidas em Pernas (2012) encontra-se na Tabela 1.

- Estilo cognitivo: forma com que os alunos tem facilidade em lembrar de coisas. Neste trabalho é utilizado o modelo de Felder e Silverman (1988), que define quatro dimensões: visual, verbal, sequencial e global, sendo esse modelo o mais frequentemente utilizado e considerado por muitos autores como o mais apropriado para o uso em ambientes educacionais.
- Atividade (exercício ou estudo): de acordo com a atividade, é possível determinar o objetivo do aluno no ambiente.
- Desempenho: de acordo com o andamento do aluno, durante a atividade, objetos são apresentados para auxílio.
- Dispositivo: a detecção do tipo de dispositivo é determinada pelo tamanho e resolução da tela de display.
- Velocidade de conexão: permite ou impõem flexibilidade para apresentação de objetos de aprendizagem.

Tabela 1 - Descrição das Situações

Situação	Descrição
S_01	aluno com estilo cognitivo visual e sequencial, desenvolvendo exercício com bom andamento, usando seu <i>smartphone</i> com baixa velocidade de conexão, em trânsito
S_02	aluno com estilo cognitivo visual e sequencial, realizando uma atividade de estudo com bom andamento, usando seu computador com alta velocidade de conexão, em casa
S_03	aluno com estilo cognitivo visual e global, desenvolvendo exercícios com mau andamento, usando seu computador com alta velocidade de conexão na universidade
S_04	aluno com estilo cognitivo visual e global, desenvolvendo atividade de estudo com bom andamento, usando seu computador com alta velocidade de conexão em casa
S_05	aluno com estilo cognitivo verbal e sequencial, desenvolvendo exercício com bom andamento, usando seu <i>smartphone</i> com alta velocidade de conexão em trânsito
S_06	aluno com estilo cognitivo verbal e sequencial, realizando exercício com mau andamento, usando seu computador com alta velocidade de conexão na universidade
S_07	aluno com estilo cognitivo verbal e global, realizando exercício com bom andamento, usando seu <i>smartphone</i> com baixa velocidade de conexão em trânsito
S_08	aluno com estilo cognitivo verbal e global, realizando exercício com mau andamento, usando seu <i>smartphone</i> com alta velocidade de conexão em trânsito
S_09	aluno com estilo cognitivo verbal e global, realizando estudo com bom andamento, usando seu <i>smartphone</i> com baixa velocidade de conexão em trânsito
S_10	aluno com estilo cognitivo verbal e global, realizando estudo com mau andamento, usando seu <i>smartphone</i> com alta velocidade de conexão em trânsito
S_11	aluno com estilo cognitivo visual e sequencial, realizando uma atividade de exercício com bom andamento, usando seu computador com alta velocidade de conexão, em casa

4. Prototipação

Foi desenvolvido um protótipo para o processamento dinâmico de regras semânticas, as quais anteriormente eram processadas estaticamente através de uma ferramenta como o Protégé (<http://protege.stanford.edu/>). Para tal, foram estudadas algumas tecnologias.

4.1. Tecnologias Empregadas

No desenvolvimento deste trabalho foram empregadas diversas tecnologias, entre elas OWL-DL (*Web Ontology Language – Description Logic*), SWRL (*Semantic Web Rule Language*), API Jena e Raciocinador Pellet.

Desde 2004, a W3C, recomenda como linguagem padrão para especificação de ontologias, a OWL [Bechhofer et al. 2004], pois apresenta todos os benefícios das outras linguagens, tais como DAML+OIL, RDF e RDF-Schema, revisando e incorporando algumas melhorias, como a criação de um vocabulário mais extenso para a descrição de propriedades e classes, permitindo cardinalidade, igualdade, características das propriedades e descrição de relacionamentos entre classes. A versão DL provê um maior grau de expressividade onde todas as conclusões são computáveis e todas as computações terminam em tempo finito.

As linguagens para definição de regras devem ser livres de ambiguidades e indeterminações, claras e intuitivas em sua sintaxe, além de respeitar uma padronização, para que se possa preservar a facilidade de troca de informação e integração de dados. Neste sentido o raciocínio sobre ontologias se faz possível, através de linguagens que agregam mecanismos de inferência e especificações a linguagem OWL. Advindo do RuleML (*Rule Markup Language*), a SWRL é baseada em OWL DL, sendo a linguagem para definição de regras padrão recomendada pelo W3C, com o objetivo de padronizar a definição de regras em ontologias.

A implementação de uma abordagem semântica, baseada em ontologias, para o processamento das informações de contexto requer o suporte de ferramentas para criação e edição de ontologias, raciocínio, bem como uso de ontologias em aplicações.

Em geral, essas ferramentas necessitam de algum tipo de API que permita que as ontologias sejam lidas, manipuladas e consultadas. No presente trabalho foi utilizado a API Jena, como interface de programação para manipulação de ontologias, integrando o processamento de regras semânticas desenvolvidas, na perspectiva da linguagem de programação Java. Esta API foi desenvolvida por pesquisadores da Hewlett Packard Labs, passando em 2009 para a Apache Software Foundation, o projeto API Jena, foi desenvolvido em Java [ORACLE 2015] e tem distribuição sob licença Apache License 2.0.

O raciocinador Pellet foi integrado ao protótipo por possuir suporte as regras SWRL e sua possibilidade de integração com a API Jena. Foi originalmente desenvolvido para servir como ferramenta de prova da viabilidade de implementação de raciocinadores para os conceitos sugeridos pelo W3C para a OWL 1. Obteve a aprovação como ferramenta para inferir sobre a OWL 1, sendo o primeiro raciocinador a suportar toda a expressividade da OWL-DL [Sirin et al. 2007]. Atualmente, o Pellet suporta parte da expressividade da OWL 2, com os construtores necessários e todos os serviços de inferência básicos são providenciados pelo algoritmo baseado em Tableau.

4.2 Desenvolvimento do Protótipo

Com base nos estudos realizados, foi implementado um protótipo visando à identificação da situação de alunos no ambiente educacional AdaptWeb[®], baseado em um processamento dinâmico das regras construídas em SWRL contidas na ontologia proposta por Pernas (2012), utilizada neste trabalho.

O caráter dinâmico diz respeito à possibilidade de acesso à ontologia e seus dados, bem como à execução das regras através de uma aplicação desenvolvida com linguagem de programação (Java), utilizando uma API (Jena) para manipulação das ontologias especificadas em OWL-DL.

Considerando que a API Jena não possui um raciocinador compatível com a linguagem SWRL, foi utilizado o raciocinador Pellet para o processo de raciocínio sobre as regras e o consequente processamento das mesmas.

Para utilizar os dados contidos nesta rede de ontologias, deve-se inicialmente carregá-la, isto é feito através da API Jena, com a qual é possível realizar a criação de um modelo ontológico. Tal modelo apropria-se de todas as informações contidas na rede ontológica. Para a validação dos dados instanciados foram realizados testes nas regras existentes, usando dados reais de alunos em disciplinas do AdaptWeb[®].

Durante a criação do modelo ontológico é definido o raciocinador Pellet que será utilizado para a realização das inferências sobre as instâncias de alunos e suas respectivas situações. Logo após a criação do modelo ontológico é lido o arquivo OWL, que contém os dados instanciados da ontologia.

Para validar a execução das regras e a instanciação das situações na ontologia, foram implementadas duas funcionalidades para o protótipo, sendo estas acessadas em abas específicas na interface do protótipo: (i) consulta das situações de alunos; e (ii) consulta dos alunos que se encontram em determinada situação.

a) Busca de Situações

A aba da Busca de Situações é composta por uma caixa de seleção, onde é possível realizar a escolha de um aluno, e de uma área de texto onde as situações em que os alunos encontram-se serão mostradas, conforme é apresentado na Figura 2.

A caixa de seleção é preenchida na execução do programa com o resultado de uma consulta SPARQL (<http://www.w3.org/TR/rdf-sparql-query/>), que retorna todos os estudantes presentes na ontologia.

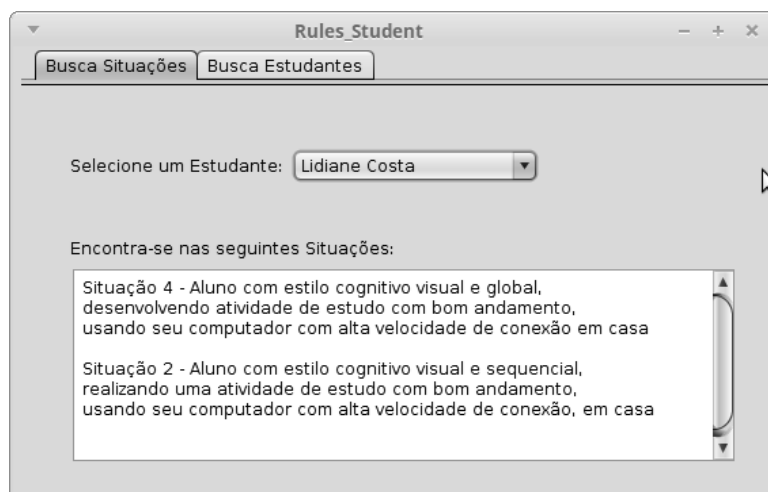


Figura 2 - Interface da Busca Situações

No momento em que um estudante é selecionado, outra consulta SPARQL é disparada para a identificação das situações nas quais este aluno se encontra. Cabe salientar que esta consulta depende das inferências realizadas pelo raciocinador utilizando as regras SWRL.

b) Busca Estudantes

Similar ao Busca Situações, a Busca Estudantes também é composta por uma caixa de seleção, onde é possível realizar a escolha de uma situação, e de uma área de texto onde os nomes dos alunos que se encontram nas respectivas situações são apresentados, conforme se observa na Figura 3.

As situações que populam a caixa de seleção são preenchidas na execução do programa com o resultado de uma consulta SPARQL.

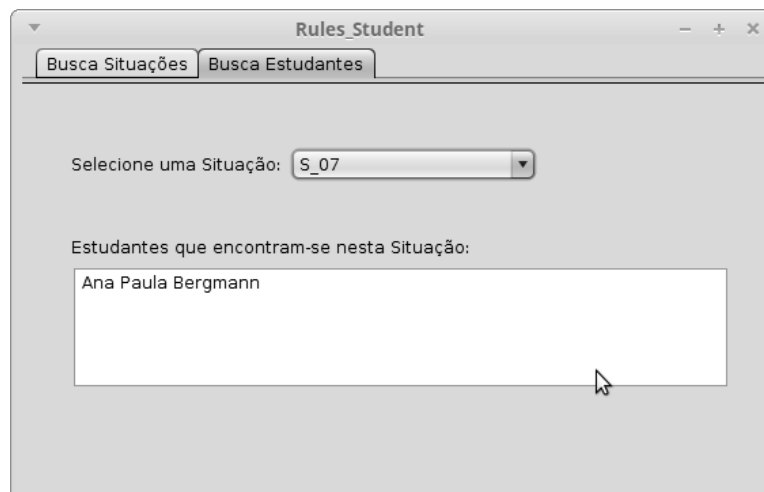


Figura 3 - Interface da Busca de Estudantes

Na Listagem 1 é descrita uma regra SWRL, aplicada aos dados da instância aluno que configuram a situação em que a mesma se encontra, utilizando as restrições já mencionadas acima e caracterizando a identificação de situações.

Listagem 1. Regra em SWRL

```
Aluno(x) ∧ estiloCaptacao (?x, visual) ∧ estiloEntendimento (?x, global) ∧ faz (?x, ?y) ∧ Estudo(?y) ∧ bomDesempenho (?x, ?y) ∧ usa (?x, ?z) ∧ ComputadorPessoal(?z) ∧ temConexao (?z, alta) ∧ localizadoEm (?x, casa) => temSituacao (?x, S_04)
```

5. Conclusões

O processo de tratamento das informações contextuais é relevante visando à identificação de situações. Nesse sentido, o presente trabalho explora uma abordagem baseada em regras relacionadas à ontologia utilizada para representação e processamento do contexto.

Na avaliação desta abordagem foi empregado um cenário de aplicação na área educacional. Para validação e testes do uso das regras semânticas foi desenvolvido um protótipo para identificação de situações dos alunos usuários do ambiente AdaptWeb®. Os resultados obtidos com a execução do protótipo permitiram comprovar a viabilidade desta abordagem.

A seguir alguns trabalhos futuros que podem ser realizados na continuidade deste trabalho são: (i) adicionar ao protótipo a funcionalidade de ativar e desativar regras dinamicamente, (ii) empregar a proposta de uso de regras semânticas para identificação de situações em outros domínios, (iii) desenvolver uma interface para criação e edição de regras (iv) realizar testes com outros modelos ontológicos e as correspondentes regras semânticas.

Referências

Anagnostopoulos, C.B., et al. "Situation Awareness: Dealing with Vague Context". In IEEE International Conference on Pervasive Services, Jun. 2006.

- Baumgartner, N. et al. "BeAware! Situation awareness, the ontology-driven way". *Data & Knowledge Engineering*, v. 69, issue 11, p. 1181-1193, 2010.
- Bechhofer, S.; Harmelen, F. van; Hendler, J.; Horrocks, I.; McGuinness, D. L.; Patel-Schneider, P. F.; Stein, L. A. "OWL Web Ontology Language Reference". Disponível em: <<http://www.w3.org/TR/owl-ref/>>. Acesso em agosto de 2015.
- Chen, 2003. "CoBrA Device Ontology", V. 4. Disponível em: <<http://cobra.umbc.edu/ontologies.html>>. Acesso em agosto de 2015.
- DCMI – "Dublin Core Metadata Initiative", Dublin Core Metadata Element Set, Version 1.1, 2011. Disponível em: <<http://dublincore.org/documents/dces/>>. Acesso em agosto de 2015.
- Felder, R.M, Silverman, L.K. "Learning and Teaching Styles in Engineering Education". *Engineering Education*, v. 78(7), p. 674–681, 1988.
- IEEE LTSC. "Learning Technology Standards Committee, Draft Standard for Learning Object Metadata", IEEE 1484.12.1-2002. 2002. Disponível em: <http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf>. Acesso em agosto de 2015.
- Lopes, J., Souza, R., Geyer, C., Costa, C., Barbosa, J., Pernas, A., and Yamin, A. (2014). "A Middleware Architecture for Dynamic Adaptation in Ubiquitous Computing". *Journal of Universal Computer Science*, v.20, n.9, p.1327–1351, sep 2014.
- Matheus, C. et al. "An Assistant for Higher-Level Fusion and Situation Awareness". In: *Spie Multisensor, Multisource Information Fusion: Architectures, Algorithms, and Applications*, 2005. Proceedings... Orlando, FL, p. 75-85.
- Muñoz, L. S. "Ontology-based Metadata for e-learning Content". 2004, 149 f, Dissertação (Mestrado em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre.
- Muñoz, L. S., Palazzo Moreira de Oliveira, José. "Adaptive Web-Based Courseware Development Using Metadata Standards and Ontologies". In: *International Conference on Advanced Information Systems Engineering - Caise*, 2004, Riga, Latvia. Proceedings... , p. 414–428.
- O'Brien, P. "An Ontology for Mobile Situation Aware Systems". *Australian Journal of Information Systems*, 2009.
- Oliveira, J. P. M. et al. "AdaptWeb: um ambiente para ensino-aprendizagem adaptativo na web". *Educar em revista*, n.107, p.175–198, 2003.
- Oracle Java. Disponível em: <https://www.java.com/pt_BR/about/whatis_java.jsp>. Acesso em agosto de 2015.
- Palazzo, José M. de Oliveira; Lima, J. V.; Gasparini, I., Pimenta, M. S., Brunetto, M. A. C., Proença JR, M., Faggion, R. (2008) "Adaptive Multimedia Content Delivery in AdaptWeb". *XIII Taller Internacional de Software Educativo TISE*, 2008, Santiago. *Nuevas Ideas en Informática Educativa*. v.4. p. 23-39.

- Pernas, A. M. “Sensibilidade à Situação em Sistemas Educacionais na Web”. 2012. 164p. Tese (Doutorado em Computação) - UFRGS, Porto Alegre, RS.
- Sirin, E.; Parsia, A. B.; Grau, B.C.; Kalyanpur, A.; Katz, Y., “Pellet: A practical OWL-DL reasoned”. Web Semantics: Science, Services and Agents on the World Wide Web, vol. 5, no. 2, pp. 51-53, June, 2007.
- Weiser, M. “The Computer for the 21st Century”. Scientific American, v.3, n.265, p.94–104, Setembro 1991.
- W3C, Semantic Web Interest Group. “WGS84 Geo Positioning: an RDF vocabulary”, 2003. Disponível em: <http://www.w3.org/2003/01/geo/wgs84_pos>. Acesso em agosto de 2015.

Uma Abordagem para Suporte à Decisão Clínica Baseada em Semântica no Domínio da Nefrologia

Esaú A. Tavares¹, Natasha C. Q. Lino¹

¹Centro de Informática, Universidade Federal da Paraíba, João Pessoa – PB, Brasil.

esau@di.ufpb.br, natasha@ci.ufpb.br

Abstract. *Chronic kidney disease is a progressive renal injury, insidious in onset, asymptomatic and associated with high morbidity and mortality rates, besides having difficult diagnose in the early stages. Faced with these facts, this work presents the development of an approach to clinical decision support based on semantics and reasoning. To this end, the current knowledge on clinical practice guidelines in nephrology was adopted. This approach seeks to assist the health professional in the early identification of kidney conditions, to support appropriate referral to specialists in nephrology and to help on the proper management of patients with chronic kidney disease.*

Resumo. *A doença renal crônica é uma lesão renal de evolução progressiva, insidiosa, assintomática e com alto índice de morbidade e mortalidade, além de possuir difícil diagnóstico nas fases iniciais. Diante desses fatos, este trabalho apresenta o desenvolvimento de uma abordagem de suporte à decisão clínica, baseada em semântica e raciocínio, utilizando o conhecimento presente nos conjuntos de diretrizes para prática clínica no domínio da nefrologia. Essa abordagem busca auxiliar o profissional da saúde na identificação precoce, dar suporte ao encaminhamento apropriado para os especialistas em nefrologia e auxiliar o manejo nefrológico adequado de pacientes com doença renal crônica.*

1. Introdução

Diante do domínio da Saúde, a Doença Renal Crônica (DRC) consiste em uma lesão renal, acompanhada de uma perda progressiva e irreversível da função dos rins (Glomerular, Tubular e Endócrina). Em sua fase mais avançada, os rins não conseguem mais manter a normalidade do meio interno do paciente, [Romão Junior, 2004] [K/DOQI, 2002]. A DRC foi escolhida como alvo da pesquisa por ser um problema de saúde pública, devido ao seu caráter progressivo é considerada, segundo o Ministério da Saúde (2014), uma das principais causas de morbidade e mortalidade em países desenvolvidos e em desenvolvimento, merecendo um novo olhar de todos os atores dos sistemas de saúde e dos envolvidos com a educação de profissionais da saúde, em especial, aqueles que atuarão na atenção primária, para o diagnóstico precoce e ações de prevenção desta doença cuja morbimortalidade é crescente, [Sodré e Oliveira, 2014], [Inker et al., 2014], [Bastos e Kirsztajn, 2011].

Outro fator motivacional para a escolha da área da Nefrologia como domínio de aplicação, está relacionado aos dados epidemiológicos sobre a DRC. De acordo com o Censo Brasileiro de Diálise publicado em 2013, o número de pacientes com DRC Terminal no Brasil praticamente duplicou na última década, passando de 42.695 em 2000 para 100.397 em 2013. Estimativas de 2011 revelam que cerca de 10 milhões de

brasileiros são portadores de DRC e 75% desconhecem o fato. [Saúde, 2014]. Por conta do desse crescente número de casos e da alta taxa de mortalidade nos estágios avançados da doença, tem sido impulsionado estudos relacionados à prevenção, diagnóstico precoce e melhorias no tratamento, [Sodré e Oliveira, 2014].

A identificação precoce e o manejo nefrológico adequado de pacientes com DRC moderada têm sido reconhecidos cada vez mais como importantes oportunidades de impedir a sua progressão e desfechos desfavoráveis, como a mortalidade. Como estas situações são geralmente de evolução progressiva, insidiosa e assintomática, há dificuldades em seu diagnóstico nas fases iniciais. Muitas vezes, a identificação de alterações da função renal é incidental, ou então ocorre através de testes laboratoriais e avaliações de rotina voltados para grupos de alto risco, por exemplo, como diabéticos e hipertensos.

Diante desses fatos, este estudo busca na Ciência da Computação mecanismos de aplicação para a gestão do conhecimento, visando o auxílio, a conscientização e vigilância nos cuidados primários e secundários à saúde, auxiliando no diagnóstico e encaminhamento precoce e apropriado ao nefrologista e na instituição de diretrizes apropriadas, retardando a progressão da DRC e prevenindo suas complicações. Para isso, esse trabalho propõe o desenvolvimento de uma abordagem de suporte à decisão clínica, baseada em semântica e raciocínio, utilizando o conhecimento presente nos conjuntos de diretrizes para prática clínica no domínio da nefrologia. Essa abordagem está relacionada ao diagnóstico, estadiamento e manejo clínico da DRC, buscando dar suporte desde o encaminhamento apropriado para os especialistas em nefrologia ao suporte na aplicação das devidas práticas clínicas.

2. Suporte à Decisão Clínica baseado em Representação do Conhecimento

Desde o primeiro momento em que foram utilizados computadores na área da Saúde, a perspectiva de que esses poderiam desempenhar um papel ativo, no sentido de ajudar a resolver problemas relacionados à tomada de decisões, estimulou muito interesse e entusiasmo dos pesquisadores e foi um dos motivadores primários para prosseguir com a utilização de computadores, [Greenes, 2007]. Abordagens computacionais de Suporte à Decisão têm como objetivo auxiliar tomadores de decisão a utilizar dados, modelos e estruturas de maneira que estes possam ajudar a resolver problemas e a tomar decisões, [Hennemann, 2006], ou seja, possuem o objetivo de auxiliar os profissionais envolvidos na tomada da decisão, a partir de situações de decisão semiestruturadas, a fim de estender suas capacidades sem substituir seus julgamentos, [Bohamec, 2001].

Observando que em todas as áreas profissionais o processo de tomada de decisão está presente e que em todas as situações é importante realizar a melhor escolha, o desenvolvimento de sistemas informatizados de representação e gerenciamento de dados em saúde foi fundamental para auxiliar a evolução da prática da medicina nas últimas três décadas. Entre as modalidades de aplicação de sistemas informatizados encontram-se abordagens capazes de auxiliar o profissional da saúde no apoio à decisão clínica, podendo ajudá-lo a formar uma segunda opinião do conhecimento médico especializado, [Musen et al., 2006].

Segundo Osheroff et. al., (2007), um Suporte à Decisão Clínica, do inglês *clinical decision support (CDS)*, deve fornecer informações certas, à pessoas certas, no

formato desejado, sem atrapalhar o fluxo de trabalho, buscando melhorar as decisões e os resultados nos cuidados de saúde. O CDS deve ser bem aceito pelos usuários finais e deve apoiar, em vez de diminuir, o fluxo de trabalho do usuário. Em seu trabalho, Osheroff et. al., (2007) também afirma que o CDS deve envolver técnicas que o façam ser bem aceito pelos usuários finais, e que sua utilização faça sentido.

Abordagens de CDS devem manipular conhecimento e informações de forma inteligente, sendo desenvolvidas para serem utilizadas em problemas que requerem uma quantidade considerável de conhecimento humano e especializado, como é o caso do manejo e diagnóstico clínico. Baseado nessas premissas, o uso de conceitos da Inteligência Artificial têm se destacado muito nesses últimos anos, entre eles o de Ontologias. Segundo o Mapeamento Sistemático sobre o uso de ontologias na Informática em Saúde, realizado por Mota et al. (2014), da maioria dos trabalhos que foram analisados, 13% utilizavam ontologias para auxiliar ou tornar sistemas de suporte a decisão mais eficientes. Isso ocorre porque, na Ciência da Computação, a Ontologia é uma técnica de Representação do Conhecimento que se refere a um artefato de engenharia constituído por um vocabulário específico usado para descrever uma determinada realidade, por uma série de suposições sobre o significado pretendido do vocabulário, [Breitman, 2005]. Russell e Norvig (2009) definem a ontologia como uma representação formal, explícita e consensual de conhecimento, passível de processamento automático, recuperação de informações e realização de mecanismos de inferência.

Com relação ao conhecimento que vai ser representado pela ontologia, segundo Greenes (2007), para que recursos de apoio à decisão possam trabalhar da melhor maneira, eles devem ser capazes de acessar um conhecimento médico baseado em evidências de alta qualidade, para isto é recomendado o uso de diretrizes de práticas clínicas. As diretrizes clínicas são declarações sistematicamente desenvolvidas por entidades de saúde para auxiliar os profissionais da saúde em decisões, fornecendo informações sobre os cuidados de saúde apropriados para cada circunstância clínica específica, buscando padronizar a prática clínica e melhorar os resultados. Em uma era da medicina baseada em evidências, as diretrizes estão se tornando uma das ligações críticas entre as melhores evidências disponíveis e a boa prática clínica, [RNAO, 2012]. Logo, sistemas de suporte à decisão clínica, desenvolvidos utilizando como base de conhecimento um conjunto de diretrizes para prática clínica, podem usufruir de um guia para fazer inferências com base nas decisões, ações e recomendações clínicas validadas, padronizar a prestação de cuidados em saúde e identificar quais os dados do paciente são realmente necessários e relevantes, [Greenes, 2007].

3. Trabalhos Relacionados

Através do trabalho de Patwardhan et al. (2009) é possível chegar a conclusão de que sistemas de suporte à decisão clínica têm um potencial enorme para facilitar melhorias na gestão da DRC. No entanto, eles não estão amplamente disponíveis nem são rotineiramente utilizados. Na doença renal, eles têm sido predominantemente utilizados para fornecer orientações sobre o uso de medicamentos em pacientes com função renal anormal, como é visto nos trabalhos de Galanter et al. (2005), Nash et al. (2005) e Oppenheim et al. (2002).

Na busca por abordagens existentes de sistemas de suporte à decisão clínica no domínio da doença renal crônica, este estudo identificou que atualmente existe um volume relativamente pequeno de literatura relacionada com aplicações de CDS na DRC, principalmente no Brasil, como também afirmam os trabalhos de Patwardhan et al. (2009) e Narva (2012).

Com relação ao uso de ontologias, no Mapeamento Sistemático de Mota et al. (2014) foi possível identificar que existe um grande número de publicações que abordam a construção e o uso de ontologias no desenvolvimento de Sistemas de Informação para o gerenciamento de informação e Suporte à Decisão na Informática em Saúde. Também identificou-se que um dos principais benefícios do uso de ontologias na área da saúde está relacionado a aquisição de conhecimento para subsidiar a tomada de decisão, [Hashmi e Zrimec, 2008] [Anyà et al., 2009] [Michin et al., 2006], [Saripalle, 2014] [Andrade, Almeida e Schulz, 2012], evitar erros médicos que surgem a partir da ausência de informação suficiente, [Anagnostakis, et. al., 2005], e orientar os profissionais da saúde na realização das práticas clínicas relacionadas ao tratamento de doenças [Abidi et al., 2012], [Colantonio et al., 2007]. Observou-se também que o uso de ontologias está permitindo a construção de sistemas de diagnóstico que auxiliam a representação de protocolos clínicos, a classificação e a especificação de doenças, através da extração de informações de banco de dados clínicos, diretrizes, livros didáticos e de páginas da web, bem como o compartilhamento dessas informações.

4. Abordagem proposta

Para que essa abordagem seja capaz de alcançar com um amplo impacto ao apoio à decisão clínica sobre a perspectiva de segurança e qualidade dos cuidados de saúde relacionados ao paciente, essa abordagem será fundamentada em metodologias validadas descritas nos trabalhos de Greenes (2007), Musen, Shahr e Shortliffe (2006) e Berner (2007), bem como as recomendações para a construção de abordagens de apoio à decisão clínica para o tratamento de indivíduos com doença renal crônica, descritas por Patwardhan et al. (2009). Na etapa de representação do conhecimento do domínio em questão, a solução encontrada foi realizar a composição do Método 101, proposto por Natalya Noy e McGuinness (2001), e da metodologia para criação de ontologias no ambiente da saúde, proposta por Lichtenstein et al. (2015). Composições de duas ou mais metodologias nessa área são reforçada no trabalho de Breitman (2005).

A primeira definição que deve ser realizada é a escolha do domínio onde a abordagem de CDS irá atuar, ou seja, a escolha do seu ambiente de aplicação e do seu objetivo. Foi então definido para essa abordagem o domínio da Nefrologia com objetivo de auxiliar profissionais da saúde, tanto da atenção primária quanto da atenção secundária, na resolução de problemas, tomada de decisões e ações, dando suporte ao encaminhamento apropriado para os especialistas em nefrologia. Logo, de acordo com o domínio escolhido, os usuários para quais essa abordagem de CDS se destinará serão os profissionais da saúde que atuam na área da nefrologia e estão relacionados a atenção primária na saúde. Observando os tipos de problemas e desafios identificados e descritos anteriormente, foi definido que essa abordagem deverá dar suporte através de alertas, chamando a atenção do usuário para um fato, e através de recomendações, indicando os passos e fornecendo informações sobre os procedimentos.

Para guiar e reforçar as escolhas durante o desenvolvimento da abordagem, foi realizado um levantamento de pressupostos gerais sobre o contexto no qual essa abordagem de CDS irá atuar. O levantamento desses pressupostos foi realizado através de pesquisas de campo, análise de trabalhos como os de Klebe et al. (2006, 2007), e RNAO (2012), com temas voltados para validação de sistemas de apoio à decisão clínica e entrevistas com profissionais da saúde. Entre os problemas encontrados, os mais impactantes são: profissionais da saúde não adotarão novas tecnologias a menos que, seja financeiramente vantajoso (por exemplo, não necessitem realizar pagamentos para utilizar uma ferramenta) ou que traga uma contribuição clara à sua prática de trabalho convencional, ou seja, ferramenta de fácil usabilidade, e abordagens simples e focadas apenas em diagnóstico auxiliado por computador são, na verdade, raramente utilizadas na prática. O diagnóstico deve ser uma das funcionalidades que a abordagem deva oferecer.

4.1 Representação do Conhecimento

Para compor a base de conhecimento dessa abordagem, será proposta a construção de uma ontologia de domínio chamada OntoDRC_DEEM (Diagnóstico-Estadiamento-Encaminhamento-Manejo Clínico), que buscará representar os conceitos clínicos subjacentes e os relacionamentos inerentes dentro das diretrizes de práticas clínicas relacionadas ao diagnóstico, estadiamento e manejo clínico da doença renal crônica, como também sobre quando devem ser realizados os encaminhamentos ao especialistas em Nefrologia.

O primeiro passo para a construção da ontologia foi a definição do seus domínios e seus escopos.

- Informações sobre os passos que devem ser seguidos para realizar o diagnóstico e estadiamento da DRC;
- Informações sobre quando deve-se encaminhar o paciente para os especialistas em Nefrologia;
- Informações sobre o manejo clínico de pacientes com doença crônica renal.

Foi considerado o reuso de outras ontologias, buscando vantagens como reduzir o tempo de desenvolvimento de ontologias, diminuir o risco de desenvolvimento (já que se está reutilizando conceitos que foram testados por outras aplicações) e incentivar à continuidade na atualização e na adição de novos conceitos na ontologia. Foram realizadas buscas em repositórios que disponibilizam ontologias eletronicamente através da internet. Também foram selecionados alguns engenhos de busca, por possuírem, segundo a literatura pesquisada, boa parte das pesquisas desenvolvidas no domínio da Ciência da Computação, [Mota et al., 2014]. No domínio da Nefrologia, foram encontradas a CD-EMR Ontology, desenvolvida por Zaidi (2012), e a CHRONIOUS Chronic Kidney Disease Ontology, proposta por Schneider e Brochhausen (2011). Apesar de ambas não possuírem especificação do conhecimento suficiente para atender aos propósitos definidos nesse trabalho, as especificações de modelagem e validação podem auxiliar no desenvolvimento da representação do conhecimento aqui proposta.

Para realizar o levantamento dos termos da ontologia OntoDRC_DEEM foram utilizadas técnicas tradicionalmente aplicadas por engenheiros de requisitos no processo de eliciação de informações. Para essa abordagem foram escolhidas como técnicas de

elicitação a leitura dinâmica de documentos, realização de entrevistas e aplicação de questionários aos médicos.

A primeira técnica de elicitação que essa abordagem realizou foi uma busca por livros e diretrizes de práticas clínicas no domínio da DRC. Nesses artefatos foram realizadas extrações de informações sintetizadas, essenciais para o propósito da ontologia. Nessa técnica as informações extraídas foram descritas em forma de axiomas formais e expressões lógicas, para facilitar a identificação de termos candidatos a serem representados como classes e suas propriedades e restrições. As informações extraídas foram revisadas e validadas com a ajuda de uma médica nefrologista.

A segunda técnica de elicitação foi a realização de entrevistas e aplicação de questionários com médicos. Com relação as entrevistas, foi estabelecida uma relação direta entre pesquisador e quatro médicos da atenção primária e um da atenção secundária, realizando o registro das respostas em anotações manuais. Com relação aos questionários, levando em consideração o ambiente conturbado em que trabalham os profissionais da saúde, foi elaborado contendo somente questões essenciais e de múltiplas escolhas, sendo disponibilizado em forma impressa e online. Apesar dessa etapa estar em andamento, já foram recolhidos e sintetizados dados suficientes para, unidos a sumarização realizada na primeira técnica, dar início a produção da ontologia, sendo possível identificar preliminarmente os termos e as informações relevantes para compor o seu conhecimento.

Com base no conhecimento extraído na primeira técnica, foram desenvolvidos fluxogramas, descrevendo elementos principais e o fluxo correto de execução das práticas relacionadas ao manejo clínico de pacientes com DRC. Esses fluxogramas foram validados por médicos nefrologistas. Após a realização desses métodos de aquisição de conhecimento, foi definido preliminarmente as classes e a suas hierarquias, bem como suas propriedades. Existem várias estratégias para se definir uma hierarquia, porém, para essa abordagem foi escolhido a estratégia topo-para-baixo (top-down), que se inicia ao definir os conceitos mais gerais e continua seguindo um processo de decomposição, onde abaixo do termo mais abrangente é colocado os termos mais específicos.

Para a definição e formalização das informações extraídas nas técnicas de elicitação, foi realizada a estratégia de definir as propriedades das classes como variáveis de decisão e variáveis de ação, que estarão presentes no conhecimento representado pela ontologia. Este fato irá auxiliar na construção e na definição dos métodos de raciocínio para suporte à decisão nas práticas clínicas. Por exemplo, dada a diretriz clínica: "*Quando um paciente apresenta taxa de filtração glomerular < 60 e já foi avaliado há 3 meses, então deve ser diagnosticada a DRC*". As informações deverão ser definidas em termos de regra de decisão através das seguintes etapas:

Etapa 1 - Identificar termos relevantes ao domínio e classificá-los como candidatos às classes da ontologia. Por exemplo na frase anterior, esses termos seriam: "*paciente*", "*taxa de filtração glomerular < 60*", "*3 meses*" e "*DRC*".

Etapa 2 - Selecionar as variáveis de decisão, que representam as premissas da regra. Por exemplo, através da relação existente entre as entidades: "*paciente*", "*taxa de filtração glomerular < 60*", "*3 meses*" e "*DRC*", e define-se como variáveis de decisão: "*apresenta*" e "*avaliado há*".

Etapa 3 - Selecionar qual será a variável de ação que representa a conclusão da regra. No caso do exemplo, a variável de ação será: “*avaliar para*”.

Etapa 4 – Definir a regra baseada nas variáveis selecionadas. No caso do exemplo, a regra geral seria:

SE Paciente APRESENTA TFG <60 ^ O Paciente FOI AVALIADO HÁ 3 meses ENTÃO DEVE SER AVALIADO DRC.

A saída do modelo de decisão do CDS será para executar alguma ação, ou para fazer uma recomendação. Por exemplo: avaliar para DRC, indicar qual estágio da DRC o paciente se encontra, recomendar refazer o exame após 3 meses, etc. Neste ponto da engenharia do conhecimento, afirma Russell e Norvig (2009), formular consultas ao procedimento de inferência e obter respostas, é a etapa em que está a recompensa, podendo deixar o procedimento de inferência operar sobre os axiomas e fatos específicos do problema para derivar os fatos que o módulo de raciocínio busca conhecer.

A ontologia encontra-se em fase de implementação, para esse passo está sendo utilizada a ferramenta *Protégé*, uma ferramenta livre desenvolvida em Java, que permite a criação e edição de Ontologias, incluindo o formato OWL. Esta ferramenta possui um mecanismo de inferência (*reasoner*) baseado em lógica descritiva para verificar a consistência da ontologia e para computar a hierarquia das classes. Ao final, será realizada a documentação e o compartilhamento para permitir que modificações e reuso sejam possíveis no futuro. Para a etapa de validação, está sendo realizada uma busca por bases de dados, de uso permitido, contendo informações sobre casos clínicos pré-diagnosticados para servirem de comparação, onde serão realizados cálculos de acurácia, precisão e exatidão.

5. Considerações Finais

Este trabalho apresentou os primeiros passos para o desenvolvimento de uma abordagem de suporte à decisão clínica, baseada em semântica e raciocínio, relacionado ao diagnóstico e estadiamento da DRC, buscando dar suporte ao encaminhamento apropriado para os especialistas em Nefrologia. Durante o trabalho, foi constatado que atividades de modelagem são essenciais para que uma abordagem possa abranger todos os requisitos necessários para dar um suporte à decisão de qualidade aos seus usuários. Deve-se lembrar e ressaltar que não existe uma maneira correta ou única de se modelar um domínio, sempre existirá várias alternativas. A melhor solução dependerá da aplicação e das possíveis extensões. O desenvolvimento de ontologias não é um processo linear. Muitas interações e refinamentos são necessários para chegar a um modelo adequado, sendo possível que novas modificações tenham que ser incorporadas à ontologia, fazendo com que as fases iniciais tenham que ser revisitadas.

Também conclui-se que a avaliação e validação devem ser parte integrante da metodologia de sistemas, ao invés de ser apenas uma atividade para ser realizada futuramente após a implementação do sistema. Similarmente ao que se observa na engenharia de software, o desenvolvimento de abordagens de suporte à decisão clínica exigem constantes ciclos de revisões e manutenções.

Por fim, como visto nos trabalhos de Greenes (2007) e Berner (2007), abordagens como a desse trabalho podem ajudar na redução dos erros e negligenciamentos relacionados ao tratamento médico, além de fornecer uma grande variedade de outros benefícios, o que indiretamente, é claro, amplia a segurança e qualidade dos cuidados de saúde do paciente, aumentando a eficácia dos tratamentos e diagnósticos. Esse trabalho também poderá futuramente auxiliar como guia para métodos relacionados ao desenvolvimento de sistemas que buscam melhorar a qualidade dos cuidados de saúde.

Referências

- ABIDI, Syed Sibte Raza et al. Using OWL Ontologies for Clinical Guidelines Based Comorbid Decision Support. In: HAWAII INTERNATIONAL CONFERENCE ON SYSTEM SCIENCES, 45., 2012, Maui. Proceedings of Hawaii International Conference on System Sciences. Maui: [s. N.], 2012. p. 3030 - 3038.
- ANAGNOSTAKIS, A. G.; Tzima, M.; Sakellaris, G. C.; Fotiadis, D. I.; Likas, A. C.; Semantics-based information modeling for the health-care administration sector: the citation platform. IEEE transactions on information technology in biomedicine: a publication of the IEEE Engineering in Medicine and Biology Society, v. 9, n. 2, p. 239-47, jun. 2005.
- ANDRADE, A; ALMEIDA, M; SCHULZ S. . Revisiting ontological foundations of the OpenEHR Entry Model. KR-MED Series CEUR Workshop Proceedings. 2012; 897: -3rd International Conference on Biomedical Ontology (ICBO 2012); JUL 21-25, 2012; Graz, Austria
- ANYA, Obinna et al. E-Workbench: A Case for Collaborative Decision Support in e-Health. In: INTERNATIONAL CONFERENCE ON COMPUTER MODELLING AND SIMULATION, 11., 2009, England. Proceedings of International Conference on Computer Modelling and Simulation.England: [s. N.], 2009. p. 25 - 27.
- BASTOS, Marcus Gomes and KIRSZTAJN, Gianna Mastroianni. Doença renal crônica: importância do diagnóstico precoce, encaminhamento imediato e abordagem interdisciplinar estruturada para melhora do desfecho em pacientes ainda não submetidos à diálise. J. Bras. Nefrol. [online]. 2011, vol.33, n.1, pp. 93-108. ISSN 0101-2800.
- BERNER, Eta S., Clinical Decision Support Systems – Theory and Practice, Springer, 2007.
- BOHAMEC, M. What is Decision Support. In: 4th International Multi-conference Information Society, out. 2001, v. A, p. 86-89.
- BREITMAN, Karin. Web Semântica: a Internet do Futuro. Rio de Janeiro: Ltc – Livros Técnicos e Científicos Ltda, 2005.
- COLANTONIO, S., MARTINELLI, M., MORONI, D., SALVETTI, O., PERTICONE, F., SCIACQUA, A., GUALTIERI, A.: An Approach to Decision Support in Heart Failure. In: Semantic Web Applications and Perspectives (Swap 2007). Proceedings, Bari Informatica, Università di Bari (2007)

- GALANTER WL, DIDOMENICO RJ, POLIKAITIS A: A trial of automated decision support alerts for contraindicated medications using computerized physician order entry. *J Am Med Inform Assoc* 12: 269–274, 2005.
- GREENES, Robert A., *Clinical Decision Support: The Road Ahead*, Academic Press, 2007.
- HASHMI, Z.I., ZRIMEC, T. (2008). Ontology-driven Modeling of Clinical Practice Guidelines (CPG) Towards Computerization of CPG knowledge sources via GEM Model. *IEEE International Symposium on Information Technology, ITSIM* , 26th - 9th August, Malaysia..
- HENNEMANN, F.A.; RABELO, R.J.; SANTOS, J.V.C.; CURY, J.E.R. Um Sistema de Apoio à Decisão formado por Redes de Petri, Simulação e Sistema Especialista. *Revista Controle & Automação*, v. 17, p. 10 a 23, 2006.
- INKER, Lesley A. et al. KDOQI US Commentary on the 2012 KDIGO Clinical Practice Guideline for the Evaluation and Management of CKD. *American Journal Of Kidney Diseases*. Usa, p. 713-735. set. 2014.
- K/DOQI clinical practice guidelines for Chronic kidney disease: evaluation, classification and stratification. *Am. J. Kidney Dis.*, v. 39, suppl. 1, p.s1-s266, 2002.
- KLEBE B, Hobbs H, Irving J, Cooley R, Stevens P, O'Donoghue D, Middleton R, Farmer C: A computer-based decision support tool to manage CKD in primary care. Presented at the American Society of Nephrology Annual Meeting, San Diego, CA, November 14-19, 2006
- KLEBE, Bernhard et al. *Kidney disease management in UK primary care: guidelines, incentives and Information Technology*. Oxford University Press: Family Practice Advance Access. Kent, p. 330-335. 24 abr. 2007.
- LICHTENSTEIN, Flávio et al. Criando uma Ontologia em Saúde com a ferramenta Protégé no padrão OWL. Disponível em: <<http://www.sbis.org.br/cbis11/arquivos/992.pdf>>. Acesso em: 13 jan. 2015.
- MICHIN, R.; PORTO, F. A. M.; VANGENOT, C.; HARTMANN, S.; Symptoms Ontology for Mapping Diagnostic Knowledge Systems. In: *Computer-Based Medical Systems*, p. 593 – 598, 2006.
- MOTA, M. R. A. ; LINO, N. C. Q. ; TAVARES, E. A. ; CAMPOS, S. P. R. . Um Mapeamento Sistemático sobre o Uso de Ontologias em Informática Médica. In: *CBIS 2014 - XIV Congresso Brasileiro de Informática em Saúde*, 2014, Santos. *Anais CBIS 2014 - XIV Congresso Brasileiro de Informática em Saúde*, 2014.
- MUSEN MICHIN, R.; PORTO, F. A. M.; VANGENOT, C.; HARTMANN, S.; Symptoms Ontology for Mapping Diagnostic Knowledge Systems. In: *Computer-Based Medical Systems*, p. 593 – 598, 2006.
- NARVA, Andrew S.. Decision Support and CKD: Not There Yet. *Clinical Journal Of The American Society Of Nephrology*. Usa, p. 525-526. abr. 2012.
- NASH IS, ROJAS M, HEBERT P, MARRONE SR, COLGAN C, FISHER LA, CALIENDO G, CHASSIN MR: Reducing excessive medication administration in hospitalized adults with renal dysfunction. *Am J Med Qual* 20: 64–69, 2005.

- NOY, NATALYA.; MCGUINESS, D. Ontology Development 101 - A guide to creating your first ontology. KSL Technical Report, Stanford University, 2001
- OPPENHEIM MI, VIDAL C, VELASCO FT, BOYER AG, COOPER MR, HAYES JG, FRAYER WW: Impact of a computerized alert during physician order entry on medication dosing in patients with renal impairment. Proc AMIA Symp: 577–581, 2002
- OSHEROFF JA, TEICH JM, MIDDLETON B, STEEN EB, WRIGHT A, DETMER DE: A roadmap for national action on clinical decision support. J Am Med Inform Assoc 14: 141–145, 2007.
- PATWARDHAN, Meenal B. et al. Recommendations for a Clinical Decision Support for the Management of Individuals with Chronic Kidney Disease. Clinical Journal Of The American Society Of Nephrology : Cjasn: American Society of Nephrology, Usa, v. 4, n. 2, p.273-283, fev. 2009. Disponível em: <http://www.ncbi.nlm.nih.gov/pmc/articles/PMC2637586/#__abstractid290419title>. Acesso em: 13 jan. 15.
- RNAO, Registered Nurses' Association of Ontario (Org.). Toolkit: Implementation of best practice guidelines. 2. ed. Toronto: [s. N.], 2012. 154 p
- ROMÃO JUNIOR, João Egidio. Doença Renal Crônica: Definição, Epidemiologia e Classificação. Jornal Brasileiro de Nefrologia. São Paulo, p. 1-3. set. 2004. Disponível em: <http://www.jbn.org.br/detalhe_artigo.asp?id=1183>. Acesso em: 13 dez. 2014.
- RUBIN, D.L.; GENNARI, J.H.; MUSEN, M.A.; Knowledge Representation and Tool Support for Critiquing Clinical Trial Protocols. In: Proceedings of the AMIA Symposium 2000.
- RUSSELL, Stuart; NORVIG, Peter. Artificial Intelligence: A Modern Approach. 3rd Edition. New Jersey: Prentice Hall, 2009.
- SARIPALLE, Rishi Kanth. Current status of ontologies in Biomedical and Clinical Informatics. Disponível em: <<http://www.engr.uconn.edu/~steve/Cse300/saripalle.pdf>>. Acesso em: 28 jan. 2014.
- SAÚDE, Ministério da. Diretrizes Clínicas Para O Cuidado Ao Paciente Com Doença Renal Crônica – DRC No Sistema Único De Saúde. 2014. É permitida a reprodução parcial ou total desta obra, desde que citada a fonte e que não seja para venda ou qualquer fim comercial. Disponível em: <http://bvsms.saude.gov.br/bvs/publicacoes/diretrizes_clinicas_cuidado_paciente_renal.pdf>. Acesso em: 13 jan. 2015.
- SCHNEIDER, Luc; BROCHHAUSEN, Mathias. The CHRONIOUS Ontology Suite: Methodology and Design Principles. International Conference On Biomedical Ontology, Usa, v. 1, n. 1, p.167-173, 28 maio 2011.
- SODRÉ, Aline Binotto; OLIVEIRA, Mauren Isfer Anghebem. Estimativa da Taxa de Filtração Glomerular Através de Fórmulas. Newslab - A Revista do Laboratório Moderno. São Paulo, p. 58-69. ago. 2014.
- ZAIDI, Syed Ali Haider. An Owl Ontology For Modeling HL-7 Compliant Electronic Patient Records For Chronic Disease Management. 2012. 373 f. Dissertação (Mestrado) - Curso de Health Informatics, Dalhousie University, Nova Scotia, 2012.

Simulação de VANETs utilizando ferramentas Omnet++, SUMO e VEINS

Jair José Ferronato¹, Marco Antônio Sandini Trentin²

¹Instituto Federal Sul-Rio-Grandense (IFSul)
Câmpus Passo Fundo - Estrada Perimetral Leste, 150 - Passo Fundo-RS - CEP: 99064-440

²Universidade de Passo Fundo (UPF)
BR 285, São José - Passo Fundo-RS - CEP: 99052-900

jair.ferronato@passofundo.ifsul.edu.br, trentin@upf.br

Abstract. *Vehicular networks - VANETs are a network category using wireless communication and communicate in an adhoc way between vehicles or with a communications infrastructure present along the public roads in order to access the Internet or use security applications or entertainment . The use of VANETs simulation tools enables to predict the behavior and performance of various components through the results generated. This article aims to present the VEINS framework that can assist in network level and traffic simulations, combining to metrics application and computational techniques used in vehicle networks to analyze the traffic infrastructure.*

Resumo. *As redes veiculares VANETs são uma categoria de rede que utilizam comunicação wireless e comunicam-se de forma adhoc entre veículos ou com uma infraestrutura de comunicação presente ao longo das vias públicas, a fim de ter acesso à Internet ou utilizar aplicações de segurança ou entretenimento. O uso de ferramentas de simulação de redes veiculares possibilita prever o comportamento e desempenho de vários componentes através dos resultados gerados. Este artigo tem por objetivo apresentar o framework VEINS, o qual pode auxiliar em simulações a nível de rede e de tráfego, aliando-se a aplicação de métricas e técnicas computacionais utilizadas em redes veiculares para análise da infraestrutura de trânsito.*

1. Introdução

As cidades, em qualquer lugar do mundo, necessitam cada vez mais de mobilidade urbana, no entanto, é preciso criar uma infraestrutura que atenda essa necessidade. Segundo [DENATRAN 2008], a indústria automotiva brasileira pretende aumentar em 62% a taxa de motorização até 2020. A intenção é passar dos atuais 154 para 250 veículos por 1.000 habitantes, de acordo com estimativa da Associação Nacional dos Fabricantes de Veículos Automotores (ANFAVEA).

O [IBGE 2010] estimou que a frota do Estado do Rio Grande do Sul aumentou de 2.631.035, em 1997, para 4.205.326 em 2007, o que significa um aumento de 62% em 10 anos. Neste mesmo período, a população no Estado cresceu de 9.637.682 habitantes para 10.582.840, aumentando 9%. A frota atual do Estado é de 5.514.581 veículos, o que equivale a 1,91 pessoas por veículo.

A cidade do futuro poderá ter melhores condições de trafegabilidade com a utilização de Sistemas Inteligentes de Trânsito (SIT). Isso poderá ser possível, por exemplo, com envio de informações de sensores espalhados por pontos estratégicos para alimentar centrais de monitoração pró-ativa de trânsito e com isso gerar informações estratégicas.

Os SIT são implantados geralmente em grandes cidades e demandam altos investimentos, no entanto, percebe-se a oportunidade de disponibilizá-los em cidades de porte médio com custos acessíveis. Um desses centros é o Município de Passo Fundo, que possui cerca de 200 mil habitantes, sendo parte destes flutuantes, os quais buscam na cidade um local para compras, tratamentos de saúde ou estudos. Considerada uma cidade de grande potencial de utilização de um SIT, pois segundo [DENATRAN 2008], em Passo Fundo existem cerca de 110 mil veículos em circulação para uma população de 185.000 pessoas, o que equivale a 1,68 pessoas por veículo, revelando uma grande frota de veículos e acima da média estadual.

Diante de muitos problemas e oportunidades com o crescimento exponencial da quantidade de veículos nas cidades, pode-se identificar como potencialidades de estudo: redes de sensores para fluxo de trânsito adaptativo; links de comunicação entre centrais semafóricas e sensores por meio de comunicação sem fio; segurança dos dados e transações distribuídas; geração de informações estratégicas para a contagem de veículos em determinado local; e geração de informações da situação atual do trânsito com base na recepção de dados de sensores e sinaleiras. Também a comunicação veicular através de redes VANETS (*Vehicular Networks*) envolvendo troca de dados entre veículos V2V (*Vehicle to Vehicle*) e veículos com a infraestrutura de trânsito V2I (*Vehicle to Infrastructure*), bem como todas as formas de conectividade para utilização destas tecnologias.

2. VANETs

As VANETs permitem usar diversos dispositivos de conectividade veicular. Verifica-se uma ascensão do interesse e pesquisas em relação às comunicações entre veículos. Devido à necessidade de uma variedade de serviços e aplicativos que viabilizem a comunicação entre veículos, são considerados relevantes os estudos sobre as VANETs. Uma rede VANET é constituída por veículos em redes *ad hoc* que usam conexões sem fio diretamente entre os veículos. Estas redes são caracterizadas pela auto-organização e auto-gerenciamento pelos nodos da rede.

Segundo [Rashvand 2013], uma rede VANET é considerada uma rede dinâmica e que apresenta impactos na robustez da rede, por exemplo, conforme a velocidade utilizada por cada veículo e a taxa de deslocamento em áreas urbanas é possível utilizar diversas formas de conexão para estas redes. Cada veículo é equipado com dispositivos de comunicação e interação sem a necessidade de estações-base. As alternâncias de velocidades causam instabilidade nas topologias e também complexidade no roteamento e transmissão de mensagens.

Uma das contribuições mais importantes das redes veiculares é propiciar a segurança no trânsito. Por meio da comunicação entre os veículos, é possível que um motorista seja alertado, enquanto trafega por uma via, da ocorrência de acidentes à sua frente ou das condições da via. Além disso, permite localização, através de dados do GPS, velocidade e direção com a rede em que está conectado.

A Figura 1 representa um cenário uso de VANETs para o trânsito. Observa-se que a comunicação ocorreu devido a um acidente, onde os veículos são avisados por mensagens e utilizam outra via para locomoção, evitando congestionamentos em caso de obstrução da pista.

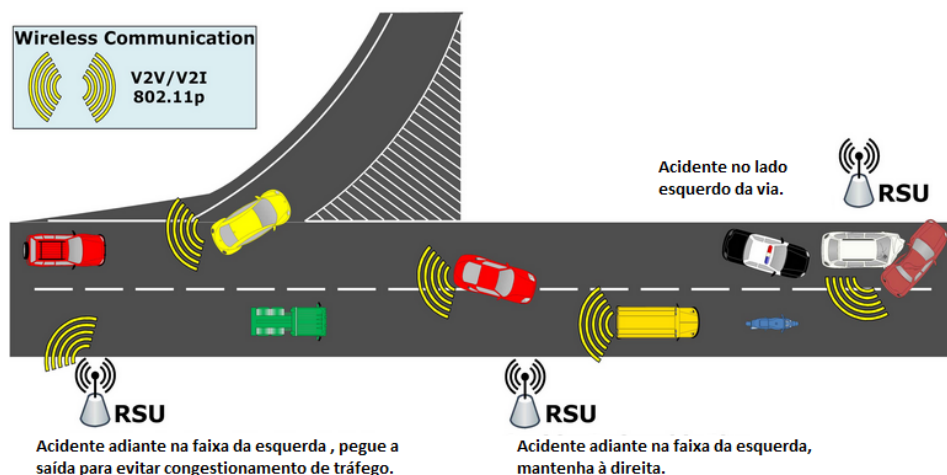


Figura 1. Cenário de uso de uma VANET. Adaptado de [Atul B.Kathole 2013]

Além de prevenir acidentes, as VANETs são úteis para informar aos motoristas sobre o trânsito, obras na pista, tempo de chegada ao destino, entre outras informações que auxiliam durante o percurso.

Também através das VANETs pode-se implementar sistemas para o transporte público e facilitar a troca de informações entre usuários e ônibus com mensagens de horários de chegada e posição atual dos coletivos em tempo real. Para isso, os ônibus devem trocar mensagens entre si e com os pontos fixados em cada local da infraestrutura de trânsito.

Para que as mensagens, alertas e toda comunicação seja realmente efetiva numa VANET, percebe-se a importância da simulação do fluxo veicular para análises e projetos de implementação de mudanças nas rodovias e centros urbanos.

A simulação de condições de tráfego são importantes para prever o comportamento de situações atípicas em fluxos de trânsito veicular. Os simuladores permitem através de aplicações computacionais gerar cenários reais e também facilitam testes de modificações para análise de viabilidade de mudanças.

3. Simuladores de VANETs

Um simulador auxilia os pesquisadores a criar verificações técnicas e métricas sem a necessidade de execução de testes em campo. A manipulação de validações pode ser inviável em determinadas situações tais como aquisição e uso de equipamentos com custo elevado e geralmente não é disposto tempo para criar uma estrutura eficiente de testes. Com isso, os simuladores são aliados a nível ferramental e com a função de criar ambientes o mais próximo possível àqueles encontrados na vida real.

Existem vários simuladores disponíveis no meio acadêmico e também comercial.

Conforme [Martinez et al. 2011], existem várias ferramentas de simulação, as quais podem ser agrupadas diferentes grupos como: Simulador de tráfego, simuladores de redes e software integrador de VANETs, os quais permitem integração através de entradas pré-definidas e troca constante de dados entre as ferramentas.

O fluxo de tráfego gerado pelo simulador pode utilizar dados reais de trânsito e de mobilidade, os quais podem ser usados como entrada em simuladores de redes. Assim, é possível calcular e criar componentes em redes wireless com detalhadas estruturas de todos os nodos que enviam e recebem pacotes, canais de tráfego e transmissão de sinais.

Segundo [Noori 2012], simular uma VANET envolve dois diferentes aspectos: Primeiro, há questões relacionadas com a comunicação entre os veículos. Simuladores de rede permitem entender problemas de comunicação através da análise específica dos protocolos de rede. O segundo aspecto está relacionado com a mobilidade dos nodos da VANET. Assim, um modelo de mobilidade, associado a um simulador de tráfego, é de grande valia para a reproduzir ambientes complexos tanto em locais urbanos como auto estradas. A Figura 2 apresenta a classificação dos simuladores de VANETs.

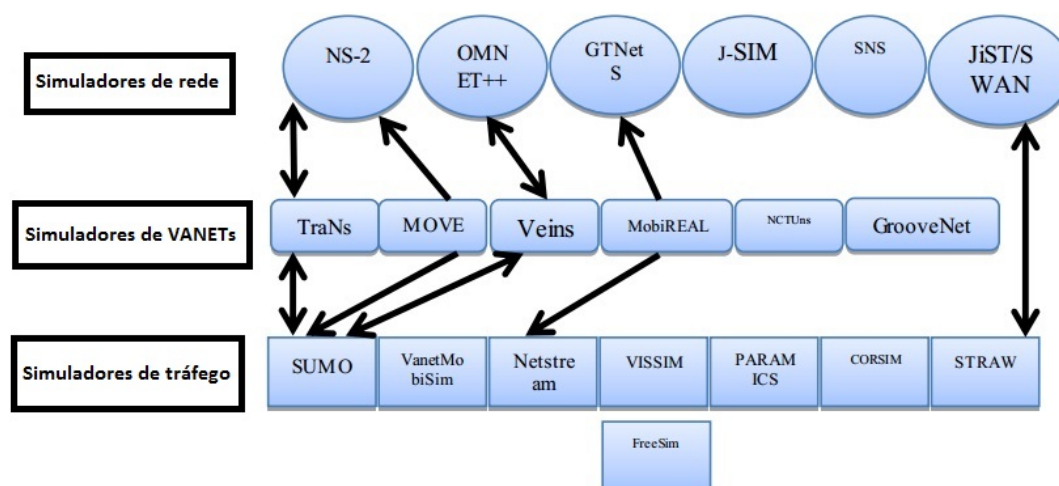


Figura 2. Classificação de simuladores veiculares. Adaptado de [Noori 2012]

3.1. Modelos de mobilidade

O conhecimento sobre as características das movimentações veiculares e a identificação de padrões nos movimentos dos veículos tornam-se importantes para propor alternativas nos modelos de locomoção existentes, ou, até mesmo, criar novos modelos de mobilidade que se adaptem melhor ao representar uma VANET. Um modelo de mobilidade busca auxiliar na criação de cenários mais próximos da realidade possível, e utilizam características específicas usadas na locomoção de veículos.

Os padrões de mobilidade dos nodos de uma VANET informam ao simulador um comportamento de movimentação dos nodos semelhantes aos do movimento de veículos em uma situação real. Nodos estáticos ou com baixa locomoção tendem a manter-se na mesma topologia e o roteamento entre os nodos permanece estável. Em contrapartida, os nodos com alta mobilidade podem causar frequentes perdas de pacotes, sendo esta uma característica muito comum em deslocamentos de veículos em uma via, seja em ambientes urbanos ou de auto estrada.

3.2. Simulador de redes Omnet++

Analisar uma rede é uma atividade complexa pois exige equipamentos e softwares específicos com intuito de montar o cenário, compreender o seu comportamento, seus protocolos e vulnerabilidades. A área de simulação de redes permite compreender, projetar e criar soluções eficientes na resolução de problemas dificilmente detectados sem o uso de ferramentas.

O *framework* OMNET++ (*Objective Modular Network Testbed in C++*) é um simulador de eventos modular orientado a objetos geralmente usado para construir simuladores de rede. A motivação do desenvolvimento do OMNet++ foi produzir um simulador de eventos discretos que pudesse ser usado no meio acadêmico, educacional e mesmo em pesquisas. Ele funciona como um simulador discreto e baseado em eventos de comunicação entre nodos em uma ampla variedade de plataformas, o qual está tornando-se mais popular no campo de simulação de rede. [Sommer et al. 2011]

3.3. Simulador de tráfego veicular SUMO

As simulações de tráfego visam auxiliar na avaliação de mudanças de infraestrutura, bem como as alterações de políticas de mobilidade antes de implementá-los na prática. Por exemplo, a eficácia dos ambientes de tráfego ou algoritmos de controle de semáforos podem ser testados e aperfeiçoados em simulações antes de serem implantados no mundo real.

O SUMO (*Simulation of Urban MObility*) é uma suíte de simulação livre e aberta para simular o tráfego veicular. Ele é implementado em C++ e usa apenas as bibliotecas portáteis. Este simulador foi desenvolvido pelo Instituto do Sistema de Transporte no Centro Alemão Aeroespacial. A simulação é feita de forma microscópica, onde é possível identificar veículos, pedestres, transporte público, e também é possível simular a temporização dos semáforos. Sua aplicabilidade é ampla e envolve a avaliação de algoritmos modernos e planos de sinalização semafórica, facilita a análise de emissão de poluentes e permite analisar escolhas autônomas de rotas. O SUMO já foi usado na visita do papa na cidade de Colônia e durante a copa do mundo de 2006. A ferramenta também já foi usada para simular o comportamento da telefonia baseada em GSM [SUMO 2001].

Os recursos do SUMO são ricos em ferramentas de apoio que cuidam de tarefas tais como encontrar rota, visualização de intersecções, troca de redes e cálculo das emissões de Co2. O simulador pode ser ampliado com modelos personalizados e fornece várias interfaces para controlar remotamente a simulação.

3.4. Integração das ferramentas OMNET++ e SUMO

A integração das ferramentas de simulação é importante visto o nível de complexidade destes ambientes e permite obter resultados completos de situações específicas.

A Figura 3 apresenta um mecanismo para integrar simuladores de redes como Omnet++ e também simuladores de tráfego como o SUMO. No entanto, é possível também integrar os mapas gerados através do OpenStreetMaps¹ e gerar arquivos de redes e rotas veiculares com a utilização de parâmetros reais das rodovias. O TRACI é um componente escrito em linguagem Python que habilita um *socket* TCP para a integração entre

¹OpenStreetMaps - www.openstreetmaps.org

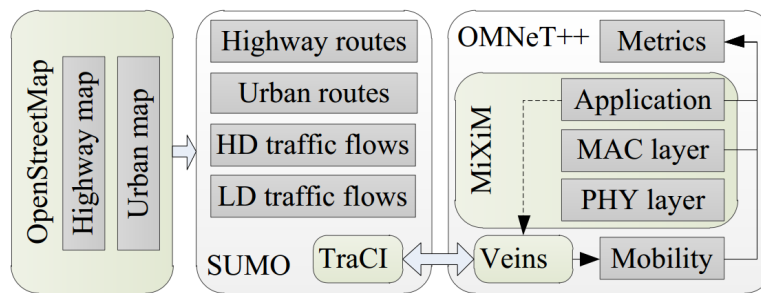


Figura 3. Integração de ferramentas de simulação. Fonte [J. 2012]

as ferramentas SUMO e o Omnet++. Dessa forma utiliza-se a integração basicamente em três etapas: Conversão de mapas, integração com os recursos de simulação de tráfego do SUMO com parâmetros de mobilidade e acoplamento com os mapas gerados e também integra a estrutura com o simulador Omnet++ para gerar análises a nível de rede.

A Figura 4 apresenta um exemplo gráfico de simulação das ferramentas SUMO e Omnet++. Pode-se observar que a simulação de tráfego à direita no SUMO reflete a realidade da quantidade de veículos em movimentação no Omnet++ à esquerda. Isso também é utilizado no sentido inverso com a alteração da quantidade de veículos no Omnet++ e posterior visualização no simulador de tráfego SUMO.

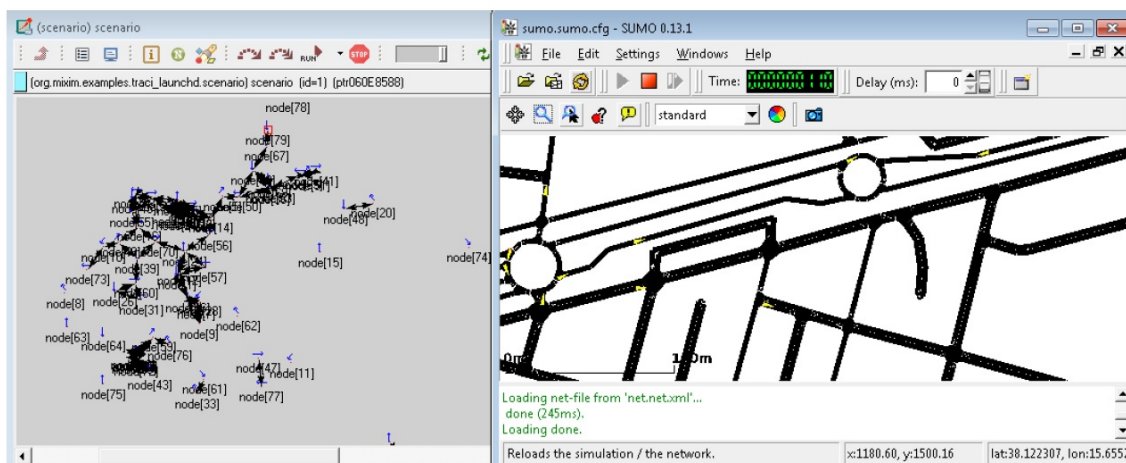


Figura 4. Resultado da integração de simuladores Omnet++ e SUMO. Fonte [J. 2012]

3.5. Simulador veicular VEINS

O framework VEINS (*Vehicular in Network Simulation*) [Sommer et al. 2011], é composto de um conjunto de modelos de simulação de redes veiculares de forma realística sem comprometer o desempenho dos testes. O *framework* VEINS, conforme Figura 4, faz a interface gráfica entre o Omnet++ e o SUMO, assim, permite rapidamente parametrizar e criar simulações de trânsito. Esta ferramenta permite acoplamento bidirecional e, com isso, facilita a simulação de redes VANETs e permite simular tráfego viário, e também permite modelar interações complexas entre cenários examinados.

4. Estudo de caso com o uso de simuladores para VANETs

A fim de avaliar as ferramentas acima descritas, foi realizado um estudo de caso composto por ambiente urbano especificamente da cidade de Passo Fundo. Os parâmetros do experimento estão descritos na Tabela 1:

Tabela 1. Configuração da simulação

Área simulada	2.500m x 2.500m
Tempo de simulação	600s
Veículos	200
Velocidade	14m/s
Alcance do rádio	300m
Protocolo MAC	802.11p
Frequência da antena	5.89MHZ
Taxa de transferência	18Mbps
Protocolos de roteamento	AODV e DYMO
Tamanho do pacote de dados	512 bytes
Acidentes simulados	1
Início do acidente	100s
Duração do acidente	50s

O modelo de mobilidade utilizado neste trabalho é definido pelo módulo associado ao VEINS chamado *TRACIMobility*, o qual inclui funcionalidades que permitem que os veículos parem em pontos pré-definidos, os quais são identificados como ponto de acidente. É parametrizado o tempo em que o acidente inicia e o tempo de duração, sendo que neste intervalo há uma troca de mensagens na rede veicular com objetivo que os veículos que fazem parte da simulação recebam os alertas da ocorrência.

Este experimento visa descrever alguns recursos de utilização frequente e disponíveis no *framework* VEINS. A escolha dos critérios a serem explorados objetivam descrever as diversas possibilidades de uso da ferramenta VEINS como:

- Distância percorrida pelos veículos;
- Velocidade de deslocamento;
- Emissão de CO₂;
- Contagem de pacotes enviados, recebidos, taxa de descarte e tempo de latência de rede utilizadas pelos protocolos de roteamento de rede;
- Mensagens de *broadcast*.

Os protocolos AODV (*Ad hoc On-Demand Distance Vector*) e DYMO (*Dynamic MANET On-demand*), utilizados na simulação pertencem à categoria de protocolos de redes adhoc e agem de forma reativa, ou seja, não mantêm rotas desnecessárias em suas tabelas. Quando precisam entregar um pacote de dados, são ativados e fazem a descoberta de vizinhos na rede para posteriormente obter o caminho para transportar este pacote de dados.

O *Framework* VEINS permite gerar gráficos pré-definidos para redes veiculares, bem como gerar estatísticas de aplicações simuladas. A Figura 5 apresenta como a ferramenta mostra os resultados obtidos com a integração do Omnet++, SUMO e VEINS.

Neste exemplo, é possível verificar a variação da velocidade e também a distância de cada veículo na simulação. Este experimento apresentou em média 2.523m de distância percorrida de cada veículo.

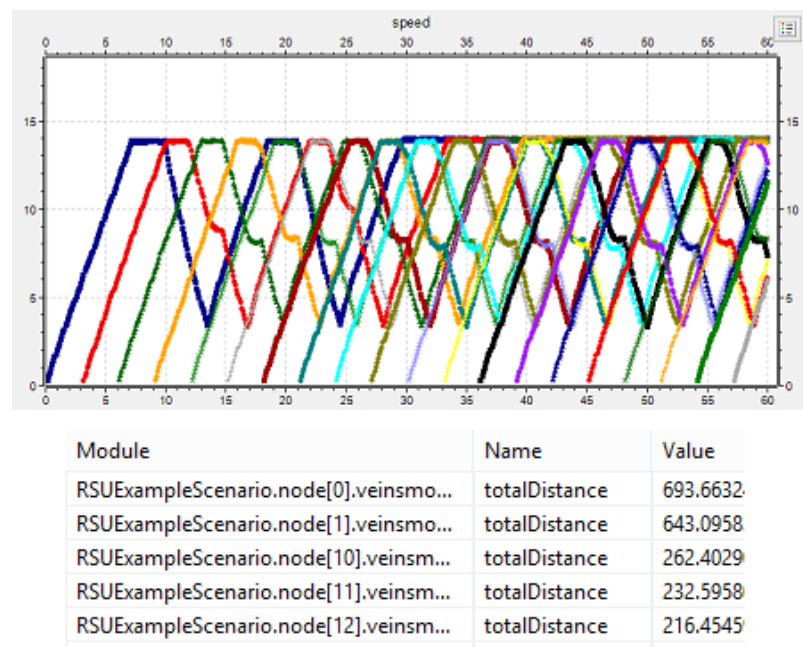


Figura 5. Gráficos gerados pelo *framework* VEINS

Além da geração de gráficos, o *framework* permite a criação de dados em formato vetorial. Pode-se observar o gráfico gerado das velocidades dos veículos e suas alterações de aceleração e desaceleração, bem como visualizar os dados referentes às distâncias percorridas por cada veículo. A Figura 6 apresenta os resultados da simulação com dois protocolos a nível de roteamento de pacotes em redes VANETS.

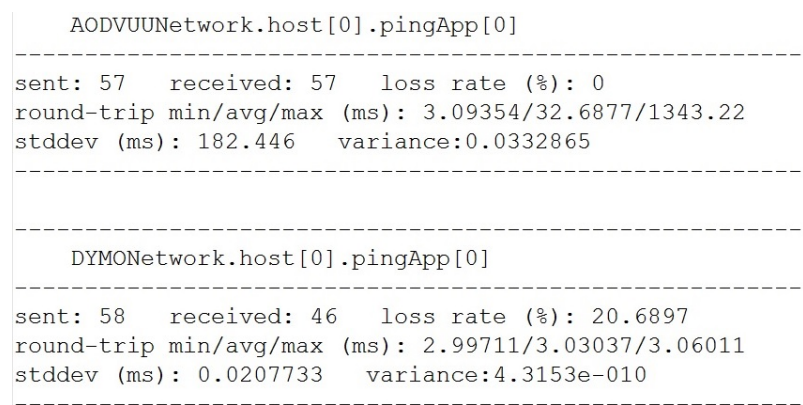


Figura 6. Resultados gerados pelo *framework* VEINS

Por exemplo, pode-se observar os índices dos pacotes perdidos. Para o protocolo DYMO a perda é de 20% e para o protocolo AODV de 0%. Para este experimento foi utilizado o aplicativo pingApp. Também é possível analisar a quantidade de pacotes enviados e recebidos nesta rede, além da média em milissegundos do parâmetro RTT (*Round Trip Time*), que corresponde ao tempo vida de um pacote em determinado segmento de

rede. Pode-se observar que o protocolo AODV teve uma latência de 32.68ms e o protocolo DYMO obteve um tempo de 3.03ms. Este parâmetro indica que o protocolo DYMO tem melhor desempenho que o protocolo AODV no quesito tempo de entrega das mensagens.

Além das estatísticas de cada protocolo de roteamento, as redes VANETs também necessitam controlar a quantidade de mensagens de controle enviadas pela rede, sendo que uma grande inundação de mensagens pode sobrecarregar os mecanismos de roteamento e a comunicação de emergência por exemplo pode ficar comprometida. A Figura 7 mostra a quantidade de mensagens de *broadcast* enviadas, sendo o total de 1.470 mensagens de *broadcast* e com uma conexão média de 256 segundos por nodo.

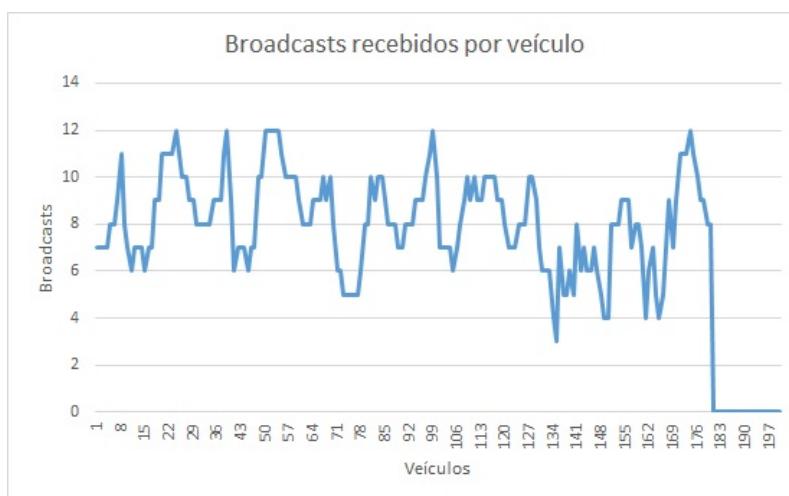


Figura 7. Broadcasts gerados pela rede

A Figura 8 apresenta os resultados da simulação com o *framework* VEINS com a emissão de CO₂ feita por cada veículo utilizado nesta simulação. Observa-se que é possível apresentar a contagem de poluentes de cada nodo da rede. É possível também compor o montante de emissões de CO₂ para cada simulação.

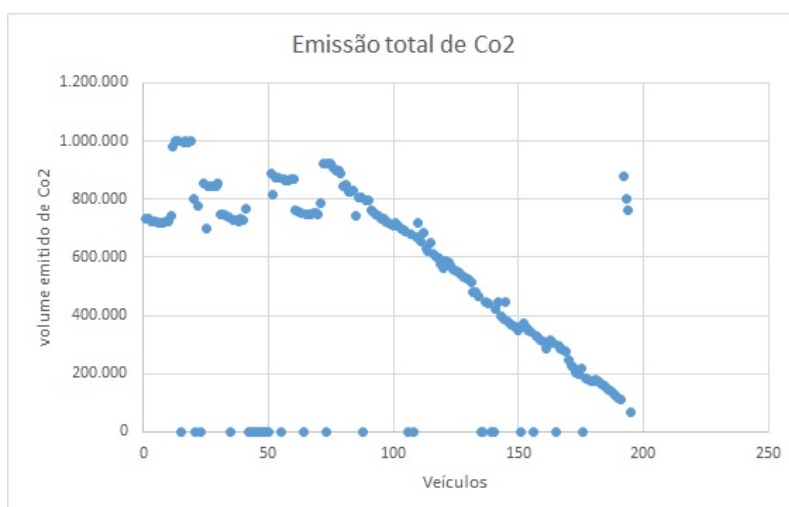


Figura 8. Resultados da emissão de CO₂ pelo *framework* VEINS

5. Conclusões e trabalhos futuros

Este trabalho descreveu conceitos gerais de redes veiculares VANETs, e também apresentou as principais ferramentas utilizadas para simulações de redes veiculares. Considera-se essencial o uso de simuladores em cenários atuais com necessidade de planejamento viário, dimensionamento de fluxo de veículos, programação semafórica para escoamento coerente do trânsito.

Conforme demonstrado neste trabalho, percebe-se que é possível utilizar ferramentas integradas como Omnet++, SUMO e VEINS para simular VANETs. Também as ferramentas permitem prever dentre outros resultados, a emissão de poluentes em determinados trechos simulados e prever como será o comportamento do trânsito e da comunicação veicular.

Como trabalhos futuros busca-se analisar as situações específicas de protocolos de redes veiculares a fim de pesquisar técnicas de transmissão de mensagens em redes veiculares, bem como contribuir com a comunidade científica com análises pertinentes aos simuladores de tráfego para planejar os controles semafóricos e fluxo veicular em centros urbanos.

Referências

- Atul B.Kathole, Y. P. (2013). Survey of topology based reactive routing protocols in vanet. *International Journal of Scientific Engineering Research*, 4(6).
- DENATRAN, D. N. d. T. (2008). Estatística da frota 2008. Disponível em: <http://www.denatran.gov.br/download/frota/Frota2008.zip> Acesso em fev. 2015.
- IBGE, I. B. d. G. e. E. (2010). Censo 2010. Disponível em: <http://www.ibge.gov.br> Acesso em mai. 2015.
- J., B. M. (2012). Tutorial for simulating vanet and its (using omnet++ and sumo).
- Martinez, F. J., Toh, C. K., Cano, J.-C., Calafate, C. T., and Manzoni, P. (2011). A survey and comparative study of simulators for vehicular ad hoc networks (vanets). *Wirel. Commun. Mob. Comput.*, 11(7):813–828.
- Noori, H. (2012). Realistic urban traffic simulation as vehicular ad-hoc network (vanet) via veins framework. In *12th Conference of Open Innovations Framework Program. FRUCT*.
- Rashvand, H. F., editor (2013). *Dynamic Ad-Hoc Networks*. Telecommunications. Institution of Engineering and Technology. Disponível em: <http://digital-library.theiet.org/content/books/te/pbte059e> Acesso em fev. 2015.
- Sommer, C., German, R., and Dressler, F. (2011). Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis. *IEEE Transactions on Mobile Computing*, 10(1):3–15.
- SUMO, S. U. M. (2001). Dlr. Disponível em: <http://www.dlr.de/ts/en/desktopdefault.aspx/> Acesso em set. 2014.

Processo de Engenharia do Conhecimento para o Desenvolvimento de uma Ontologia do Domínio da Doença Renal Crônica para Capacitação de Profissionais de Atenção Primária à Saúde

Sidney Patrício Reinaldo Campos¹, Natasha Correia Queiroz Lino¹

¹Programa de Pós Graduação em Informática (PPGI), Universidade Federal da Paraíba (UFPB), João Pessoa – PB – Brasil

sidney@di.ufpb.br, natasha@ci.ufpb.br

Abstract. *This article describes the process of knowledge engineering for representation of a subdomain of chronic kidney disease (definition, diagnosis, stages, clinical factors that affect disease and protective measures for each stage and clinical factors addressed) in an ontology that allows the sharing and reuse of information on the part of educational softwares developed to train primary health care professionals in the care of patients with chronic kidney disease.*

Resumo. *Este artigo descreve o processo de engenharia de conhecimento para representação de um subdomínio da Doença Renal Crônica (definição, formas de diagnóstico, estágios, fatores clínicos que afetam a doença e medidas nefroprotetoras para cada estágio e fatores clínicos abordados) em uma ontologia que permita o compartilhamento e o reuso de informações por parte de softwares educacionais desenvolvidos para capacitar profissionais de Atenção Primária à Saúde nos cuidados de pacientes com Doença Renal Crônica.*

1. Introdução

Diante da natureza assintomática da Doença Renal Crônica (DRC) até o seu estágio mais avançado, os profissionais de Atenção Primária à Saúde (APS) precisam estar cada vez mais atentos aos fatores clínicos que afetam a progressão da doença. Uma vez ciente dos fatores clínicos, precisam conhecer as medidas nefroprotetoras que necessitam ser tomadas para cada um desses fatores, com a finalidade de amenizar o risco da DRC. Além de ter a capacidade de caracterizar a DRC e, consequentemente, realizar o estadiamento. Do estágio 3 ao 5, o paciente portador de DRC deve ser acompanhado por um especialista. Nos estágios 1 e 2, o profissional de APS pode atuar e impedir que a doença progrida.

Por esses motivos, é necessário facilitar o acesso às informações relacionadas à DRC por parte dos profissionais de APS. Tais informações devem estar organizadas para que sejam encontradas de forma rápida, clara e sucinta. Segundo autores da área [Bastos e Kirsztajn 2011, Paige e Nagami 2009], a carência dos profissionais de APS se encontra principalmente nos seguintes tópicos: definição, formas de diagnóstico, estágios, fatores clínicos que afetam a DRC e medidas nefroprotetoras para cada estágio e fatores clínicos abordados.

Este trabalho aborda o processo de engenharia de conhecimento para a representação das informações relativas à DRC, mencionadas anteriormente, em uma

ontologia, que é a técnica de representação do conhecimento escolhida, justamente para facilitar o compartilhamento e reuso de informações por parte, principalmente, de *softwares* educacionais. Ontologias descrevem conceitos, relações, restrições e axiomas de um domínio usando uma organização taxonômica, baseada em generalização e especialização [Breitman 2005]. Além da capacidade de classificação, a expressividade semântica das ontologias reside na possibilidade de inferência sobre seu próprio conteúdo [Albrecht 2007].

2. Fundamentação Teórica

Esta seção fornece uma visão geral sobre a área de representação do conhecimento, focando em uma de suas técnicas, a ontologia. Além disso, aborda a importância da Atenção Primária nos cuidados ao paciente com Doença Renal Crônica.

2.1. Representação do Conhecimento

Um dos principais paradigmas da “representação do conhecimento” no contexto da IA é: “Como transmitir o conhecimento do mundo para um robô ou outro sistema computacional, dando – lhe a capacidade adequada de raciocínio, de modo que este conhecimento possa ser utilizado para permitir ao sistema identificar, atribuir significado e explorar os fatores existentes em seu ambiente?” [Brachman 1990].

Para fornecer o conhecimento para um software, é necessário escolher uma maneira de representá-lo. Uma representação de uma situação é a tradução desta situação em um sistema, que compreende não só um vocabulário que dá nomes às coisas e às relações, mas também as operações que podem ser realizadas sobre elas, e os fatos e restrições acerca dessas coisas. Assim, os principais elementos de uma representação são a informação que está explicitada e a forma em que está fisicamente codificada [Rover 2001].

A seguir, ontologia, que é uma das principais técnicas de representação do conhecimento, será apresentada pela perspectiva da Ciência da Computação.

2.2. Ontologia e Ciência da Computação

Em ciência da computação, ontologias são desenvolvidas para facilitar o compartilhamento e reuso de informações. Elas descrevem conceitos, relações, restrições e axiomas de um domínio usando uma organização taxonômica, baseada em generalização e especialização [Breitman 2005].

Considerando que D representa um domínio específico e que W corresponde ao conjunto de conceitos existentes neste domínio, pode-se utilizar um conjunto de relações, representadas por R , para interligar os conceitos e criar uma estrutura pretendida deste domínio, ou seja, uma conceituação C que representa o domínio. Neste contexto, a ontologia define uma “linguagem” que permite que a conceituação seja especificada do ponto de vista computacional, impondo regras de combinação entre os conceitos e seus relacionamentos dentro de um domínio de conhecimento, em um formato taxonômico, de forma clara, buscando interpretação única. [Moraes 2011].

Resumidamente, o consórcio W3C coloca que ontologias devem prover descrições para os seguintes elementos [Bailer 2013]:

1. Classes (ou “coisas”) nos vários domínios de interesse;

2. Relacionamentos entre essas “coisas”;
3. Propriedades (ou atributos) que essas “coisas” devem possuir.

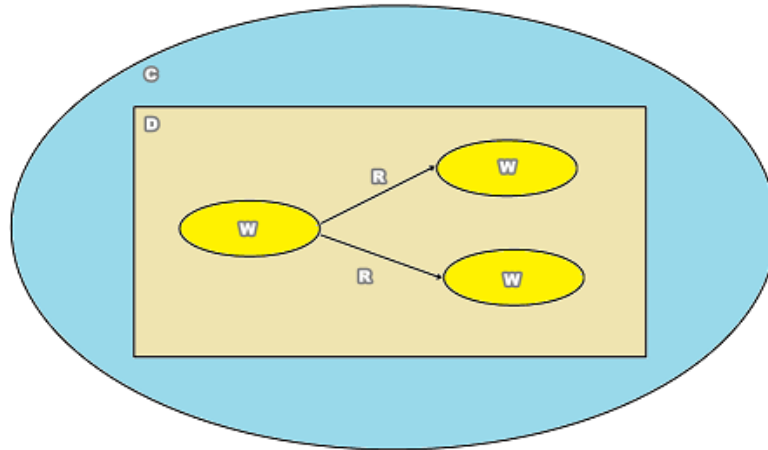


Figura 1. Estrutura da Ontologia

A expressividade semântica das ontologias reside na possibilidade de inferência sobre seu próprio conteúdo [Albrecht 2007]. Como Ontologia representa uma linguagem formal para definição explícita de uma base de conhecimento, métodos de raciocínio permitem inferir informações implícitas em tal base de dados, diminuindo o esforço de representação redundante, já que se pode inferir novas informações baseadas nas informações já existentes [Albrecht 2007]. Neste contexto, os processos de inferência são executados por motores de raciocínio (*reasoners*).

Raciocinadores computacionais realizam operações lógicas aplicadas sobre bases de conhecimento, agindo de modo a executar regras que buscam alcançar respostas a um processo de inferência [Brignoli 2011].

2.3. Importância da Atenção Primária nos cuidados ao paciente com Doença Renal Crônica

De acordo com Censo de Diálise realizado pela Sociedade Brasileira de Nefrologia (2013), o número estimado de pacientes em tratamento dialítico no Brasil é de 100.397, o que significa que existem mais de cem mil pessoas que dependem de procedimentos para remoção das substâncias tóxicas que ficam retidas quando os rins deixam de funcionar adequadamente. Sob a perspectiva da DRC, são submetidos à diálise ou a outro tipo de Terapia de Substituição Renal (TRS), pacientes que se enquadram no último estágio doença (Estágio 5) [Bastos et al. 2010]. A evolução para o último estágio pode ser provocada, entre outros fatores, pela falta de monitoramento dos indicadores de DRC e consequente encaminhamento tardio de pacientes para o atendimento nefrológico [Bastos e Kirsztajn 2011], que é geralmente resultado da escassez de informações por parte dos profissionais de APS (Atenção Primária à Saúde) sobre a epidemiologia da doença, os critérios para diagnóstico ou dos objetivos e resultados dos cuidados nefrológicos nos estágios iniciais da doença [Bastos et al. 2007].

A APS surge como um modelo de organização que atua na área da saúde com objetivo de detectar e manejar enfermidades, prevenir doenças e promover a saúde,

contribuindo diretamente para redução dos custos crescentes dos sistemas de saúde [Bastos et al. 2007]. O profissional de APS é o de primeiro contato do usuário e, geralmente, possui uma formação geral e abrangente para atender os problemas de saúde mais frequentes da população [Pena et al. 2012].

Segundo Bastos e Kirsztajn (2011), conscientizar profissionais de APS, que frequentemente lidam com pacientes em risco para DRC, sobre a importância de encaminhar pacientes para acompanhamento conjunto com um nefrologista, gera benefícios potenciais como, por exemplo, identificação e tratamento das causas reversíveis da insuficiência renal; diagnóstico e correção de fatores que pioram a função renal (por exemplo, o uso de agentes nefrotóxicos); estabilização da Taxa de Filtração Glomerular; identificação e correção de grandes complicações e morbidades prevalentes na DRC; obtenção de melhores parâmetros bioquímicos, psicológicos e físicos no início da TRS; e redução de gastos com o programa de diálise e transplante renal, que no Brasil situa-se ao redor de 1,4 bilhões de reais ao ano [Pena et al. 2012].

Além da cooperação entre nefrologistas e não nefrologistas, a qualidade das práticas de APS exige que se invista em uma nova política de formação e em um processo permanente de capacitação dos seus profissionais [Bastos et al. 2007].

Diversas iniciativas devem ser tomadas com o objetivo de inserir profissionais de APS no tratamento da DRC. Ações educativas, por exemplo, podem ser um valioso instrumento para compartilhar o conhecimento de nefrologistas com profissionais de APS, desde que assumam um formato acessível (palestras, folhetos educativos, sistemas de informação em saúde) e sejam baseadas em um conjunto de informações relevantes sobre a DRC que forneçam a capacidade de detecção de casos, prevenção das complicações e preparo adequado dos pacientes para o enfrentamento das fases mais avançadas da doença [Bastos et al. 2007].

3. Trabalhos Relacionados

O trabalho de Bittencourt (2006) descreve as ontologias existentes em ambientes interativos de aprendizagem. Segundo o autor, um ambiente interativo de aprendizagem deve possuir uma ontologia do domínio, do estudante, do modelo colaborativo e do modelo pedagógico. A proposta deste trabalho é contribuir com ambientes de aprendizagem da área da Nefrologia, através do desenvolvimento de uma ontologia do domínio da Doença Renal Crônica como um formalismo de representação do conhecimento da área, a exemplo de outros trabalhos [Bittencourt et al. 2007, Bogdan 2011, Walls et al. 2012, Ferreira Filho et al. 2004] que também utilizam ontologias para modelar o conhecimento do domínio de interesse.

4. Processo de Engenharia do Conhecimento

A Engenharia do Conhecimento (EC) é responsável pelo desenvolvimento de metodologias e técnicas para construir sistemas baseados em conhecimento.

Nesta seção serão discutidas as quatro etapas da EC Clássica: Aquisição de Conhecimento (extração do conhecimento de suas diversas fontes), Representação de Conhecimento (modelagem do domínio), Validação do Conhecimento (verificação de consistência) e Recuperação do Conhecimento (apresentação do raciocínio ao usuário) [Nazario et al. 2012].

4.1. Aquisição de Conhecimento

O conhecimento sobre DRC, utilizado na modelagem da ontologia, foi retirado de vários artigos científicos da área [Bastos e Kirsztajn 2011, Bastos et al. 2007, Bastos et al. 2010, Paige e Nagami 2009]. Como foco principal, ele busca responder cinco perguntas específicas:

1. O que é a DRC?
2. Como detectar pacientes com DRC?
3. Como se caracterizam os estágios da doença?
4. Quais são os fatores clínicos que determinam a progressão da DRC?
5. Quais são as medidas nefroprotetoras indicadas para impedir a progressão da DRC sob a perspectiva de cada fator clínico e dos estágios da doença?

As perguntas foram formadas a partir de diferentes aspectos da DRC, que alguns autores da área [Bastos e Kirsztajn 2011, Bastos et al. 2007, Bastos et al. 2010, Paige e Nagami 2009] consideram como fundamentais para que um profissional da APS tenha capacidade de perceber os indicadores da doença em pacientes que eles assistem.

4.2. Representação do Conhecimento

O processo de desenvolvimento da ontologia deste trabalho é baseado no método 101 proposto por Noy e Guinness (2001). O método 101 descreve o processo de modelagem de uma ontologia em sete passos.

O passo um exige três informações específicas: “Qual o domínio, propósito e para que informações a ontologia deve fornecer respostas?”. A ontologia, proposta neste trabalho, representa um micro domínio da DRC, abordando a definição, formas de diagnóstico, estágios, fatores clínicos que afetam a DRC e medidas nefroprotetoras para cada estágio e fatores clínicos abordados. Desta forma, busca responder as cinco perguntas identificadas na fase de aquisição de conhecimento.

O passo dois chama atenção para a importância do reuso de ontologias que já foram desenvolvidas para representar o domínio especificado no passo anterior. *The Human Phenotype Ontology* (HPO) [Köhler et al. 2014] e *International Classification of Diseases, Version 10 - Clinical Modification* (ICD10CM) [Brooks 2015] foram investigadas, porém elas restringem o domínio da DRC a sua definição e estágios. E, como citado anteriormente, o objetivo da ontologia, proposta neste trabalho, é representar e fornecer informações que abordam, além da definição da DRC e seus estágios, formas de diagnóstico, fatores clínicos que afetam a doença e medidas nefroprotetoras para cada estágio e fatores clínicos abordados. Portanto, este trabalho cria sua própria ontologia para representar o recorte do domínio da DRC pretendido.

O passo três e quatro propõem a enumeração dos termos importantes da ontologia e a definição de classes e hierarquia de classes, respectivamente. Porém, as classes isoladamente não fornecem subsídios suficientes para responder às questões de competência do passo um. Uma vez que as classes foram definidas, é necessário descrever a estrutura interna dos seus conceitos [Breitman 2005], portanto, o passo cinco é reservado para a definição de propriedades das classes, e o passo seis para definir os valores das propriedades que, neste caso, serão cadeias de caracteres.

Consultando os artigos científicos utilizados para estruturar as informações que serão representadas na ontologia, pode-se observar um conjunto de axiomas do domínio da DRC, ou seja, sentenças ou preposições consideradas um consenso sobre o tema pelos profissionais da área. A partir destes axiomas, podem ser extraídos os termos relevantes do domínio que, conseqüentemente, serão convertidos em classes ou em propriedades de objeto da ontologia. Entende-se por propriedades de objeto, aquelas que são responsáveis por conectar duas classes da ontologia [Horridge et al. 2004]. Além disso, são elas que impõem as restrições, utilizadas para definir alguns limites para indivíduos (instâncias) que pertencem a uma classe [Breitman 2005].

Os principais axiomas serão mencionados a seguir:

1. A **Doença Renal Crônica** é baseada em três componentes: (1) um componente anatômico ou estrutural (**marcadores de lesão renal**); (2) um componente funcional (baseado na **Taxa de Filtração Glomerular**, TFG) e (3) **um componente temporal**.
2. A KDOQI sugeriu que a **Doença Renal Crônica** deveria ser classificada em estágios baseados na Taxa Filtração Glomerular e na presença de lesão renal. A DRC é dividida em **estágios de 1 a 5**, nos estágios 3, 4 e 5 não há necessidade de se documentar a presença de lesão da estrutura renal.
3. A **probabilidade de progressão da Doença Renal Crônica** é determinada por interações complexas que envolvem vários **fatores clínicos**.

Utilizando as questões de competência do passo 1, pode-se identificar as perguntas para as quais a ontologia deve fornecer respostas e, conseqüentemente, identificar os termos relevantes do domínio a partir de palavras-chaves que se encontram nas perguntas. Questões como “O que é a DRC?”, “Como se caracterizam os estágios da doença?” e “Quais são os fatores clínicos que determinam a progressão da DRC?” justificam o motivo pelo qual os termos em negrito nos axiomas são relevantes ao domínio e precisam ser transformados em classes da ontologia. Em consequência disso, os sublinhados, que conectam os termos relevantes, são transformados em propriedades de objeto.

O segundo axioma sugere que a classe Estágios tenha subclasses que representem cada estágio em específico. Outros dois axiomas permitem que as classes Marcadores de Lesão Renal e Fatores Clínicos também tenham subclasses. São eles:

4. Entre os marcadores de lesão renal, a **proteinúria** é o mais frequentemente utilizado. Porém, outros marcadores de lesão renal também podem ser empregados, tais como outras alterações na urina (por exemplo, **hematúria glomerular**), imagens ultrassonográficas anormais (por exemplo, **cistos na doença renal policística do adulto**) ou alterações histopatológicas vistas em biópsias renais (por exemplo, **alterações glomerulares com ou sem envolvimento túbulo-intersticial**).
5. A probabilidade de progressão da DRC é determinada por interações complexas que envolvem vários fatores clínicos como, por exemplo, como, por exemplo, **idade, sexo, proteinúria, hipertensão, Diabetes Mellitus, obesidade, tabagismo, anemia, doença cardiovascular, acidose metabólica, ingestão de proteínas, creatinina, uso de antiácidos e uso de polietileno glicol, além de fatores genéticos**.

Os termos em **negrito** no axioma 4 são marcadores de lesão renal e no axioma 5 são fatores clínicos. Portanto, constituem uma especialidade das classes Marcadores de Lesão Renal e Fatores Genéticos, respectivamente, o que os tornam subclasse dentro do contexto da ontologia descrita neste trabalho.

Diante das informações estruturadas até agora, mesmo com todas as classes e propriedades de objeto definidas, a ontologia é capaz de informar apenas quais são os componentes da DRC, quantos são os estágios, quais são os fatores clínicos que afetam a doença e os tipos de marcadores de lesão renal. Tais informações não respondem todas as questões de competências do passo 1, perguntas como, por exemplo, “Como se caracterizam os estágios da doença?” e “Quais são as medidas nefroprotetoras indicadas para impedir a progressão da DRC sob a perspectiva de cada fator clínico e dos cinco estágios da doença?” estão sem resposta.

Nesse aspecto são usadas as propriedades de anotação, ou seja, propriedades usadas para adicionar metadados as classes [Horridge et al. 2004]. Cada classe tem um conjunto específico de propriedades de anotação, que são determinadas de acordo com as informações que a ontologia tem que prover sobre o tópico que a classe representa.

Considerando que a primeira questão que a ontologia deve responder é “O que é a DRC?”, a classe Doença Renal Crônica deve ser associada a propriedade de anotação Definição. Tal propriedade, como o próprio nome diz, fornece uma definição para a doença. A segunda questão é “Como detectar pacientes com DRC?”. Como já foi mencionado anteriormente, a DRC é associada a três componentes: um anatômico ou estrutural (marcadores de lesão renal), um funcional (Taxa de Filtração Glomerular) e um temporal. Sabe-se que para cada um desses componentes existe uma classe, e que a classe Marcadores de Lesão Renal está associada a um conjunto de subclasses. Portanto, para a superclasse Marcadores de Lesão Renal é necessária apenas a propriedade de anotação Definição. Mas quando se trata de cada marcador de lesão renal em específico, que são representados pelas Subclasses, é necessário fornecer, além da definição, as formas de diagnóstico do marcador. Também deve-se definir e apresentar as formas de detecção da Taxa de Filtração Glomerular, sendo assim, a classe Taxa de Filtração Glomerular tem como propriedades de anotação Definição e Formas de Detecção. Já para a classe Componente Temporal, é necessária apenas a propriedade de anotação Definição com o objetivo de descrever tal componente.

As questões relacionadas ao estadiamento da doença são “Como se caracterizam os estágios da doença?” e “Quais são as medidas nefroprotetoras indicadas para impedir a progressão da DRC sob a perspectiva de cada estágio da doença?”. Para respondê-las, a ontologia deve conhecer a definição de cada estágio e as medidas nefroprotetoras que impedem que a doença progrida de um estágio para o outro. Por isso, cada classe que representa um estágio conta com as propriedades de anotação Definição e Medidas Nefroprotetoras Correspondentes.

As últimas questões que precisam ser respondidas são “Quais são os fatores clínicos que determinam a progressão da DRC?” e “Quais são as medidas nefroprotetoras indicadas para impedir a progressão da DRC sob a perspectiva de cada fator clínico?”. É fato que a primeira, das duas perguntas restantes, já foi respondida previamente desde que os fatores clínicos, presentes no axioma 5, foram transformados em subclasses da classe Fatores Clínicos. Mais ainda é necessário definir cada um desses fatores e fornecer, com o objetivo de impedir a progressão da DRC, as medidas nefroprotetoras utilizadas na

presença deles em pacientes. Por este motivo, a superclasse Fatores Clínicos tem apenas Definição como propriedade de anotação, em contrapartida, cada subclasse tem Definição e Medidas Nefroprotetoras Correspondentes como suas propriedades.

É importante destacar que, diferentemente das subclasses da classe Marcadores de Lesão Renal que possuem uma propriedade que fornece informações sobre formas de diagnosticar o marcador em questão, as subclasses da classe Fatores Clínicos não possuem propriedades que forneçam informações do diagnóstico de tais fatores, pois este trabalho pressupõe que o profissional de APS é capacitado para detectar a presença desses fatores no paciente, mas pode desconhecer a relação que existe entre eles e a DRC. A única subclasse de Fatores Clínicos que possui a propriedade Formas de Diagnóstico é a Proteinúria, o que se deve ao fato dela também ser subclasse de Marcadores de Lesão Renal, sua presença é um dos pré-requisitos para se detectar a DRC.

Como essa ontologia tem propósito educacional, este trabalho desconsidera o passo sete, que defende a criação de instâncias para as classes que a compõem, pois o propósito da utilização de ontologia, neste trabalho, é sua capacidade de inferência que permite a um profissional de APS realizar consultas que englobam o domínio do trabalho.

4.3. Validação do Conhecimento

O conhecimento do domínio e sua representação (ontologia) foram avaliados por uma nefrologista da Universidade Federal da Paraíba (especialista da área), que também é membro do grupo de pesquisa ao qual este trabalho está associado.

4.4. Recuperação do Conhecimento

Depois de estruturar conceitualmente a ontologia, a conceituação é formalizada na linguagem OWL [W3C Recommendation 2004], através do editor de ontologias Protégé [Gonçalves 2015] (Este trabalho encontra-se nesta fase). Uma vez que o conhecimento do domínio de interesse deste trabalho já está estruturado na ontologia, a mesma torna-se apta a responder as questões de competência levantadas na fase de aquisição do conhecimento.

O editor Protégé fornece raciocinadores computacionais que permitirão a um profissional de APS realizar consultas como, por exemplo, quais medidas nefroprotetoras devem ser aplicadas a um paciente que possui hipertensão arterial, é obeso e que está no Estágio 1 da DRC, entre outras consultas que englobam o domínio do trabalho.

5. Conclusão

Com a futura disponibilidade da ontologia, que representa os aspectos principais da DRC, na página do grupo (<http://www.di.ufpb.br/ai-medical-informatics/news/>), espera-se que *softwares* educacionais a utilizem para tornar informações relativas à doença mais acessíveis de forma simplificada, personalizada e controlada, tanto para profissionais de APS, como para outras partes interessadas (por exemplos, pacientes). Dessa forma, além de ser destinada para autoaprendizagem de profissionais de APS, ela também poderia ser usada para instruir pacientes.

Sob a perspectiva da Nefrologia, a expectativa é que o treinamento dos profissionais de APS, que lidam com pacientes em risco para DRC, resulte na compreensão do diagnóstico e tratamento a ser seguido, dos serviços de prevenção, e no

desenvolvimento da capacidade de detectar os indicadores da DRC para tomar decisões com a finalidade de melhorar o estado de saúde do paciente.

6. Referências

- BASTOS, Marcus Gomes; KIRSZTAJN, Gianna Mastroianni. Doença renal crônica: importância do diagnóstico precoce, encaminhamento imediato e abordagem interdisciplinar estruturada para melhora do desfecho em pacientes ainda não submetidos à diálise. **Jornal Brasileiro de Nefrologia**. São Paulo, p. 93-108. mar. 2011.
- PAIGE, Neil M.; NAGAMI, Glenn T.. The Top 10 Things Nephrologists Wish Every Primary Care Physician Knew. **Mayo Clinic Proceedings**. New York, p. 180-186. fev. 2009.
- ALBRECHT, Felipe; CUNHA, Adriano; FERNANDES, Ricardo Queiroz Araujo. **Inferência sobre Ontologias: o reencontro com PROLOG**. 2007. 27f. Monografia - Instituto Militar de Engenharia, Rio de Janeiro, 2007.
- BRACHMAN, Ronald Jay. The Future of Knowledge Representation. In: NATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE, 8., 1990, Boston. **Proceedings of Eighth National Conference on Artificial Intelligence**. Boston: [s. N.], 1990. v. 2, p. 1082 - 1092.
- ROVER, Aires José. **Informática no direito: Inteligência Artificial, introdução aos sistemas especialistas legais**. Curitiba: Jurua Editora, 2001.
- BREITMAN, Karin. **Web Semântica: a Internet do Futuro**. Rio de Janeiro: Editora Ltc – Livros Técnicos e Científicos Ltda, 2005.
- MORAIS, Edison Andrade Martins; AMBRÓSIO, Ana Paula L.. *Ontologias: conceitos, usos, tipos, metodologias, ferramentas e linguagens*. Goiás: Relatório Técnico, 2007.
- BAILER, Werner et al. **Ontology for Media Resources 1.0 W3C Recommendation**. Disponível em: <<http://www.w3.org/TR/mediaont-10/>>. Acesso em: 13 ago. 2013.
- BRIGNOLI, Juliano T.; SELL, Denilson; GAUTHIER, Fernando Ostuni. Abordagens Estocásticas para Raciocinadores Aplicáveis em Web Semântica. In: ONTOBRAS
- MOST, [s. N.], 2011, Gramado. **Proceedings of ONTOBRAS-MOST 2011**. Gramado: [s. N.], 2011. p. 105 - 110.
- PENA, Paulo Félix de Almeida et al. Cuidado ao paciente com Doença Renal Crônica no nível primário: pensando a integralidade e o matriciamento. **Revista Ciência & Saúde Coletiva**, Rio de Janeiro, v. 17, n. 11, p. 3135 – 3144. nov. 2012.
- BASTOS, Rita Maria Rodrigues; BASTOS, Marcus Gomes; TEIXEIRA, Maria Teresa Bustamante. A Doença Renal Crônica e os desafios da Atenção Primária à Saúde na sua detecção precoce. **Revista Aps**, [s. N.], v. 10, n. 1, p.46-55, jun. 2007.
- SOCIEDADE BRASILEIRA DE NEFROLOGIA. **Censo de Diálise 2013**. Disponível em: <http://www.sbn.org.br/pdf/censo_2013-14-05.pdf>. Acesso em: 05 jan. 2015.
- BASTOS, Marcus Gomes; BREGMAN, Rachel; KIRSZTAJN, Gianna Mastroianni. Doença renal crônica: frequente e grave, mas também prevenível e tratável. **Revista da Associação Médica Brasileira**, São Paulo, v. 56, n. 2, p.248-253, jan. 2010.

- BITTENCOURT, Ig I. et al. Um Sistema Tutor Baseado em Agentes no Domínio da Medicina. In: CONGRESSO DA SOCIEDADE BRASILEIRA DE COMPUTAÇÃO (CSBC), 27., 2007, Alagoas. **Anais do XXVII Congresso da Sociedade Brasileira de Computação (CSBC)**. Rio de Janeiro: Sociedade Brasileira de Computação, 2007. p. 372 - 379.
- BITTENCOURT, Ig I.. Sistema de Autoria para Construção de Ambientes Interativos de Aprendizagem baseada em agentes. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 17., 2006, Alagoas. **Anais do XVII do SBIE**. Brasília: Sociedade Brasileira de Computação, 2006. p. 487 - 496.
- BOGDAN, Crenguta Madalina. Domain Ontology of the VirDenT System. *International Journal Of Computers Communications & Control*. Romania, p. 45-52. mar. 2011.
- WALLS et al. **Ontologies as integrative tools for plant science**. 2012. Disponível em: <<http://www.ncbi.nlm.nih.gov/pubmed/22847540>>. Acesso em: 18 jan. 2014.
- FERREIRA FILHO, Raymundo Carlos Machado et al. Implementação de Sistema Tutor Inteligente para Geotecnia. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 15., 2004, Rio Grande do Sul. **Anais do XV SBIE**. Manaus: Sociedade Brasileira de Computação, 2004. p. 16 - 18.
- NAZARIO, Debora Cabral et al. O Processo de Aquisição na Engenharia do Conhecimento: técnicas de Extração e Elicitação. In: CONTECSI - INTERNATIONAL CONFERENCE ON INFORMATION SYSTEMS AND TECHNOLOGY MANAGEMENT, 9., 2012, São Paulo. **Proceedings of CONTECSI**. São Paulo: Usp, 2012. p. 499 - 517.
- NOY, Natalya; GUINNESS, Deborah. **Ontology Development 101: A Guide to Creating Your First Ontology**. 2001. Disponível em: <http://protege.stanford.edu/publications/ontology_development/ontology101.pdf>. Acesso em: 16 jun. 2014.
- KÖHLER, Sebastian et al. The Human Phenotype Ontology project: linking molecular biology and disease through phenotype data. **Nucleic Acids Research**. United Kingdom, p. 966-974. jan. 2014.
- BROOKS, Patricia. **International Classification of Diseases, Version 10 - Clinical Modification**. 2015. Disponível em: <<http://bioportal.bioontology.org/ontologies/ICD10CM>>. Acesso em: 10 jan. 2015.
- HORRIDGE, Matthew et al. **A Practical Guide To Building OWL Ontologies Using The Protégé-OWL Plugin and CO-ODE Tools Edition 1.0**. 2004. Disponível em: <<ftp://gi29.geoinfo.tuwien.ac.at/courses/Ontology/ProtegeOWLTutorial.pdf>>. Acesso em: 02 fev. 2015.
- W3C RECOMMENDATION. **OWL Web Ontology Language Overview**. 2004. Disponível em: <<http://www.w3.org/TR/owl-features/>>. Acesso em: 08 jan. 2014.
- GONÇALVES, Rafael et al. **Protégé - A free, open-source ontology editor and framework for building intelligent systems**. Disponível em: <<http://protege.stanford.edu/>>. Acesso em: 04 fev. 2015.

GROSS: Um framework para auxiliar na experimentação de escalonadores de Sistemas Operacionais

Ádler J. Gross¹, Elder F. F. Bernardi¹

¹Instituto Federal de Educação, Ciência e Tecnologia Sul-rio-grandense (IFSUL) –
Passo Fundo, RS – Brasil

{adlerjgross, elderbernardi}@gmail.com

Abstract. *This paper presents an open source framework that allows users program, test and compare algorithms that control processor management of a simulated operating system. Its main use is facilitate the testing of new scheduling algorithms, but also assist in the education of operating systems managers, providing a framework for practical programming activities. This paper also will presents the details of development, operation and implementation of the framework.*

Resumo. *Este trabalho apresenta um framework de código fonte aberto que permite o usuário programar, testar e comparar seus próprios algoritmos que controlam a gerência de processador de um sistema operacional simulado. Sua principal utilidade é facilitar a experimentação de novos algoritmos de escalonamento, como também auxiliar no ensino das gerências de sistemas operacionais, visando atividades práticas de programação. Serão apresentados detalhes de desenvolvimento, funcionamento e de implementação do framework.*

1. Introdução

A experimentação de algoritmos para gerências do sistema operacional não é uma atividade trivial para aqueles que não dominam o funcionamento da programação a nível de kernel. Tal dificuldade também acontece quando há necessidade do ensino dessas gerências por meio de abordagens práticas, tanto para testar e exemplificar as técnicas e algoritmos já existentes, quanto para a programação e teste de novas abordagens. Tais dificuldades demandam a construção de um *framework* que permita a simulação de desempenho de técnicas de gerência, bem como a programação de novas técnicas, sem exigir um alto conhecimento de programação para alterar o kernel de sistemas já existentes. O trabalho aqui apresentado tem por objetivo apresentar uma proposta que enderece os problemas acima expostos, de modo que possa ser utilizado principalmente em ambiente de ensino, mas que também possa ter ferramentas para uso em ambientes de prototipação de algoritmos e técnicas de gerência de Sistemas Operacionais. Este artigo mostra a arquitetura do sistema, ilustrando com um estudo de caso de simulação da gerência do processador em sistemas operacionais.

Os requisito inicial buscado para o *GROSS* é criar um pequeno modelo com os aspectos básicos de um sistema operacional e que seja possível programar diferentes tipos de escalonadores, que têm o controle sobre como os processos devem ser executados no processador, de modo a poder compará-los.

Com o desenvolvimento se conseguiu um protótipo funcional que atende esse requisito. Portanto, o *software* simula a gerência de processador de um sistema operacional e permite a programação de seus escalonadores para que possam ser alterados de acordo como o usuário do *framework* preferir, para que se tenha várias possibilidades de testes de algoritmos.

2. Simulação para ensino de Sistemas Operacionais

O principal objetivo da simulação é servir como uma ferramenta para ilustrar aulas das disciplinas de SO, pois estudar os sistemas operacionais para uso final poderia dificultar o aprendizado do aluno devido à sua complexidade. O ensino de sistemas operacionais tem o desafio de conseguir colocar em prática as técnicas estudadas de modo que se possa visualizar resultados concretos e também que se permita o exercício de proposição e programação destas técnicas.

Para se propor o GROSS, estudou-se alguns trabalhos relacionados que têm objetivos semelhantes. A seguir são apresentados alguns simuladores para então na próxima seção apresentar de maneira detalhada o *framework* GROSS.

2.1 SOsim

SOsim¹ é um simulador de SO criado por Maia com o propósito de ser educacional. O *software* permite escolher algoritmos de escalonamento e operações do SO através da sua interface gráfica.

Um dos maiores problemas encontrados nesse simulador é a falta do código fonte. O simulador emula os principais subsistemas de um SO multiprogramável, como gerência de processos, escalonamento e memória virtual por paginação [MAIA, 2001].

2.2 BACI

Ben-Ari Concurrent Interpreter (BACI)² é um sistema que foi criado para o ensino de Programação Paralela e Distribuída. Ele é desenvolvido desde 1995 e possui executáveis para Linux, Windows, entre outros, além de existir uma versão em Java. Possui um compilador de linguagem de programação similar ao C++, o C--. Ele “simula a execução de processos concorrentes, suportando monitores e semáforos.” [STALLINGS, 2012].

Para interação com o sistema, é necessário trocar os arquivos de configuração e programar os programas que serão executados, verificando o que acontece. O interpretador possui diversos recursos, como janelas de exibição de entradas, exibição de saída, debug, monitoração das variáveis nos processos e tabela de processos, também permitindo criar pontos de parada e inspeção de programas em execução [BACI, s.d.].

2.3 SimulaRSO

SimulaRSO é uma ferramenta online criada para facilitar o ensino a distância da disciplina de SO, portanto, permite a demonstração on-line do cálculo dos tempos de

¹ Endereço eletrônico: <http://www.training.com.br/sosim/>

² Endereço eletrônico: http://inside.mines.edu/~tcamp/baci/baci_index.html

escalonamento de processos, disco e paginação de memória, cujos resultados são apresentados para o usuário através de gráficos em uma interface web.³

3. GROSS

GROSS é um *framework* para construir e experimentar algoritmos de uma maneira facilitada para gerências de sistemas operacionais existentes em sistemas operacionais. Neste primeiro protótipo do GROSS, apresenta-se como estudo de caso a gerência do processador. Na arquitetura proposta, os algoritmos criados funcionam e interagem em uma simulação simplificada, sendo responsáveis por decidir qual processo deve utilizar o processador, apresentando ferramentas para simular elementos como o processador em si, processos e estruturas para controle e organização destes.

Na utilização do GROSS os algoritmos criados são compilados junto com o *framework*, e são escolhidos em tempo de execução por um arquivo de configuração que é fornecido ao executar a aplicação. Estes algoritmos são chamados de escalonadores, cuja função principal é escolher qual processo na simulação deve executar. Estes são escalonadores de alto nível, pois devem se preocupar também em como armazenar a lista de processos, verificar se os processos são válidos para execução e fazer requisições de troca de contexto.

É possível verificar o algoritmo a ser estudado contra situações determinadas pelo próprio usuário, já que além de definir quais programas serão executados e qual algoritmo a simulação deverá usar, também é possível modificar os programas para que façam entrada e saída (I/O) e processamento conforme o necessário. O software possui a licença MIT, permitindo que seja modificado, distribuído e utilizado comercialmente sem responsabilizar o autor pelo que possa ocorrer [Open Source Initiative, 2015],

3.1 Arquitetura

A arquitetura do sistema pode ser acompanhada na Figura 1, onde está ilustrado o seu funcionamento. Assim como o BACI, preferiu-se que a aplicação receba arquivos de entradas, um que configure a simulação e um outro que indique as ações de cada programa, de modo que cada programa possa ser único.

As classes Program e Instruction estão relacionadas com o carregamento dos programas. O arquivo com as ações de um programa é carregado e dividido em várias Instructions, que por sua vez formam um Program. Desse modo, o arquivo fonte de um programa só precisa ser carregado uma única vez e então ser copiado quando for colocado em execução (transformado em um processo).

Depois de carregar os arquivos e estar com os programas preparados para executar, deve-se criar um processo para cada pedido de execução, originárias do arquivo de configuração, e informar para o Manager a adição de um novo processo (através de addProcess). A adição de todos os processos deve ser feita antes do início da simulação, para que todos processos iniciem na simulação com estado ‘pronto’.

Process funciona como um descritor de processo, contém todas informações sobre cada processo (prioridade, tempos, registrador PC que indica a instrução para

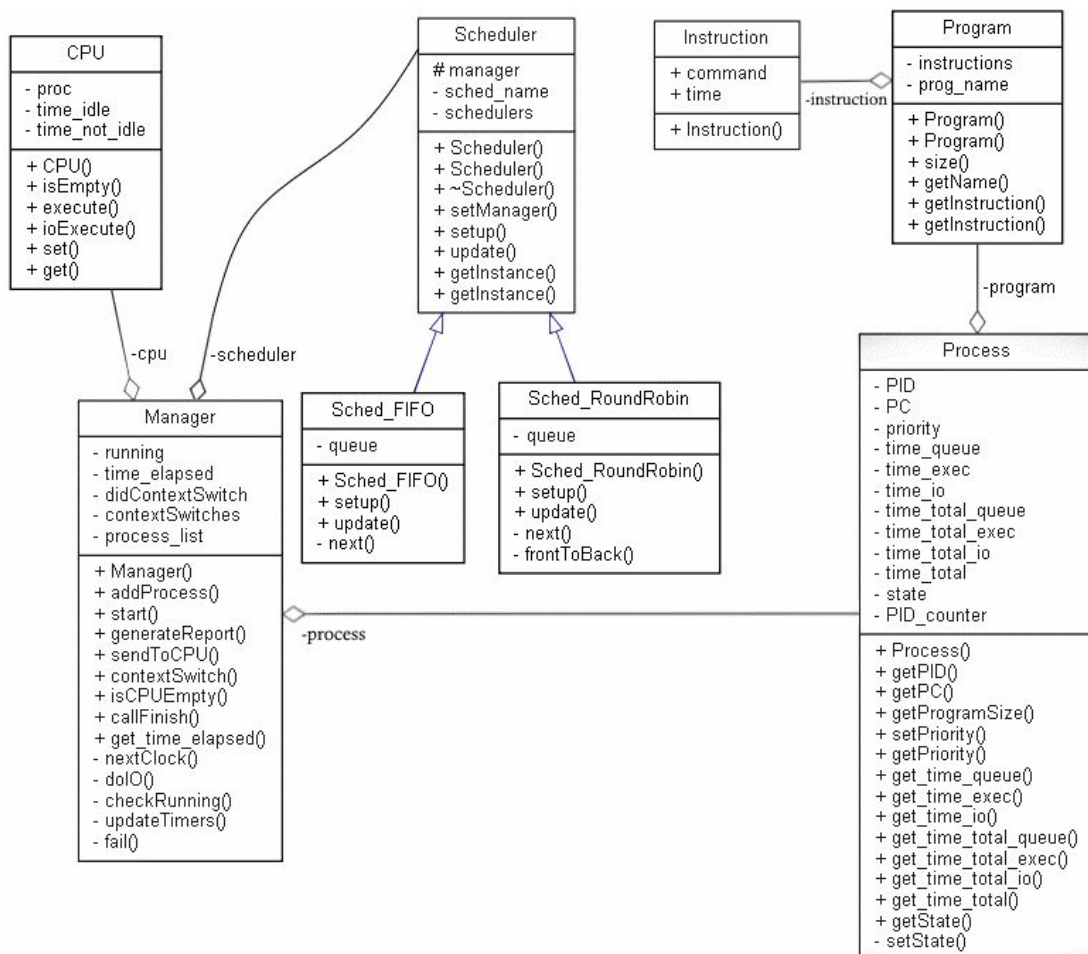
³ Código fonte do projeto disponível em <https://github.com/caio-ribeiro-pereira/SimulaRSO>

executar, o tamanho do programa e seu estado). Caso algum algoritmo de escalonamento precise de qualquer informação de um processo, deverá utilizar-se dos getters da classe, medida de segurança para que um algoritmo de escalonamento não altere as informações de um processo.

Scheduler é a classe base para os algoritmos de gerência do processador, todo algoritmo que pretende gerenciar a CPU deverá herdar de Scheduler, assim como Sched_FIFO e Sched_RoundRobin. Portanto para o usuário do *framework* a única classe que será necessária implementação é uma classe que herde Scheduler. Herdando de Scheduler, o algoritmo é automaticamente cadastrado para ser utilizado na simulação.

Manager é a classe gerente da simulação (controla a simulação), e por questões de organização e controle das ações no *framework*, apenas Manager tem controle sobre a CPU e sobre a simulação. Desse modo, tudo o que envolve a lista de processos, a CPU e dados da simulação, devem ser pedidos para Manager, de modo que qualquer algoritmo de gerência de processador deve fazer requisições para Manager, que então repassa para CPU.

Figure 1. Modelagem do Framework



A classe CPU é a parte do sistema responsável por executar as instruções dos programas e realizar a entrada e saída de cada programa (que ocorre de modo

independente, desde que o processo esteja executando I/O). Isso para que toda a troca de estados dos processos seja centralizada na classe CPU (na adição da Gerência de Memória é possível a necessidade de remodelar essa classe). Os métodos set e get, colocam e tiram respectivamente um processo na CPU. O método 'isEmpty' retorna o estado da CPU, se algum processo está executando ou não. Execute e ioExecute são métodos utilizados respectivamente por Manager para realizar a execução do processo que está na CPU e de I/O de determinado processo.

3.2 Arquivos de entrada

Baseado na modelagem, a aplicação recebe um arquivo de entrada que contém as informações sobre qual escalonador será utilizado na simulação e os programas que deverão ser executados com suas respectivas prioridades. Como GROSS é uma aplicação para terminal, o arquivo de entrada é informado após o nome do programa. O Quadro 1 mostra um exemplo de como é executado a aplicação.

Quadro 1. Exemplo de chamada pelo terminal da aplicação

```
>GROSS exemplo.txt
```

O arquivo de entrada segue o seguinte padrão (veja o exemplo no Quadro 2):

- É obrigatório a primeira palavra ser “escalonador” ou “scheduler”;
- Após a palavra obrigatória deve haver o nome de um escalonador existente. Dois escalonadores estão programados como exemplo e se chamam “sched_fifo” e “sched_roundrobin”.
- Depois do escalonador estar definido, coloca-se a lista de programas que devem ser executados e suas respectivas prioridades.

Quadro 2. Exemplo de arquivo de configuração

```
1   escalonador sched_roundrobin
2   cpuIntensive 8
3   cpuIO 5
4   ioIntensive 3
5   ioIntensive 3
6   cpuIntensive 9
```

Os “cpuIntensive”, “cpuIO” e “ioIntensive” são programas. Para definir um programa é necessário seguir algumas regras:

- Cada programa é definido por um arquivo que deve estar no diretório “programs”.
- O diretório “programs” deve estar no mesmo diretório que a aplicação GROSS.
- O nome do arquivo onde está definido o programa é o nome do programa com a extensão “.prog”, por exemplo, o programa “cpuIntensive” está definido no arquivo “cpuIntensive.prog”.

Os programas também possuem uma estrutura específica (veja o exemplo no quadro 3), esta estrutura pode ser observada a seguir:

- Cada linha é uma instrução;
- Linhas em branco são ignoradas;
- Uma instrução é definida por uma operação e o número de *clocks* (tempo de simulação) necessário para realizar esta operação;
- As operações válidas são “io” e “process”;
- Para o número de *clocks* apenas serão considerados números positivos até 32767, e ao se encontrar algum carácter inválido, o restante é ignorado, sendo considerado apenas o trecho válido;
- A operação “process” indica a quantidade de *clocks* que o programa precisa para realizar algum processamento na CPU da simulação.
- A operação “io” indica a quantidade de *clocks* que o programa ficará em I/O. Devido a isso, coloca-se o processo em estado bloqueado pelo tempo da operação. Essa operação de maneira implícita utiliza 1 *clock* de processador, o equivalente à instrução “process 1”, um pequeno custo para entrar em estado de I/O.

Quadro 3. Exemplo de programa

```
1 process 10
2 io 5
3 io 5
4 process 8
```

Como os tempos de instrução são bem definidos, é possível saber quantos *clocks* o processo que estiver executando um programa ficará utilizando o processador e quantos *clocks* ficará bloqueado. O processo que executar o programa do exemplo do Quadro 3 utilizará 20 *clocks* de processamento - 10 da primeira instrução, mais 1 (*clock* implícito) da segunda instrução, mais 1 (*clock* implícito) da terceira instrução, e por fim, mais 8 da quarta instrução – e ficará 10 *clocks* bloqueado – 5 da segunda instrução mais 5 da terceira instrução.

3.3 Funcionamento

Depois do usuário informar o arquivo de entrada, a aplicação valida-o e carrega os programas necessários para a execução. Caso algum erro ocorra nessa etapa, este é informado ao usuário. Após carregar os programas, ocorre a etapa de criação dos processos. Nessa etapa, é criado um processo na simulação para cada chamada de execução que foi especificado pelo usuário no arquivo de entrada (cada linha com o nome de um programa).

Antes de iniciar a simulação, o Gerente da Simulação inicializa o escalonador escolhido com a lista de processos que estão executando, para que o escalonador possa reorganizar a lista caso necessário – como armazenar em uma fila ou pilha.

Em seguida a simulação é iniciada e a cada *clock* são executadas várias tarefas até o fim da simulação. Essas tarefas são apresentadas a seguir:

- 1) A execução do escalonador para que este tome as decisões que foram programadas. Todas estatísticas de *clocks* da simulação são altamente influenciadas pelas tomadas de decisões do escalonador.
- 2) Atualizar os tempos da simulação.
- 3) Execução do I/O de cada processo, pois como ocorre a simulação de uma DMA e o I/O de cada processo é considerado independente, o I/O pode ocorrer em paralelo.
- 4) Execução da CPU, que pode executar uma instrução de um processo ou não executar nada, depende das decisões tomadas pelo escalonador.
- 5) Verificações se a simulação deve terminar.

Assim que a simulação termina, os resultados são gerados e apresentados na tela, mas também podem ser redirecionados para um arquivo. Caso existam muitos processos é altamente recomendado redirecionar a saída do terminal para um arquivo de texto, para não correr o risco do terminal cortar partes dos resultados.

Todos os valores relacionados a tempo estão em *clocks* (unidade de tempo de simulação). São apresentados 4 informações gerais antes das informações de cada processo, e são elas:

- *Total Simulation Time*: o tempo total da simulação.
- *CPU Idle Time*: a quantidade de *clocks* em que não houve processamento, em uma comparação entre escalonadores, quanto maior este valor, pior o escalonador.
- *CPU Not Idle Time*: a quantidade de *clocks* que o processador da simulação processou.
- *Context Switches*: mostra a quantidade de trocas de contextos que foram executadas durante a simulação.

Para cada processo é exibido um cabeçalho e os seguintes itens. O cabeçalho contém o PID e o nome do programa entre colchetes. Os dados são compostos na seguinte ordem:

- *Time Waiting*: Tempo total que o processo ficou esperando pelo processamento.
- *Time Running*: Tempo que o processo executou na CPU.
- *Time Blocked*: Tempo que o processo ficou bloqueado em I/O.
- *Life Span*: Tempo de vida do processo.
- *State*: Apresenta o estado que o processo se encontra ao terminar a simulação. O padrão é DESTROYED, pois um término bem sucedido significa que todos os processos devem ter sido finalizados, mas é possível o escalonador pedir o fim da simulação por algum motivo, por isso é apresentado o estado do processo.

No Quadro 4 há um exemplo de saída para duas execuções, cada uma com um escalonador diferente selecionado no arquivo de configuração, ambos já estão programados conforme a modelagem, à esquerda o “*sched_roundrobin*” e à direita o “*sched_fifo*”.

Quadro 4. Exemplo de saída

1	Total Simulation Time: 190	1	Total Simulation Time: 126
2	CPU Idle Time: 64	2	CPU Idle Time: 0
3	CPU Not Idle Time: 126	3	CPU Not Idle Time: 126
4	Context Switches: 0	4	Context Switches: 21
5		5	
6	Process 0 [cpuIntensive]	6	Process 0 [cpuIntensive]
7	Time Waiting = 0	7	Time Waiting = 71
8	Time Running = 50	8	Time Running = 50
9	Time Blocked = 0	9	Time Blocked = 0
10	Life Span = 50	10	Life Span = 121
11	State = DESTROYED	11	State = DESTROYED
12	Process 1 [cpuIO]	12	Process 1 [cpuIO]
13	Time Waiting = 50	13	Time Waiting = 51
14	Time Running = 20	14	Time Running = 20
15	Time Blocked = 10	15	Time Blocked = 10
16	Life Span = 80	16	Life Span = 81
17	State = DESTROYED	17	State = DESTROYED
18	Process 2 [ioIntensive]	18	Process 2 [ioIntensive]
19	Time Waiting = 80	19	Time Waiting = 24
20	Time Running = 3	20	Time Running = 3
21	Time Blocked = 27	21	Time Blocked = 27
22	Life Span = 110	22	Life Span = 54
23	State = DESTROYED	23	State = DESTROYED
24	Process 3 [ioIntensive]	24	Process 3 [ioIntensive]
25	Time Waiting = 110	25	Time Waiting = 25
26	Time Running = 3	26	Time Running = 3
27	Time Blocked = 27	27	Time Blocked = 27
28	Life Span = 140	28	Life Span = 55
29	State = DESTROYED	29	State = DESTROYED
30	Process 4 [cpuIntensive]	30	Process 4 [cpuIntensive]
31	Time Waiting = 140	31	Time Waiting = 76
32	Time Running = 50	32	Time Running = 50
33	Time Blocked = 0	33	Time Blocked = 0
34	Life Span = 190	34	Life Span = 126
35	State = DESTROYED	35	State = DESTROYED

4. Validação

Nessa seção serão mostrados alguns exemplos de utilização e comparados vantagens e desvantagens com os outros sistemas citados na Seção 2.

O *software* pode ser utilizado com fins educativos, para verificar os efeitos na mudança do algoritmo de escalonamento, como comparar o algoritmo FIFO contra o algoritmo Round Robin, além de abrir a possibilidade para eventuais testes de desempenho. Pode ser útil em aulas de sistemas operacionais onde deseja-se lições práticas de programação, nessa situação o professor pode pedir para que alunos programem um escalonador seguindo determinadas regras.

Além de ser uma ferramenta para a gerência de processador, o *framework* pode ser utilizado para aprendizado da linguagem de programação C++, como também para criação de estruturas de dados que armazenam a lista de processos com determinadas regras especiais de modo que estas estruturas possam ser reutilizadas entre os algoritmos.

O objetivo de construir um *framework* para a experimentação e ensino de gerências de sistemas operacionais, que inicialmente facilite a construções de algoritmos para a gerência de processador foi alcançado.

Uma comparação entre GROSS e as ferramentas comentadas na Seção 2 pode ser visualizada na Tabela 1. Como pode ser visto, dentre as ferramentas, GROSS é a mais portátil (no sentido de não precisar de executáveis externos, como servidores) e com o ideal de programação.

SimulaRSO segue alguns padrões de programação (como a generalização dos algoritmos) que implica em facilitar a adição de novos recursos. Porém, essas facilidades foram voltadas para o desenvolvimento da ferramenta em si. Isso pode ser observado já que esta ferramenta foi construída no intuito de auxiliar em aulas de sistemas operacionais a distância, portanto, seu objetivo não é ser expandida.

O SOsim é de fácil utilização se comparado com GROSS, já que possui interface gráfica, porém não possui código fonte disponível (no tempo deste artigo), por mais que o site do projeto indique uma futura divulgação, então acaba sendo inviável para a finalidade de programação.

O sistema BACI foi construído para aprendizagem de sistemas paralelos e distribuídos, portando não entra diretamente no escopo da gerência de processador, mas com o mesmo foi possível se obter várias ideias de implementação como a utilização de arquivos de configuração e programas com a possibilidade de alteração.

Tabela 1. Comparação de recursos entre as ferramentas relacionadas

	GROSS	SOsim	BACI	SimulaRSO
Código fonte aberto	Sim	Não	Sim	Sim
Gerência de Processador	Sim	Sim	Não	Sim
Gerência de Memória	Não	Sim	Não	Sim
Simulação	Sim	Sim	Sim	Sim
Arquivos configuram a simulação	Sim	Não	Sim	Não
Programas customizáveis	Sim	Não	Sim	Não
Facilita novos recursos	Sim	Não é possível	Não	Sim
Necessidade de Servidor	Não	Não	Não	Sim

5. Considerações finais

Com a utilização deste *framework* espera-se que a prototipação e construção de algoritmos para as gerências de processador de um sistema operacional seja muito mais simples do que a construção direta em um sistema para uso final. Também que seja

comum nas aulas da disciplinas de Sistemas Operacionais assuntos sobre implementação dos algoritmos envolvidos, não apenas na gerência do processador.

Por mais que o *software* não tenha a implementação da Gerência de Memória, com a base já existente, a adição dessa gerência precisará menos esforço se comparado com a criação a partir do zero. Como exemplo, a adição de comandos como “alloc” (para alocar memória) e “free” (para liberar memória) para os programas, pois toda a estrutura para as instruções já está construída. Outro ponto importante é a necessidade de adaptar a lógica envolvida nos ciclos da simulação com a adição da gerência de memória.

No GROSS buscou-se seguir técnicas e mecanismos que ofereçam maior desempenho para a aplicação como um todo, e também boas práticas de programação em C++, como a não utilização de ponteiros, new e delete.

Embora a realização dessa simulação por software tenha sido uma simplificação da realidade, o desenvolvimento mostra toda a complexidade que envolve a área de Sistemas Operacionais, inclusive nas áreas mais básicas como gerência de processador e de um escalonador.

Seria de grande interesse o aumento da complexidade da simulação, visando representar um sistema operacional com maiores detalhes e não apenas a gerência de processador, portanto como trabalhos futuros temos:

- Uma gerência de memória, que inclua o acesso à memória RAM;
- Possibilitar a entrada dinâmica de novos processos durante a simulação;
- Custo da troca de contexto de cada processo, durante o escalonamento;
- Internacionalização do programa, priorizando inglês e português;
- Resultados da execução oferecendo vários formatos, como xml e csv, visando facilitar o uso da saída para ferramentas de comparação e geração de gráficos;
- Criação de uma interface gráfica amigável para usuários mais leigos.

Referências

BACI, BACI Docs. Disponível em:

<<http://inside.mines.edu/~tcamp/baci/bacidocpdf.tar.gz>> Acesso em: 5 mar. 2014.

MAIA, Luiz P. SOsim: Simulador para o ensino de sistemas operacionais. (2001). 97f. Dissertação (Mestrado em Ciências em Informática) – Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2001. Disponível em: <http://infocao.dominiotemporario.com/doc/sosim_tese.pdf> mar. 2014.

OPEN SOURCE INICIATIVE. MIT License. Open Source Initiative, (2014). Disponível em: <<http://opensource.org/licenses/MIT>>. Acesso em: 17 nov. 2014.

STALLINGS, W. (2012), Operating Systems – Internals and Design Principle. 7 Edição, Pearson Education.

Desenvolvimento e execução de jogos eletrônicos através do recurso Canvas do HTML5

Alisson Alberti Tres¹, Lisandro Lemos Machado¹

¹Instituto Federal de Educação, Ciência e Tecnologia Sul-Rio-Grandense (IFSUL)
Caixa Postal 99064-440 – Passo Fundo – RS – Brasil

alisson.tres@gmail.com, lisandro.machado@passofundo.ifsul.edu.br

Abstract. *Electronic games stands out as one of the current most popular means of entertainment. These complex applications carry a great potential to be applied in many different fields. Among the variety of technologies available to develop games, HTML5 is becoming an increasingly prominent platform to create and execute games. This paper aims to present the functionalities that enable executing games through this technology and the resources that it offers to develop this kind of application.*

Resumo. *Jogos eletrônicos, normalmente referidos como games, vem se destacando como um dos meios de entretenimento mais populares atualmente. São aplicações complexas que carregam um grande potencial para serem aplicadas em diversas áreas. Dentre a variedade de tecnologias disponíveis para realizar o desenvolvimento de games, o HTML5 tem se tornado cada vez mais proeminente como uma plataforma para se criar e executar jogos. Através deste artigo, pretende-se apresentar as funcionalidades que viabilizam a execução de jogos através desta tecnologia e os recursos oferecidos por ela para que se possa realizar o desenvolvimento deste tipo de aplicação.*

1. Introdução

Os jogos eletrônicos, comumente referidos através do termo inglês “games”, se destacam como sendo um dos meios de entretenimento mais populares existentes atualmente. O número de pessoas que recorrentemente escolhem os games como forma de entretenimento é crescente, sendo possível executar esse tipo de aplicação em diversos tipos de dispositivos distintos, dentre eles: consoles de vídeo game, os quais são dedicados para aplicações deste tipo, computadores e também em smartphones e tablets.

Tal popularização tornou possível que estes jogos fossem observados como sendo mais do que apenas uma mera forma de diversão. Novak (2010) observou as diferentes motivações e benefícios existentes no ato de jogar um game, apontando: a satisfação presente ao se competir com outros jogadores; os conhecimentos que podem ser adquiridos através do game; a capacidade de domínio e conquista que o jogador poderá estabelecer dentro do universo do jogo; e o aspecto de escapismo das tensões e desafios existentes na vida real que estes são capazes de proporcionar.

No entanto, realizar o desenvolvimento de jogos pode ser considerado como sendo uma tarefa com um certo grau de complexidade. O planejamento de um game requer a criação de uma extensa documentação e modelagem para cada elemento que irá compô-lo: seus objetivos, sua forma de jogar, seus personagens, seus cenários e inclusive um enredo para interligar todos estes elementos. Não obstante, a criação de tais aplicações também requerem um certo grau de familiaridade com o hardware no qual o jogo será executado e envolvem o emprego de diferentes tipos de conhecimentos da computação. O esforço mínimo necessário para criar um game envolve o tratamento de comandos do jogador recebidos através de um dispositivo de entrada (como um joystick, teclado e mouse ou mesmo uma tela tátil), e a renderização de elementos gráficos em uma tela para contextualizar ao jogador a situação atual da aplicação. O desenvolvimento de jogos mais complexos ainda podem envolver a aplicação do carregamento e detecção de colisão entre objetos dentro do jogo, reprodução de áudio e comunicação via rede [Bassa et al. 2007].

Ao realizar o desenvolvimento de um game, é recomendável trabalhar com uma ferramenta capacitada que ofereça facilidade para o processo de desenvolvimento e recursos para tratar de seus diversos requisitos tecnológicos. Em meio à todas as tecnologias existentes atualmente para o desenvolvimento de jogos eletrônicos, a quinta versão da linguagem HTML vem ganhando destaque neste meio. Isso se deve à sua proposta de viabilizar um melhor tratamento nativo de recursos multimídia dentro dos navegadores de Internet. E por meio deste artigo, pretende-se abordar os recursos presentes no HTML5 que o viabiliza como uma plataforma para desenvolver e executar jogos eletrônicos.

2. Tecnologias utilizadas para o desenvolvimento de games

Jogos eletrônicos são aplicações computacionais e, como todos os demais tipos de aplicações computacionais, são construídos a partir da escrita de código em uma determinada linguagem de programação. Atualmente, as linguagens de programação mais populares dispõem dos recursos que contemplam diversos dos requisitos básicos esperados de um bom jogo eletrônico como gráficos, música e a construção de sua jogabilidade.

A linguagem C++ é conhecida por oferecer um bom equilíbrio entre programação de baixo nível e programação à nível de Sistema Operacional. Linguagens como Java e C# se apresentam como excelentes alternativas por apresentarem uma grande quantidade de bibliotecas padrões que muito facilitam o trabalho de programação. Há também as linguagens de script que podem ser executadas tanto de maneira independente como também dentro de outros programas e outras linguagens de programação. Python, Ruby e Lua são exemplos de linguagens de script, criadas visando proporcionar simplicidade, dinamismo, objetividade e portabilidade à prática de programação.

Em todas estas linguagens podemos encontrar os recursos que oferecem ao programador a capacidade de manipular os elementos que compõem o game. Na maioria dos casos, no entanto, estas linguagens não são puramente e diretamente utilizadas no processo de desenvolvimento do jogo propriamente dito. O que costuma ocorrer é o emprego de softwares denominados *engines* (ou motores, em português) de

desenvolvimento de jogos, por serem capazes de abstrair várias das etapas do desenvolvimento através de editores e ferramentas gráficas [Feijó, Pagliosa, Clua 2006].

A essência do uso de *engines* para o desenvolvimento de jogos está no fato de que estas visam suprir grande parte da demanda computacional, integrando diversas técnicas frequentemente necessárias nos jogos, simplificando muito a criação de um novo game [Bessa et al 2007]. Uma determinada engine pode trabalhar em cima de uma específica linguagem de programação e funcionar de maneira a, estritamente, necessitar que o seu usuário possua conhecimentos de programação, ao passo que outra consegue abstrair tais conceitos de tal modo que o processo de construir um jogo possa se resumir a simplesmente selecionar e jogar elementos na tela.

2.1 HTML5 como ferramenta para desenvolver jogos

O HTML é conhecido como sendo a tecnologia padrão usada para a criação de páginas da Web. Embora essa tecnologia tenha se estagnado por um longo período de tempo sem atualizações, as entidades mantenedoras da linguagem, W3C e WHATWG, têm trabalhado desde 2008 em sua quinta versão. Tal atualização é formulada visando propor uma completa reformulação da linguagem para aumentar as suas capacidades e, através da utilização conjunta de tecnologias como Javascript e CSS, possibilitar a geração de páginas web com mais funcionalidades e melhor usabilidade.

Seguindo as perspectivas de uma web mais centrada em recursos multimídia e adaptável para dispositivos móveis, as novidades do HTML5 objetivam torná-lo o mais capacitado e independente o possível. Isso se contrapõe a um passado no qual era necessário instalar aplicativos prioritários feitos por terceiros, como o Adobe Flash por exemplo, para possibilitar a existência de páginas web mais dinâmicas e interativas.

E é através de uma das novas funcionalidades propostas em meio a essa reformulação que a tecnologia do HTML tornou-se viável para o desenvolvimento de jogos eletrônicos: o *Canvas*. Canvas em inglês significa tela, em referência às telas utilizadas por artistas para realizar pinturas. Assim como artistas usam sua tela para jogar cores e realizar desenhos, Silva (2011) explica que o elemento Canvas possibilita aos desenvolvedores web delimitar uma área para criação dinâmica de imagens, como gráficos estáticos, jogos (como pode ser visto na Figura 1) e gráficos dinâmicos e imagens em geral criadas com linguagem de programação dinâmica.

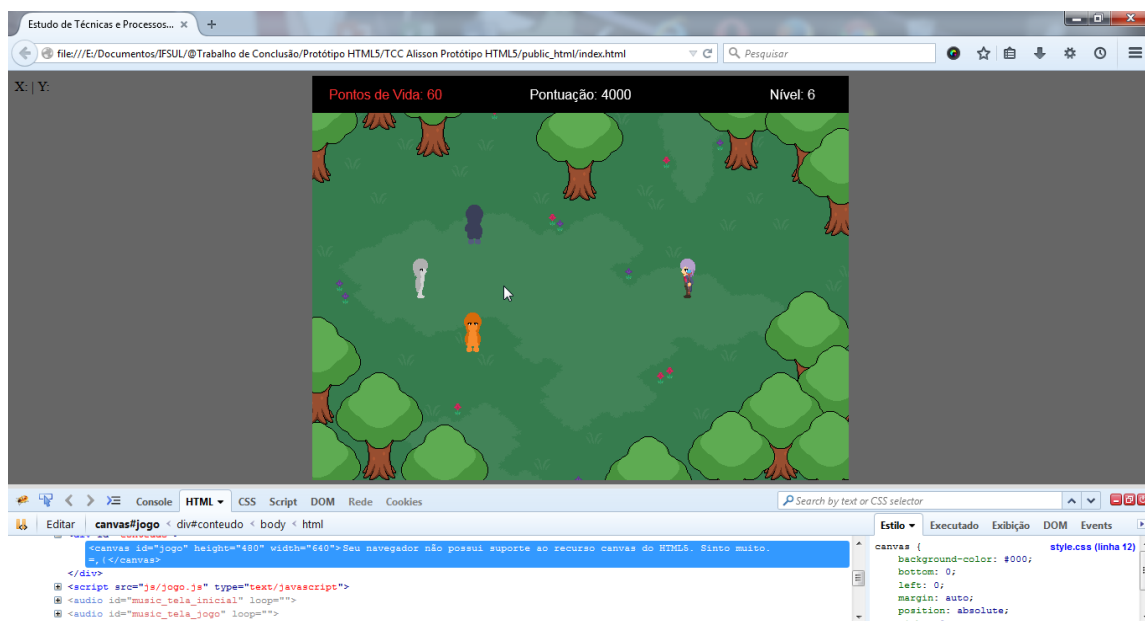


Figura 1. Exemplo de um protótipo de um jogo construído em HTML5 e Javascript rodando em um navegador de Internet.

O que torna atrativa a ideia de sua utilização como uma plataforma de desenvolvimento de jogos é a perspectiva de que qualquer desenvolvedor web possui à sua disposição uma ferramenta para produzir games. Com capacidade suficiente para criar até mesmo games mais complexos, sem a necessidade da aquisição de licenças de uso e com um instantâneo suporte multiplataforma, graças ao seu caráter de linguagem interpretada através de softwares amplamente disponíveis em vários tipos de dispositivos.

3. Funcionamento de um jogo eletrônico rodando em HTML5

A essência dos games em HTML5 é que pelo intermédio do recurso Canvas aliado à programação em Javascript é possível gerar e manipular elementos gráficos dentro dos softwares que são utilizados para acessar a Internet. Através da API do Canvas os navegadores podem, em tempo real, realizar a renderização de diversos tipos de elementos gráficos, sendo capaz de: desenhar texto; configurar e desenhar formas geométricas como retângulos, círculos, linhas e outros tipos de formas através do uso de vetores; e desenhar imagens a partir da importação de arquivos presentes no servidor.

Para que se possa gerar imagens na tela de um Canvas, é preciso determinar o contexto de sua superfície de desenho: se iremos trabalhar com elementos gráficos bidimensionais (2D) ou tridimensionais (3D). Isso é determinado instanciando um objeto de contexto através da chamada do método *getContext()*. Esta variável permitirá realizar a chamada dos diversos métodos existentes para gerar elementos gráficos dinamicamente dentro de um Canvas.

Uma vez conseguindo imprimir as imagens do seu jogo na tela do Canvas, pode-se prosseguir com o trabalho de programação em Javascript para realizar a manipulação desses elementos gráficos, sendo possível programar para gerar animações e reconhecer

cliques do mouse ou o pressionar de uma determinada tecla do teclado. Consequentemente, isso torna o Canvas passível de ser programado para exibir a movimentação do personagem em um jogo, estabelecer um sistema de colisão entre os objetos ali presentes e implementar a simulação dos elementos necessários para o funcionamento de sua jogabilidade.

A construção de todo o restante da lógica desta aplicação envolve combinar a API do Canvas com os demais artifícios do HTML5 e da linguagem Javascript. Para que possamos ter um jogo em HTML5, é necessário que existam uma série de rotinas para estabelecer o fluxo do jogo, controlar os estados e as telas da aplicação e executar a lógica da jogabilidade.

Os métodos `addEventListener()` e `removeEventListener()` são particularmente úteis por atribuir a algum elemento presente em uma página da Internet uma rotina que deverá ser executada caso haja a ocorrência de um determinado evento, como um clique do mouse, por exemplo. Outro método relevante é o `setInterval()`, que funciona de maneira similar, mas executando uma determinada função continuamente num intervalo de tempo, terminando apenas quando `clearInterval()` é chamado. Outro exemplo interessante é o uso do marcador *Audio*, também introduzido no HTML5, que possibilita tratar a sonoplastia do game, utilizando comandos em Javascript para carregar uma música de fundo e executar efeitos sonoros em determinados momentos no decorrer da execução do jogo.

4. Aplicando o Canvas no desenvolvimento de um game em HTML5

Para que o Canvas possa ser utilizado para o propósito de se executar um jogo, é necessário que ele seja constantemente atualizado dentro de um intervalo de tempo. Isso pode ser facilmente realizado através de variáveis, funções e objetos em Javascript. Uma peculiaridade relativa ao funcionamento do Canvas a ser observada é que este não guarda nenhuma informação ou qualquer referência aos elementos gráficos nele contidos.

O Canvas literalmente é apenas um vetor bitmap que simplesmente recebe e exibe imagens e elementos gráficos dentro de si. Por conta disto, não seria possível apagar ou modificar um elemento específico. Ao se inserir novos elementos em um Canvas, estes são simplesmente posicionados por cima de seja lá o que já exista dentro dela. E a única forma de se retirar um recurso gráfico de uma canvas é limpado tudo que está nela presente.

Cabe, então, ao programador fazer o serviço de manualmente estabelecer e controlar todos os dados referentes ao posicionamento ou, no caso de um jogo eletrônico, dos comportamentos que são visualizados através das imagens presentes no Canvas. Levando em consideração tudo isso, podemos pensar a lógica de execução de um jogo em HTML5 da seguinte forma: Nosso código Javascript terá variáveis que guardam informações como o estado atual da aplicação ou as características do personagem controlado pelo jogador (a quantidade de pontos de vida, o nível atual, sua pontuação ou as imagens que deverão aparecer no Canvas, por exemplo).

Enquanto a aplicação estiver no estado de jogar, é preciso executar uma rotina para verificar se o jogador interagiu com a aplicação, verificar o estado em que cada

objeto existente dentro do jogo se encontra e chamar o objeto de contexto para exibir os gráficos na tela do Canvas. Essa rotina permanecerá constantemente em execução enquanto a condição que marcaria o fim do jogo não for alcançada.

Nesse fluxo, deve haver uma cronologia para se renderizar os elementos gráficos na tela, pois os mais novos sempre irão sobrescrever em cima dos já existentes. Normalmente, o primeiro elemento a ser gerado seria o cenário. Em seguida poderemos inserir a imagem correspondente ao jogador. Os próximos elementos da lista seriam os possíveis inimigos e objetos com os quais jogador poderá interagir. E neste momento poderia ser necessário realizar algumas verificações, como por exemplo: se o jogador tiver registrado o comando de ataque e o ataque tiver colidido com um inimigo, deve-se subtrair um determinado valor dos pontos de vida deste inimigo. Se os pontos de vida do inimigo chegarem a zero, este inimigo deverá desaparecer, portanto não há mais a necessidade de se renderizar a sua imagem durante este fluxo de execução.

Pode parecer muito, mas com a tecnologia que existe atualmente consegue-se tranquilamente executar essa rotina de 30 a 60 vezes em um único segundo, criando assim a tão desejada ilusão de movimento somada à interação com o usuário que caracteriza os jogos eletrônicos. Com um fluxo de jogo estabelecido e uma rotina para atualizar o posicionamento de cada elemento gráfico do jogo dentro de um intervalo de tempo, o programador poderá progredir construindo a lógica da jogabilidade, estabelecendo a interação do jogador com a aplicação e dos objetos presentes no game entre si.

6. Considerações Finais

Olhar para os rumos que tem sido tomados no desenvolvimento do HTML5 e da linguagem Javascript é nada menos do que fascinante. Trata-se de um reflexo do quanto a tecnologia da Internet se desenvolveu nos últimos anos. Não apenas na perspectiva do puro avanço tecnológico, mas do impacto na cultura e na maneira como utilizamos a informática. Os Navegadores de Internet já serviram apenas para carregar conteúdos de texto por intermédio do HTML presente em um site. Hoje, eles podem ser o palco para a execução de qualquer tipo de aplicação que se possa imaginar, inclusive os tão adorados jogos eletrônicos.

Para Harris (2013) as vantagens oferecidas pelo HTML5 para o desenvolvimento de jogos seriam: a ampla disponibilidade e gratuidade da ferramenta; sua maior acessibilidade para iniciantes em programação, se comparada com linguagens como o C++; o vasto público-alvo que um jogo desenvolvido em HTML5 pode ser alcançado em virtude do fato deste não precisar de plug-ins adicionais, funcionando apenas com as capacidades já existentes nos navegadores de Internet modernos; e a facilidade de criação de jogos para dispositivos móveis (como pode ser visto através da Figura 2). O HTML5 possui a capacidade de viabilizar o desenvolvimento de games para todos, permitindo que o estudo do design de games e a criação de jogos eletrônicos possam ser praticados por uma maior parcela de pessoas.

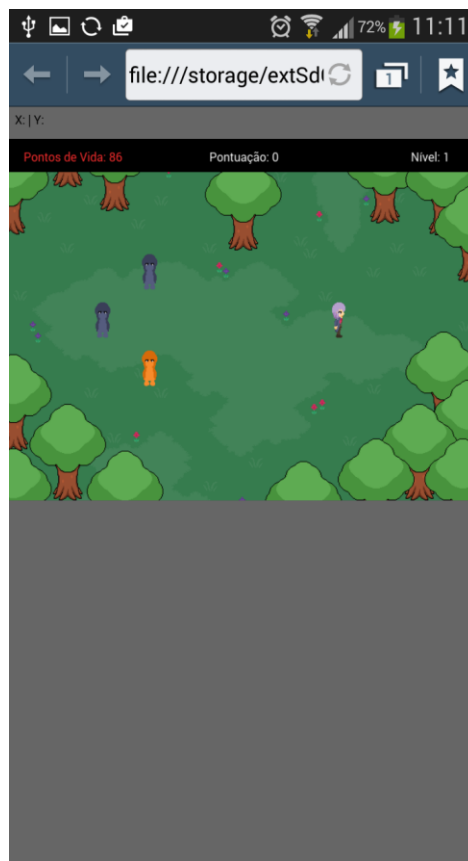


Figura 2. Exemplo de um protótipo de jogo construído em HTML5 e Javascript rodando no navegador padrão de um dispositivo Android.

Ainda que esta tecnologia por si só ofereça os recursos necessários para desenvolver jogos, fazê-lo utilizando apenas o Canvas do HTML5 e Javascript ainda faz do desenvolvimento de jogos uma atividade complexa. Além da lógica do game, também seria necessário elaborar uma estrutura de execução contínua para reproduzir na tela o estado atual da aplicação, controlar manualmente todos os recursos do jogo e garantir boa performance ao tentar executar tudo isso simultaneamente.

Por conta disso, o uso de frameworks ou engines para auxiliar no processo de desenvolvimento é muito recomendável. Através delas os requisitos técnicos para se rodar um jogo estão encapsulados, de modo que o desenvolvimento possa ser focado inteiramente na lógica que irá compor o game e no funcionamento de sua jogabilidade, além de também oferecerem ferramentas para realizar atividades mais complexas. Dada a relevância atual do HTML5, há um número cada vez mais crescente de frameworks para se criar jogos dentro desta plataforma, muitos deles gratuitos e de código-aberto, citando como exemplos interessantes para serem estudados a *Quintus Engine*, *CreateJS*, *three.js*, *Phaser* e *Panda.js*. Observa-se também que o suporte a essa tecnologia também está presente em engines mais robustas como o Game Maker:Studio e o Construct 2. Projetos criados nestas ferramentas são passíveis de serem instantaneamente compilados para serem executados de forma nativa em navegadores de Internet, sem a necessidade se instalar plug-ins proprietários desenvolvidos por terceiros. Contemplando tamanha variedade de possibilidades oferecidas, é evidenciado o potencial que as tecnologias do HTML5 possuem para esta área.

Referências

- Bessa, A. et al. (2007). “O Desenvolvimento de um Motor Multiplataforma para Jogos 3D”. <http://www.inf.ufrgs.br/~cebbezerra/publications/bessa2007dmm.pdf>, Agosto.
- Feijó, B., Pagliosa, P., Clua, E. (2006). “Visualização, Simulação e Games”. In: Anais da XXIV Jornada de Atualização em Informática do Congresso da Sociedade Brasileira de Computação, Campo Grande. <http://www.lbd.dcc.ufmg.br/colecoes/jai/2006/004.pdf>, Agosto.
- Harris, A. (2013). “HTML5 Game Development For Dummies”. John Wiley & Sons, Inc., New Jersey.
- Novak, J. (2010). “Desenvolvimento de Games”. Cengage Learning, São Paulo.
- Silva, M. S. (2011). “HTML5 A linguagem de marcação que revolucionou a web”. Novatec Editora , São Paulo.

Desenvolvimento de Uma Aplicação Interativa para Visualização e Análise de Dados

Samantha Rico Gonçalves¹

¹ Instituto Federal Sul-Rio-Grandense (IFSul) – Câmpus Passo Fundo – RS – Brasil

sah.rico@gmail.com

Abstract. *This article presents a partial account of the development of the final project of the course of Technology on System for the Internet with as objective the development of an interactive visualization and data analysis application, using the R language and Shiny package as key technologies.*

Resumo. *Este artigo apresenta o relato parcial sobre o desenvolvimento do trabalho de conclusão de curso de Tecnologia em Sistemas para Internet, que possui como objetivo o desenvolvimento de uma aplicação interativa de visualização e análise de dados utilizando a linguagem R e o pacote Shiny como principais tecnologias.*

1. Introdução

A tecnologia tem papel fundamental em diversas áreas do mundo e está sempre em busca de inovações e formas de facilitar a vida das pessoas. Atualmente a geração de dados sobre os mais diversos temas é uma realidade, e se tornou cada vez mais importante a organização e análise desses dados.

Como exemplo, é possível citar alguns sites governamentais como o Portal Brasileiro de Dados Abertos e o Instituto de Pesquisa Econômica Aplicada, entre outros, que disponibilizam diversos tipos de dados de forma pública.

Porém, não é útil possuir um grande número de dados disponíveis, se não for possível extrair informações deles. Assim como Idreos et al. (2015) diz, a exploração de dados tem como objetivo a extração eficaz das informações dos dados mesmo que o usuário não saiba o que está procurando. E esse é o propósito principal do desenvolvimento da aplicação de visualização e análise de dados, permitir que os usuários extraiam de forma eficaz e interativa o conteúdo informacional dos dados.

O desenvolvimento da aplicação será realizado utilizando a linguagem de programação R, que irá permitir através de suas características estatísticas, que seja possível a geração de diferentes gráficos e informações estatísticas a partir do conjunto de dados, sendo utilizado também o pacote Shiny, que permitirá o desenvolvimento da aplicação para a web.

Neste artigo é apresentado, portanto, o objetivo principal que é o desenvolvimento da aplicação de visualização e análise de dados, onde será descrito como será a aplicação e qual a função das tecnologias utilizadas para seu desenvolvimento. Em seguida, será

apresentada algumas aplicações similares, a metodologia utilizada para o desenvolvimento e os resultados esperados.

2. Objetivos

O objetivo geral do trabalho é o desenvolvimento de uma aplicação que permita a visualização e a análise de um conjunto de dados de forma interativa utilizando a linguagem R e o pacote Shiny.

2.1. Objetivos Específicos

- Realizar uma pesquisa sobre as tecnologias utilizadas no desenvolvimento.
- A aplicação deve ser interativa e de fácil utilização.
- Realizar upload de arquivos de dados;
- Gerar gráficos e informações estatísticas;
- Realizar a análise e projeto da aplicação;
- Realizar testes com usuários;

3. Sobre a Aplicação

A aplicação de visualização e análise de dados encontra-se atualmente em fase de desenvolvimento. Pretende-se com o seu desenvolvimento facilitar e agilizar o processo de análise e visualização de dados, tornando-a mais interativa possível, para que se consiga melhores resultados na extração de informações relevantes de um conjunto de dados.

Um grande diferencial da aplicação em relação a outras similares, que serão apresentadas mais adiante no artigo, é a possibilidade de utilizar qualquer tipo de dados, independente do tema das informações, por exemplo, o usuário poderá utilizar um arquivo de dados padrão no formato .csv ou .txt de dados econômicos, tanto quanto poderá utilizar um conjunto de dados geográficos e assim por diante. Ou seja, a aplicação terá a capacidade de ser genérica ao ponto de não precisar que os dados sejam de assuntos específicos, apenas que sigam um padrão de organização.

Através da aplicação o usuário poderá carregar um arquivo de dados (.csv ou .txt), visualizá-los na tela, gerar diferentes tipos de gráficos (boxplot, plot e histograma) e informações estatísticas (média, mediana, menor valor...), além da possibilidade da realização do download dos gráficos gerados no formato pdf.

Na figura 1 é possível visualizar o diagrama de caso de uso da aplicação onde se demonstra as possibilidades de ações que o usuário poderá realizar com a utilização da aplicação.

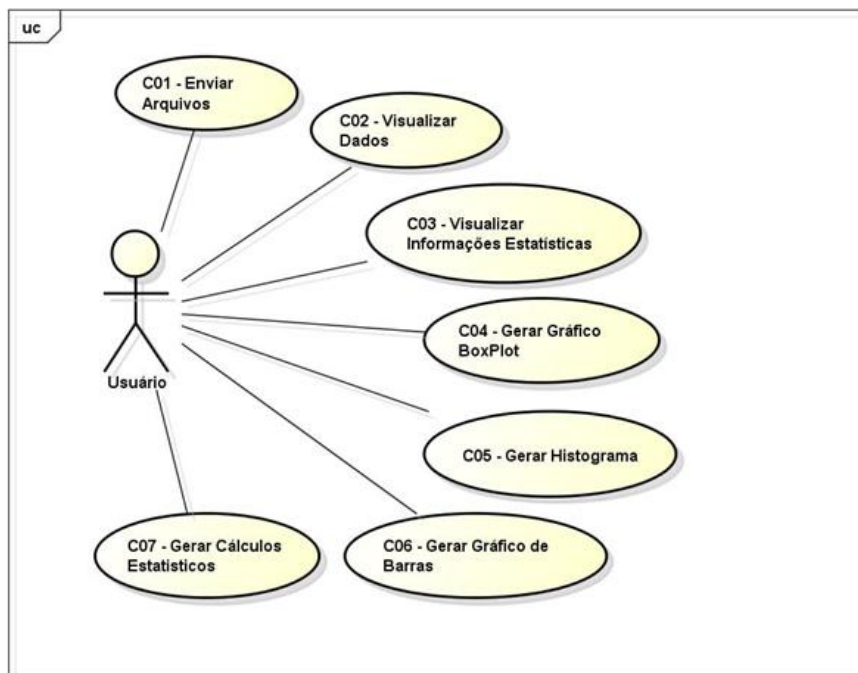


Figura 1 - Diagrama de Caso de Uso

Na figura 2 é possível ver uma proposta inicial da interface do sistema onde acontece o carregamento de um arquivo de dados e a exibição dos dados na tela.

Visualização e Análise de Dados

Escolha o arquivo

Choose File ...suitsBrusoneIncidence.csv

Upload complete

☒ Cabeçalho

Separador

☒ Vírgula

☐ Ponto e Vírgula

☐ Tabulação

Aspas

☐ Nenhuma

☒ Aspas Duplas

☐ Aspas Simples

Opções Gráfico Boxplot:

x:

Cultivar

y:

Cultivar

Visualização dos dados

Gráfico BoxPlot

	Cultivar	Local	Data.Semeadura	Valor
1	BRS-LOURO	CIANORTE	26/03/2001	0.00
2	BRS-LOURO	CIANORTE	31/03/2001	0.00
3	BRS-LOURO	CIANORTE	05/04/2001	0.00
4	BRS-LOURO	CIANORTE	10/04/2001	0.00
5	BRS-LOURO	CIANORTE	15/04/2001	0.00
6	BRS-LOURO	CIANORTE	20/04/2001	0.00
7	BRS-LOURO	CIANORTE	25/04/2001	0.00
8	BRS-LOURO	CIANORTE	30/04/2001	0.00
9	BRS-LOURO	CIANORTE	05/05/2001	0.00
10	BRS-LOURO	CIANORTE	10/05/2001	0.00
11	BRS-LOURO	CIANORTE	26/03/2002	0.00
12	BRS-LOURO	CIANORTE	31/03/2002	0.00
13	BRS-LOURO	CIANORTE	05/04/2002	0.00
14	BRS-LOURO	CIANORTE	10/04/2002	0.00
15	BRS-LOURO	CIANORTE	15/04/2002	0.00
16	BRS-LOURO	CIANORTE	20/04/2002	0.00

Figura 2: Upload e Exibição de Dados

No exemplo da figura 2, o usuário faz o carregamento de um arquivo de dados e escolhe as opções que melhor se adequam as configurações dos dados no arquivo enviado, informando o separador, se possuem aspas e se os dados possuem ou não cabeçalho.

A aplicação de visualização e análise de dados será disponibilizada através da web, atualmente um protótipo da aplicação já está disponível nos servidores do Instituto Federal Sul-rio-grandense Campus Passo Fundo, no endereço a seguir: <http://187.7.88.182:6838/apps/appAnaliseDados/>.

3.1. Linguagem R

Para o desenvolvimento da aplicação será utilizada a linguagem R que segundo Matloff et al. (2011, p. 21) “É uma linguagem de script para manipulação de dados e análise estatística”. Portanto, voltada para fins científicos, matemáticos e estatísticos.

O R é um software livre, desta forma, possui código fonte aberto e pode ser modificado ou implementado com novos recursos por seus usuários e conta com um grande número de colaboradores das mais diversas áreas. (SOUZA et al., 2008, p. 5)

A linguagem R foi criada em 1996 pelo neozelandês Ross Ihaka e pelo canadense Robert Gentleman inicialmente como um projeto de pesquisa. Foi baseada em duas outras linguagens, sendo elas a linguagem S e a Scheme. (MELLO et al., 2013, p. 16)

Segundo Mello et al. nos dias atuais o R está sob constante desenvolvimento por um grupo chamado R Core Team. E continua em grande crescimento com a colaboração dos seus usuários. (2013, p. 16)

A aplicação vai ser desenvolvida através do RStudio que é uma IDE (Integrated Development Environment ou Ambiente Integrado de Desenvolvimento) gratuita que possui versões para diferentes sistemas operacionais. Essa IDE oferece uma interface que integra o R e um avançado editor de textos voltado para edição de comandos. (MELLO et al., 2013, p. 24)

3.2. Pacote Shiny

Segundo Radu et al. (2014, p. 13) o R não serve apenas para análise quantitativa, mas é usado também para construir aplicativos de desktop e aplicações web. E foi para permitir o desenvolvimento de aplicações web interativas com a utilização do R que surgiu o pacote Shiny.

O pacote Shiny permite que os usuários transformem suas análises em aplicações web interativas que podem ser acessadas por diversos outros usuários. E o mais interessante é que para desenvolver a aplicação, não é necessário possuir um conhecimento prévio em HTML, CSS ou Javascript, já que toda a programação é realizada utilizando a linguagem R. (RADU et al., 2014, p. 13)

3.2. Aplicação Similar

Existem alguns trabalhos que são similares ao trabalho que está sendo desenvolvido, entre elas é possível citar o trabalho “*Seed: a user-friendly tool for exploring and visualizing microbial community data*” (Seed: Uma ferramenta amigável para exploração e visualização de dados da comunidade microbiana), neste trabalho, é desenvolvido uma

aplicação de código aberto que permite que pesquisadores da comunidade microbiana explorem visualmente os dados. (BECK, 2014, p. 1)

Segundo Beck (2014, p. 1), o Seed é projetado para permitir muitas análises e visualizações diferentes, incluindo componentes principais e análise de coordenadas, agrupamento hierárquico, gráficos de dispersão, gráficos de barra e mapas de calor. Essa aplicação foi desenvolvida com a utilização da linguagem R e do pacote Shiny.

Na figura 3 é possível visualizar a interface principal da aplicação Seed.

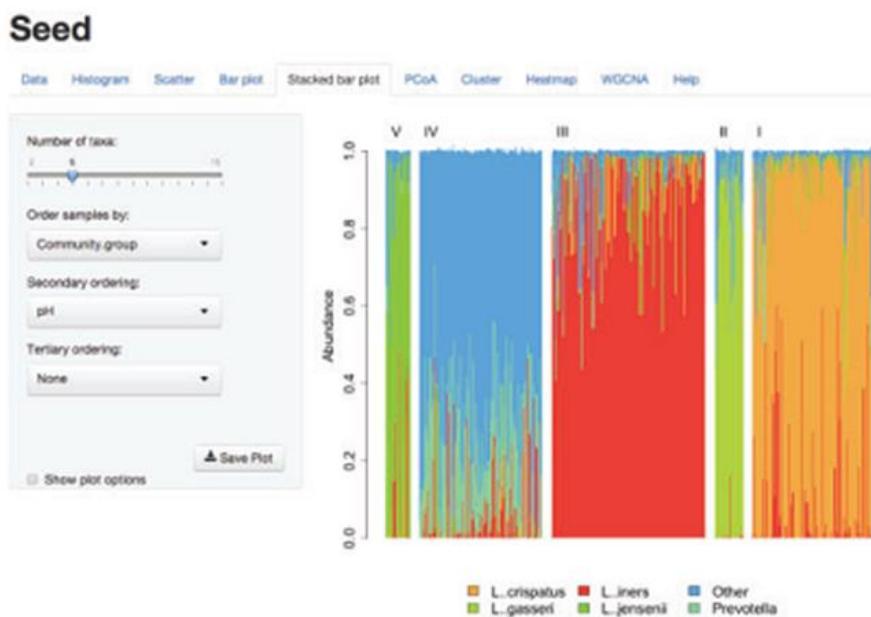


Figura 3 - Interface da Aplicação Seed

Outro trabalho relacionado que pode ser citado é o artigo “Integrando visualização e análise de dados em sistema de gerenciamento de dados de biodiversidade” realizado na Universidade Federal do Rio Grande do Sul.

Neste trabalho foi desenvolvido um sistema chamado de TaxonomyBrowser que tem como objetivo, conforme Cañete (2010, p. 1), “auxiliar os biólogos no gerenciamento de dados a respeito de espécimes por eles coletados assim como exemplares já catalogados em coleções de museus”.

Nessa aplicação os criadores utilizaram o Sistema R, para segundo Cañete permitir “a execução de consultas avançadas para análise de dados por parte dos biólogos. As consultas otimizadas são processadas através de aplicações externas [...], que são integradas no sistema TaxonomyBrowser de forma transparente ao usuário” (2010, p. 3)

4. Metodologia

Os profissionais têm dificuldades em encontrar aplicações que permitam que eles interajam com seus dados de uma maneira que possam analisá-los e visualizá-los de forma interativa, por isso a aplicação deve atender e respeitar alguns requisitos e funcionalidades específicas para cumprir o objetivo de facilitar a interação dos profissionais com os seus conjuntos de dados.

Os requisitos que foram levantados com base nas principais necessidades para a realização de análise e visualização de dados, foi que a aplicação deverá permitir o envio de arquivos de dados no formato csv ou txt. Dando opções para o usuário informar as características do arquivo, por exemplo, qual é o separador dos dados (ponto, vírgula, ponto e vírgula), se os dados possuem cabeçalho ou não, entre outras informações pertinentes.

Além disso, é de suma importância para o cumprimento do seu objetivo mostrar os dados do arquivo enviado na tela de forma clara e organizada para que o usuário possa visualizá-los de forma interativa.

O usuário, também deverá ter a opção de gerar diferentes tipos de gráficos a partir dos dados do arquivo enviado, gráficos como boxplot, histograma e plot. Assim como, realizar alguns cálculos básicos de estatística (média aritmética, mediana, moda, quartis), para que se possa realizar uma melhor análise sobre as informações.

A partir dos requisitos, portanto, será desenvolvida a aplicação utilizando a linguagem de programação R e sua IDE (Ambiente de Desenvolvimento Integrado) o RStudio, juntamente com o pacote Shiny para que seja possível disponibilizar a aplicação na Web.

5. Resultados Esperados

A aplicação encontra-se em fase de desenvolvimento, portanto, não é possível ainda apresentar resultados mais objetivos sobre a disponibilização da aplicação para os usuários.

Assim como Radu et al. (2014) apontam raramente é possível ver aplicações, resultados ou soluções para problemas reais utilizando dados abertos para o bem público. Ou seja, ainda é escasso o número de aplicações que propõe a utilização dos dados de forma interativa para usuários que necessitam de um melhor controle sobre as informações.

Com o desenvolvimento da aplicação, espera-se suprir a atual necessidade por aplicações que permitam uma visualização e análise rápida e eficiente dos dados de forma gratuita, já que a grande maioria dos usuários que necessitam deste tipo de serviço, acabam por recorrer a programação, o que se torna um empecilho para quem não possui conhecimento nesta área.

Portanto o resultado esperado é que a aplicação auxilie os usuários que necessitam visualizar e analisar seus dados de forma rápida, fácil e interativa.

6. Referências

- Idreos S; Papaemmanouil O; Chaudhuri S. (2015) “Overview of Data Exploration Techniques”. ACM SIGMOD International Conference on Management of Data, Tutorial. Melbourne, Austrália.
- Beck, Daniel, Dennis, Christopher; FOSTER, James. (2014) “Seed: a user-friendly tool for exploring and visualizing microbial community data”. Department of Biological Sciences, University of Idaho, Moscow, v. 31, n. 4, p. 602-603, Outubro.
- Souza, Emanuel Fernando Maia; Peternelli, Luiz Alexandre; De Mello, Márcio Pupin. (2008) “Software Livre R: aplicação estatística”. Disponível em < <http://goo.gl/xQqK27>> Acesso em: 08 jun. 2015.
- Matloff, Norman. (2011) “The Art of R Programming: A tour of statistical software design.” 1. ed. San Francisco: No Starch Press.
- Mello, Marcio; Peternelli, Luiz. “Conhecendo o R: Uma visão estatística”. (2013) 1. ed. Viçosa: Editora UFV.
- Radu, Marius; Muresan, Ioana; Nistor, Răzvan. (2014) “Using R To Get Value Out Of Public Data”. Disponível em < <http://goo.gl/GQQckw>> Acesso em: 10 jun. 2015.



IFSUL 2015
Passo Fundo - RS
20 a 22 de Outubro de 2015

Anais do
II Simpósio de Informática
Big Data

simpinf.passofundo.ifsul.edu.br