

O Uso de Raciocínio Baseado em Casos com clusterização para Auxiliar na Gestão de Problemas da Base Aérea de Florianópolis

Rafael Coutinho de Sousa¹, Fernando Machado Bernardino¹, Roberto Fabiano Fernandes^{1,2}, Flávio Ceci^{1,2}

¹Departamento de Ciências Tecnológicas e Ciências Exatas. Universidade do Sul de Santa Catarina (UNISUL) – Palhoça – SC – Brasil

²Departamento de Engenharia do Conhecimento – Universidade Federal de Santa Catarina (UFSC) – Florianópolis – SC – Brasil

coutasurf@gmail.com, fernandombernardino@gmail.com,
robertofabiano.fernandes@gmail.com, flavio.ceci@unisul.br

Resumo. *O presente trabalho trata como assunto principal o problema em encontrar informações relevantes existentes na Base Aérea de Florianópolis (BAFL). Tendo em consideração que em uma organização militar o volume de informações cresce gradualmente, um grande problema vem a tona. Como recuperar estas informações após um certo período de tempo? Graças a implementação do RBC (Raciocínio Baseado em Casos), é possível encontrar soluções para novos problemas através de casos iguais ou similares. Juntando esta técnica a outras, como por exemplo, a recuperação de informações, a indexação textual e a clusterização é possível facilitar a recuperação e identificação rápida e fácil dos casos pertinentes a uma possível solução de um novo problema. Também validou-se a solução a partir de um estudo de caso junto a BAFL e percebemos que o resultado foi satisfatório, devido ao volume de informação resultante e sendo condizente com os termos da busca.*

1. Introdução

Em praticamente todos os seguimentos organizacionais a prática de resolução de problemas é comum. Ao longo da jornada diária de trabalho é comum deparar-se com uma série de situações que já foram vividas pelo indivíduo ou por algum outro membro da organização. Percebe-se que muito do conhecimento organizacional está na verdade na cabeça dos seus funcionários (HENRIQUE, 2005). Tendo em vista este cenário, é preocupante para os gestores de uma organização perder algum dos seus membros, pois junto com eles pode ser levada toda a expertise para lidar com uma determinada situação.

Para solucionar um novo problema procura-se primeiramente identificar situações parecidas que levem a possíveis soluções, a área da inteligência artificial criou os chamados Sistemas de Raciocínios Baseados em Casos (RBC). Para Wangenheim (2003), é uma técnica poderosa para solução imediata de problemas, sendo aplicada de forma simples e direta a um amplo quadro de tarefas, tipicamente relacionadas à inteligência artificial.

O presente trabalho tem por objetivo auxiliar a Base Aérea de Florianópolis (BAFL) a encontrar soluções para problemas já conhecidos ou similares e sugerir possíveis

soluções aos problemas enfrentados pelos militares. Entretanto, somente a técnica de RBC para o problema da BAFL, ocasionaria outros problemas na busca da informação requisitada, onde uma grande quantidade seria armazenada e posteriormente estas informações seriam resgatadas, todavia, quanto maior o número de casos, maior será o tempo necessário para realizar a recuperação. Ainda assim, depois de recuperados, é necessário criar uma regra para que não ultrapasse um número excessivo de casos sem similaridade com o termo da busca ou, então, um patamar de similaridade em que serão recuperados apenas os casos cujas medidas ultrapassem esse patamar.

Para isso, a melhor utilização do RBC é feita em conjunto com a técnica de clusterização, ou seja, agrupar os casos em determinados grupos com base na sua similaridade, com isso tendo uma informação mais precisa em um menor tempo de espera. Segundo Jain (1999), a técnica de clusterização é uma das principais etapas de processos de análise de dados, formando agrupamentos ou clusters.

Tem-se como objetivo desenvolver um sistema utilizando o raciocínio baseado em caso combinado com a técnica de clusterização para auxiliar a base aérea de Florianópolis a encontrar soluções para novos problemas através de casos iguais ou similares. Para isso foi desenvolvido um protótipo a fim de atestar a viabilidade do modelo proposto. Para a validação submeteu-se o protótipo e um questionário para os militares da BAFL, onde eles classificavam se o protótipo atende ou não as necessidades previamente levantadas.

2. Referencial Teórico

A presente seção descreve alguns elementos necessários para o entendimento deste artigo. São abordados conceitos básicos relacionados à recuperação de informação (RI), raciocínio baseado em caso (RBC), e Clusterização.

2.1. Recuperação de Informação

De acordo com Ferneda (2003) o autor determina que a recuperação da informação como recuperação de documentos, pois os sistemas não recuperam informações e sim textos e/ou documentos que podem ser utilizados como informação em potencial. Segundo Korfhage (1997), existem muitos modelos de recuperação de informação e que são utilizados especificamente de acordo com os objetivos da busca. Os modelos mais utilizados são: booleano, vetorial, probabilístico e modelos que utilizam linguagem natural.

De acordo com Wives e Loh (1998), se uma expressão aparece em muitos documentos o grau de discriminação será baixo (onde diversos documentos serão resgatados a partir dessa expressão) enquanto que, se uma expressão aparece em alguns documentos então é dito que ela caracteriza bem estes documentos. Obviamente, as expressões que aparecem em todos os documentos não serão analisadas estes são identificados como *stopwords* e, geralmente, são as preposições, artigos, pronomes, etc.

Para a recuperação de informação geralmente são retiradas as palavras *stopwords* durante a indexação, bem como a aplicação de algoritmos de *stemming*. Para Juniore Kaestner (2002), algoritmos de *stemming* é o processo de conversão de cada palavra para o seu radical, excluindo sufixos representados por flexão verbal e plural. Os

algoritmos de *stemming* normalmente incorporaram o conhecimento extensivo da língua, portanto, são dependentes da linguagem.

2.1.3 Indexação

Segundo Rezende (2003), a indexação consiste em realizar uma pesquisa em textos de forma eficiente sem a necessidade de analisar todo o documento, semelhante ao que acontece com o processo de “homônimo” utilizado convencionalmente em bancos de dados.

A técnica de indexação utilizada no presente trabalho é conhecida como técnica de índice invertido. O processo de indexação consiste em criar estruturas de dados associados com a parte textual dos documentos, como por exemplo, as estruturas de tabelas de sufixos e arquivos invertidos. (FRANKES e BAESA-YATES, 1992).

Um índice invertido, segundo Manning, Raghavan e Schütze (1999), é uma estrutura de dados que relaciona cada palavra com todos os documentos que a possui e também armazena a frequência com que a palavra é encontrada no documento. Com a utilização do índice invertido torna-se mais fácil a busca de informação em documentos.

2.2. Raciocínio Baseado em Caso

Wangenheim (2003) afirma que o Raciocínio Baseado em Casos é uma técnica da Inteligência Artificial entusiasmada no modelo de cognição e aspecto humano, que através de um problema já conhecido seja possível encontrar respostas para um novo problema. Segundo Kolodner (1993), as experiências adquiridas são entidades denominadas “casos”, e que devem estar descritas em termos que representem seu conteúdo e contexto. O caso pode ser representado de diferentes formas. O exemplo mais simples é uma experiência descrita por meio de atributos com valores.

2.2.1 O ciclo de raciocínio baseado em caso

Aamodt & Plaza (1994) afirmam que o ciclo de funcionamento de um sistema RBC é composto de quatro processos principais: recuperação, reutilização, revisão e retenção.

2.2.2 Recuperação

O objetivo desse processo é recuperar o caso ou os casos, da “Base de Casos”, que se assemelhem mais ao novo caso, para que a solução seja mais precisa. A recuperação é feita através da utilização dos atributos do novo caso que são relevantes na solução de um problema. Aamodt e Plaza (1994) afirmam que o processo de recuperação de casos inicia-se com uma descrição de um problema e termina somente quando o caso mais semelhante é localizado.

2.2.3 Reutilização

Segundo Aamodt e Plaza (1994), existem dois pontos principais na reutilização da solução do caso recuperado em relação ao novo caso:

- As diferenças entre o caso recuperado e o caso novo;
- Qual parte do caso recuperado poderá ser utilizada no novo caso.

A simples tarefa de classificar as diferenças é abstrata, pois é considerada uma tarefa não relevante, enquanto que a similaridade é considerada relevante e a solução do caso recuperado é transferida para o novo caso como sendo a solução proposta. Esse é o tipo trivial de reutilização, contudo, há outros sistemas que implementam a reutilização adaptando a solução do caso recuperado para o novo caso.

2.2.4 Revisão

Quando uma solução gerada para um caso na fase de reutilização não for correta, aparece uma oportunidade de aprendizado a partir dessa falha, que é conhecida como chamada de revisão que consiste de duas tarefas: “diagnóstico e recuperação da informação” (WANGENHEIM, 2003).

1.1.1.1.1. 2.2.4.1 Avaliação da solução

A tarefa de avaliação considera o resultado de aplicar a solução em um ambiente real, pedindo para um especialista ou aplicando regras que valide uma solução. O resultado da aplicação da solução pode levar algum tempo a aparecer, dependendo do tipo da mesma. Por exemplo, em um sistema de apoio à decisão médica, o sucesso ou fracasso do tratamento pode levar várias horas ou mesmo vários meses. O caso pode ainda ser instruído e validado na base de casos, por um período intermediário, mas devem ser marcados como validados temporariamente. (BEPPLER, 2002).

2.2.5 Retenção

Beppler, (2002) afirma: após ter resolvido o caso e passado pelo processo de revisão, o mesmo poderá ser armazenado. Sendo assim, o sistema de RBC só é eficaz, quando é capaz de aprender a partir de experiências anteriores e da correta indexação dos problemas.

2.3. Clusterização

Segundo Moscato e Von Zuben (2012), clusterização (ou agrupamento) significa a organização dos dados em conjuntos, de acordo com as suas similaridades. Afirmam também que dados pertencentes a certo cluster, deverão ser mais similares entre si do que se comparados a dados de outros clusters.

Seguindo a mesma linha de pesquisa, Ochi, Dias e Soares (2012) definem que cluster, de uma forma geral, nada mais é do que um agrupamento de elementos de uma base de dados, de modo que estes grupos representem uma relação de similaridade maior com qualquer elemento pertencente ao mesmo cluster do que com elementos pertencentes a outros clusters.

Oliveira (2008) afirma que a clusterização dos dados pode facilitar a extração de informações relevantes sobre grandes volumes de dados.

2.3.1 Suffix Tree Clustering (STC)

O algoritmo STC (Suffix Tree Clustering), ou Árvore de Sufixo, agrupa os textos ou documentos de acordo com frases idênticas (ZAMIR, 1999). Segundo Osinski (2003), a

lógica por trás deste algoritmo é que as frases quando comparadas com palavras chave, tem maior capacidade descritiva. Isto resulta da capacidade de manter uma relação de proximidade, ou semelhança, e ordem entre as palavras. Uma das vantagens do uso do STC é que as frases são utilizadas tanto para descobrir, quanto para descrever os clusters.

2.3.2 Algoritmo Lingo

Segundo Osinski (2003), a ideia geral por trás do Lingo é encontrar as descrições significativas dos clusters e então com base nestas descrições, determinar o seu conteúdo. O Lingo é utilizado para resolver problemas de clusterização de resultados de uma pesquisa. Ceci (2010) acredita que a maior vantagem da utilização deste algoritmo é a sua preocupação com os nomes dos *clusters*.

3. Arquitetura Lógica

O esquema lógico do sistema é constituído a partir do que denominamos como os quatro “R” (Recuperar, Reutilizar, Revisar e Reter), que representam as quatro etapas que caracterizam sistemas de raciocínio baseados em casos. A figura 1 ilustra a arquitetura lógica, onde o núcleo do sistema proposto é representado pelo componente Locant:

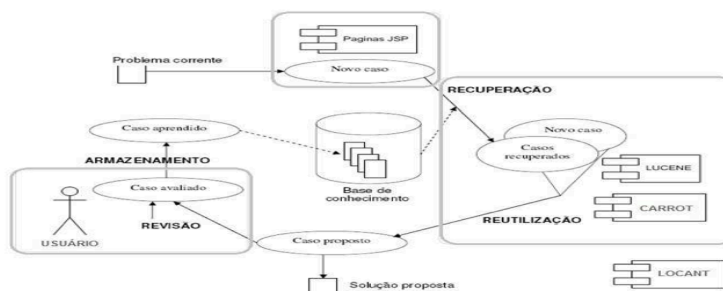


Figura 1 – Arquitetura lógica do protótipo proposto.

Além disso, a figura 1 apresenta como cada módulo do sistema proposto está distribuído no ciclo de RBC padrão, as próximas seções apresentam cada uma das etapas com mais detalhes.

3.1. Recuperar

O processo de indexação do Locant funciona da seguinte maneira, toda vez que uma dúvida for solucionada, realiza-se o processo de indexação.

Além do Lucene², que é um motor de indexação e busca da Apache, que faz a busca nos índices textuais no processo de recuperação, é utilizado o framework Carrot³, que tem o

² <http://lucene.apache.org>

papel de pegar os resultados do Lucene e analisá-los, com o objetivo de clusterizar, ou seja, gerar grupos de palavras relevantes de acordo com o resultado vindo do Lucene, para que se possa chegar a informação desejada com uma maior facilidade e transparência. Sendo assim, este sistema cria grupos específicos a partir dos termos mais relevantes de cada caso, formando uma espécie de filtro dos resultados obtidos.

3.2. Reutilizar

Com a recuperação dos casos a partir da busca feita pelo Lucene nos índices, o mesmo processa os casos retornados em relação com a busca feita pelo usuário e a partir disso é feito uma comparação e tentativa de torná-lo adequado para o uso.

Essa adaptação pode ser feita através de heurísticas ou modelos. Nesse caso é sugerido fazer uma busca em cima dos resultados obtidos, ou seja, em soluções já conhecidas a fim de identificar palavras chaves e a partir delas extrair partes dos casos recuperados. Dessa forma é gerar a solução para o novo caso.

3.3. Revisar

Esta etapa consiste em verificar a solução proposta pela etapa da reutilização (adaptação), é utilizada a manifestação do usuário que é o responsável por determinar se o caso retornado pela aplicação é suficiente para a solução do problema ou não.

É nessa etapa que o usuário contribui de forma direta para o aprendizado do sistema, pois é ele que avalia os casos sugeridos a partir da similaridade do termo da sua busca, com os casos já solucionados.

Quando o usuário encontrar uma ou mais soluções para o seu problema, ele deve selecionar estas soluções para que então o sistema possa neste momento reter essa nova informação e a tornar mais um caso para as próximas buscas, dessa forma servindo como solução para novas dúvidas.

3.4. Reter

Após a etapa de revisão do caso, o Locant armazena o mesmo na base de dados e logo em seguida chama a aplicação do Lucene para indexá-lo junto ao arquivo de índices. Dessa forma o sistema armazena o novo caso e com isso “aprende” a solução para mais um novo problema.

Nesta etapa, ocorre efetivamente o aprendizado do sistema, sendo que após o usuário informar a solução para o problema informado, o sistema se encarrega de armazenar essa informação em forma de caso, guardando arquivo de índices do Lucene para posteriormente ser resgatado nas próximas buscas, configurando-se como uma solução para um caso antigo e uma opção de sugestão para resolver dúvidas e problemas futuros das próximas buscas dos usuários.

Portanto, é no arquivo de índices do Lucene que toda busca realizada atrás de soluções antigas para um novo problema, é pesquisada e não no banco de dados.

³ <http://project.carrot2.org/>

4. Proposta de solução

O esquema físico da solução proposta é demonstrado na Figura 2. Onde o usuário a partir de um terminal, acessa o sistema com o suporte do servidor web “Tomcat” junto com o banco de dados Mysql e com os Frameworks Lucene e Carrot.

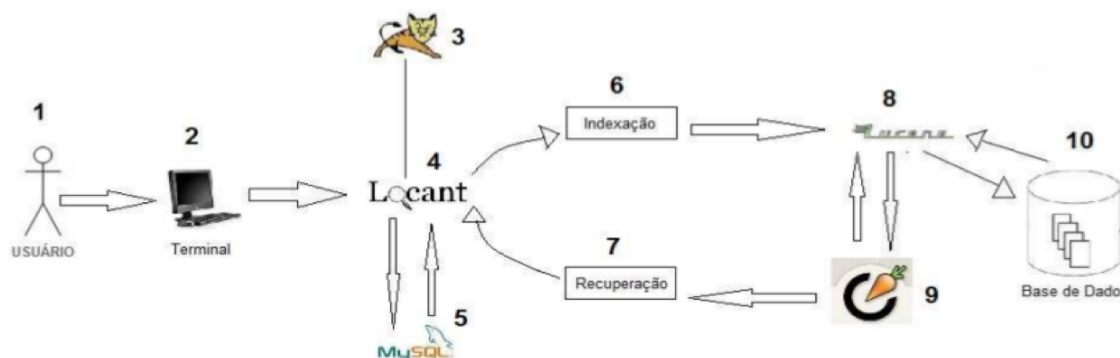


Figura 2 – Arquitetura Física do protótipo proposto.

A figura 2 tem a seguinte descrição:

- ✓ usuário representado pelo número 1;
- ✓ um ponto de acesso representado pelo número 2;
- ✓ um servidor de aplicação WEB “Tomcat” para poder funcionar o sistema representado pelo número 3;
- ✓ aplicação “Locant”, sendo o centro do fluxo representado pelo número 4;
- ✓ banco de dados “MySQL” responsável por armazenar informações do sistema, representado pelo número 5;
- ✓ processo de indexação, representado pelos números 6, 8 e 10;
- ✓ processo de recuperação, representado pelos números 7, 8, 9 e 10.

5. Estudo de caso

Para validar o modelo foi desenvolvido um protótipo com base em um estudo de caso, como cenário do estudo de caso optou-se pelo desenvolvimento de um sistema para auxiliar na manutenção de aeronaves da Base Aérea de Florianópolis, este sistema recebeu o nome Locant. Inicialmente o usuário deve buscar por um determinado problema, caso o mesmo não retorne soluções adequadas ele pode abrir um novo caso, que, ou por aproveitar de partes de um caso anteriormente cadastrado e solucionado. A Figura 3 apresenta como exemplo o resultado de uma busca pelo seguinte problema: “Motor não liga”, onde o pode-se selecionar o caso que se aproxima mais do real problema enfrentado. No exemplo retratado na Figura 3 selecionou-se o seguinte caso: “Motor não liga”.

Gerenciar Usuário Gerenciar Dívida Olá, Rafael Coutinho Sair

Loquant Motor não liga [Buscar] Nova Dívida

Casos Semelhantes com: Motor não liga

Todos

Elevado (4)

Rotação (4)

Pressão de Óleo (3)

Atinge ou Demora a Atingir a Rotação (2)

Baixa (2)

Baixo (2)

Figura 3-7 (2)

Fluxo de Combustível (2)

Motor não Acelera (2)

Motor não Acende (2)

Motor não desacelera : CAUSA PROVÁVEL : Válvula "by-pass" do FCU travada na posição fechada. INVESTIGAÇÃO : CORREÇÃO.

Motor não acelera normalmente : CAUSA PROVÁVEL : Filtro de P3 contaminado (quando instalado) INVESTIGAÇÃO : CORREÇÃO. Substitua

O motor não acende : CAUSA PROVÁVEL : Vela de ignição ou cabos defeituosos (veja a figura 7-4) INVESTIGAÇÃO : Verifique

Motor não acende durante a partida : CAUSA PROVÁVEL : Bomba de combustível do motor defeituosa. INVESTIGAÇÃO : Afrouxe a linha entre a

Motor nao gira durante a partida (Ng zero) : Causa Provável : Motor não gira livremente. Investigaçao : Remova o tacogerador de Ng e tente gir

O motor apresenta uma partida atrasada ou quente : CAUSA PROVÁVEL : Vela de ignição ou cabos defeituosos (veja a figura 7-4) INVESTIGAÇÃO : Encarete

motor não acelera na hora de levantar vôo : CAUSA PROVÁVEL : Sujeria nas seções pneumáticas. INVESTIGAÇÃO : CORREÇÃO. Substitua o FCU

Motor não atinge ou demora a atingir a rotação de marcha lenta (Ng) : CAUSA PROVÁVEL : Corrosão ou gelo na seção pneumática do FCU. CORREÇÃO: Substitua o FCU.

Motor limitado em temperatura (o TIT máximo é atingido antes de se atingir o torque de decolagem) : CAUSA PROVÁVEL : Leitura dos termômetros baixas INVESTIGAÇÃO : Verifique a instrumentação. CO

Resultado Selecionado

Figura 3 - Tela com resultados ao problema consultado

Com os casos expostos na tela, o usuário conseguiu visualizar diversas opções que possivelmente poderiam atender ao seu propósito. Ao selecionar uma destas opções, foram expostas detalhadamente todas as alternativas que serviram para solucionar o respectivo problema, conforme a Figura 4.

Gerenciar Usuário Gerenciar Dívida Olá, Rafael Coutinho Sair

Loquant

Termo pesquisado : motor não liga

Titulo do Caso :	Motor nao gira durante a partida (Ng zero)	
Solução Autor: Rafael Coutinho Data: 07/04/2012	Causa Provável : Arranque/gerador alimentado com voltagem insuficiente. Investigaçao : Verifique bateria(ou fonte externa), cablagens a conexões. Correção : Corrija	<input type="checkbox"/>
Solução Autor: Rafael Coutinho Data: 07/04/2012	Causa Provável : Arranque-gerador defeituoso. Investigaçao : Se forem ouvidos ruídos de funcionamento do arranque durante a tentativa de partida, mas no ocorrer rotação do gerador de gases, o eixo está cisalhado. Correção : Substitua o arranque-gerador.	<input checked="" type="checkbox"/>
Solução Autor: Rafael Coutinho Data: 07/04/2012	Causa Provável : Motor não gira livremente. Investigaçao : Remova o tacogerador de Ng a tente girar o compressor através do eixo de acionamento deste. Remova a seção de potência (veja o "PT6A-27 and PT6A-28 Engines Maintenance Manual") a verifique as folgas das pontas das palhetas da turbina do compressor. Correção : Se as palhetas estiverem roçando nas camisas ("shrouds"), remova a turbina do compressor e esmerilhe-as (veja o "PT6A-27 and PT6A-28 Engines Maintenance Manual"). Se as folgas das pontas das palhetas estiverem corretas, erve o motor a uma oficina autorizada.	<input checked="" type="checkbox"/>

Resoluiu Resoluiu Parcialmente Não Resoluiu

Figura 4 – Tela de detalhamento do caso.

Com isso, o usuário analisou as soluções que resolveram o caso exposto, e selecionou aquela que também veio a resolver o seu problema atual. Após o usuário ter escolhido as soluções do caso o qual também resolveu o seu problema, o sistema armazenou estas informações, ocorrendo assim um novo aprendizado do sistema, fazendo com que esse problema atual que o usuário acabou de solucionar, transformasse-se em um novo caso que possa servir de solução para futuros problemas.

A validação foi efetivada através de um questionário referente ao sistema e à sua proposta de solução. Primeiramente foi realizada uma breve explanação do sistema, seus objetivos e suas funcionalidades, sendo o entrevistado posteriormente liberado para testar o sistema. Na sequência, foi apresentado ao usuário um questionário com dez perguntas, que tem como objetivo validar a proposta de solução. Para validar o sistema, foram utilizadas perguntas que abordaram desde a percepção dos usuários sobre os seguintes pontos: resultados relevantes e pertinentes as buscas, forma dinâmica da apresentação dos resultados, desempenho, interface, satisfação com a solução

apresentada. Compõe o questionário quatro alternativas de respostas: atende completamente, atende, atende em partes, não atende.

Quanto a amostra ela foi composta por vinte militares do sexo masculino, com idades entre 20 e 44 anos, da área de tecnologia da informação e mecânica do avião. O resultado obtido foi o seguinte:

Tabela 1. Resultados percentuais do questionário

QUESTÃO	ATENDE	ATENDE EM PARTES	ATENDE COMPLETAMENTE	NÃO ATENDE
1	50%	25%	25%	0%
2	0%	0%	100%	0%
3	15%	10%	75%	0%
4	40%	15%	45%	0%
5	60%	20%	20%	0%
6	30%	10%	60%	0%
7	30%	10%	60%	0%
8	50%	0%	50%	0%
9	10%	60%	5%	25%
10	40%	0%	60%	0%

Pode-se perceber pelos resultados da validação que a soluções apresenta resultados positivos o que possibilita evoluções para um futuro produto.

6. Conclusão

Este trabalho tratou como assunto principal o problema em encontrar informações relevantes às dificuldades atuais existentes na Base Aérea de Florianópolis (BAFL) tendo em consideração que cada vez mais as organizações tendem a documentar suas tarefas e conhecimentos, criando um grande número de informações que, em algum momento, poderá ser preciso recuperá-las.

Neste contexto, foi desenvolvida uma proposta de solução onde se consegue gravar essas informações de modo organizado, para que em uma futura busca por determinado assunto, venha a ser facilmente encontrada a solução.

Existem diversas técnicas para realizar buscas por informações, mas a técnica utilizada no presente trabalho foi a de Raciocínio Baseado em Casos (RBC). Com este propósito, levou-se em conta o fato de que a simples recuperação de informação não agregaria maiores conhecimentos ao sistema, apenas recuperaria as informações armazenadas e não criaria um histórico de buscas ou nem mesmo um conhecimento para cada termo pesquisado com as informações encontradas.

Já o RBC, uma técnica da Inteligência Artificial (IA), resolve esse problema e é responsável pelo papel da recuperação da informação, além de que, cria um aprendizado para cada termo de busca utilizado, como as soluções encontradas pelo o usuário. Ou seja, para todo termo pesquisado, quando informado a solução encontrada, cria-se uma espécie de histórico de aprendizado do sistema, onde o RBC não se prende a apenas uma simples recuperação da informação, ele efetivamente aprende com cada busca feita pelo usuário.

Através da aplicação de um questionário para validar a proposta de solução deste trabalho, foi constatado a sua eficácia para o problema da recuperação da informação de forma organizada, com qualidade e similaridade com os termos da busca.

Pode-se concluir que a técnica de RBC juntamente com a indexação utilizando o Lucene e, clusterização utilizando o Carrot é extremamente eficiente para os cenários que precisam armazenar informações e depois recuperá-las com eficácia, não se limitando a apenas aos contextos da BAFL.

Referências

- Aamodt, a., Plaza, E. Case-based reasoning: foundational issues, methodological variations and system approaches, In: Artificial Intelligence Communications, 1994, Vol. 7
- Beppler, F. D. Emprego de RBC para Recuperação Inteligente de Informação, 2002. 100 p. Dissertação de Mestrado. Programa de Pós-Graduação em Engenharia de Produção. Universidade Federal de Santa Catarina, Florianópolis 2002.
- Ceci, F. Um modelo semiautomático para a construção e manutenção de ontologias a partir de bases de documentos não estruturados. 2010. 129 p. Dissertação de Mestrado. Engenharia do Conhecimento. Universidade Federal de Santa Catarina. 2010.
- Ferneda, E. Recuperação de informação: análise sobre a contribuição da Ciência da Computação para a Ciência da Informação. 2003. Tese (Doutorado em Ciências da Comunicação) – Universidade de São Paulo, São Paulo, 2003.
- Frakes, W. B., Baesa-Yates, R. Information Retrieval Data Structures & Algorithms, Prentice Hall, 1992.
- Henrique, L. C. J., Barbosa, R. R. Gestão da informação e do conhecimento organizacionais: em busca de uma heurística adaptada à cultura brasileira, *Perspect. ciênc. inf.*, Belo Horizonte, v.10 n.1, p. 4-17, jan./jun. 2005.
- Jain, A.K. et. al.. Data Clustering: A Review, In: ACM Computing Surveys, vol. 31, no. 3, p.. 264-323, 1999.
- Junior, Carlos N. Silla ; Kaestner, Celso A. A.. Estudo de Métodos Automáticos para Sumarização de Textos. Disponível em: <<http://www.ppgia.pucpr.br/~silla/publications/2002-STD-SillaKaestner.pdf>>. Acesso em: 12 set. 2011
- Kolodner, J. L. Case-Based Reasoning. Morgan Kaufmann, San Francisco, California, 1993.
- Korfhage, Robert R. Information Storage and Retrieval. New York: John Wiley & Sons, Inc., 1997.
- Manning, Christopher D.; Raghavan, Prabhakar; Schütze, Hinrich. Introduction to information retrieval. New York: Cambridge University Press, 2008.

- Oliveira, T. B. S. Clusterização de dados utilizando técnicas de redes complexas e computação bioinspirada, Instituto de Ciências Matemáticas e de Computação - USP. 2008.
- Osinski, S. An Algorithm for Clustering of Web Search Results. Poznan University of Technology, Poland, June 2003.
- Rezende, S. O. Sistemas inteligentes: fundamentos e aplicações. São Paulo: Manole, 2003.
- Wangenheim, C. G. V. Raciocínio Baseado em Casos / Christiane Gresse Von Wangenheim, Aldo Von Wangenheim. São Paulo: Manole, 2003.
- Wives, L. K. Stanley. Recuperação de Informações usando a Expansão Semântica e a Lógica Difusa. In: Congreso Internacional En Ingenieria Informatica, ICIE. Universidad de Buenos Aires, Buenos Aires, 1998.
- Zamir, O. E. Clustering Web Documents: A Phrase-Based Method for Grouping Search Engine Results. Doctoral Dissertation, University of Washington, 1999.