

# Ferramenta para Ensino da Técnica de Raciocínio Baseado em Casos

Anita Maria da Rocha Fernandes  
anita.fernandes@univali.br  
UNIVALI

Helton Machado Kraus  
heltonkraus@univali.br  
UNIVALI

**Resumo:** O Raciocínio Baseado em Casos (RBC) estabeleceu-se nos últimos anos como uma das tecnologias mais populares para o desenvolvimento de Sistemas Baseados em Conhecimento. Para o desenvolvimento deste tipo de sistema existem ferramentas que fornecem funções para elaboração e testes, porém obrigam os usuários a terem pleno domínio da técnica de RBC. O presente artigo apresenta uma ferramenta para o desenvolvimento de sistemas de RBC, que utiliza como base o framework jColibri 2, onde os usuários podem entender o funcionamento da técnica de RBC e elaborar novos sistemas interagindo com a ferramenta.

**Palavras Chave:** RBC - IA - Ensino de computação - -



## **1. INTRODUÇÃO**

A Inteligência Artificial (IA) é uma área de pesquisa da Ciência da Computação que estuda a representação de conhecimento e a capacidade das máquinas de pensar e aprender a resolver problemas de maneira similar a mente humana (Fernandes, 2003). A IA tem como objetivo modelar o conhecimento humano através de esquemas computacionais e neste contexto existem diferentes técnicas sendo empregadas, tais como a técnica de Raciocínio Baseado em Casos (RBC). Sistemas de RBC permitem a extração, organização e reuso do conhecimento utilizado para tomada de decisões no passado, tornando explícitos os métodos utilizados e permitindo o seu aperfeiçoamento (AAMODT & PLAZA, 1994).

Raciocínio Baseado em Casos, ou Case Based Reasoning (CBR), é uma técnica que busca a solução para uma situação atual através da recuperação e adaptação de soluções passadas semelhantes, dentro de um mesmo domínio do problema. O sistema é capaz de localizar e encontrar partes de casos que não se adequem ao problema, criando um novo caso para uso posterior (FERNANDES, 2003).

Elementos importantes que devem ser considerados durante o desenvolvimento dos sistemas de RBC são o armazenamento e o processamento necessário para recuperação dos casos na base de casos. Com isso desenvolvedores deste tipo de sistema necessitam analisar as diferentes representações de conhecimento e métricas de similaridade para definir qual a mais adequada para cada tipo de domínio de aplicação. A má definição destes dois elementos pode trazer problemas na eficiência dos sistemas de RBC.

Para a criação e aplicação deste tipo de sistema, os desenvolvedores devem optar por criar sua própria ferramenta, programando toda a lógica do sistema, ou utilizar uma das soluções de desenvolvimento existentes.

As soluções existentes fornecem funções para elaboração de sistemas, porém os usuários precisam ter pleno domínio da técnica de RBC, conhecimento pelos quais alunos de cursos de graduação ou programadores em geral possuem de forma limitada, por não aplicarem a técnica em situações reais. Outro agravante é a falta de uma interface amigável para definição da representação do conhecimento e das métricas de similaridade.

Observando a complexidade envolvida no desenvolvimento dos sistemas de RBC e a falta de soluções apropriadas para o ensino e desenvolvimento, o presente artigo descreve brevemente o estado da arte dos sistemas de RBC e apresenta uma nova ferramenta para elaboração de sistemas de RBC.

Segundo Jaczynski e Trousse (1998) as shells permitem a um usuário, em geral não programador, a geração de aplicações de RBC por meio de uma interface gráfica sofisticada. Nesta interface, os parâmetros necessários ao desenvolvimento da aplicação podem ser definidos de modo interativo pelo usuário, como por exemplo especificar os descritores dos casos relativos ao domínio de conhecimento, bem como o vetor de pesos usados para o cálculo da medida de similaridade usada para recuperar casos.

A ferramenta proposta possibilita desenvolvedores com pouca experiência na área de Inteligência Artificial desenvolver um sistema de RBC com facilidade. Desta forma na próxima seção são apresentadas as principais características dos sistemas de RBC e as soluções de desenvolvimento existentes. Na sequência é apresentada a arquitetura e as principais funcionalidades da ferramenta, os resultados e as conclusões do trabalho.



## 2. RACIOCÍNIO BASEADO EM CASOS

RBC é uma técnica de Inteligência Artificial que busca a solução para uma situação atual através da recuperação e adaptação de soluções passadas semelhantes, dentro de um mesmo domínio do problema (Fernandes, 2003). A recuperação dos dados é realizada em memória, verificando se existem casos semelhantes com as características atuais do problema, podendo encontrar um ou mais casos e adaptá-los de alguma maneira, para que se ajuste ao problema atual, criando um novo caso para uso futuro.

Wangenheim e Wangenheim (2003) afirmam que: “RBC é um enfoque para a solução de problemas e para o aprendizado baseado em experiência passada. RBC resolve problemas ao recuperar e adaptar experiências passadas – chamadas casos – armazenadas em uma base de casos. Um novo problema é resolvido com base na adaptação de soluções de problemas similares já conhecidas.”

Para o desenvolvimento de sistemas RBC é necessário inicialmente utilizar soluções já aplicadas anteriormente com sucesso e determinar qual das experiências mais se assemelha ao problema atual. Para isso se torna necessário que estas experiências passadas sejam analisadas e armazenadas de maneira organizada. O sistema RBC é responsável por realizar a pesquisa nestas experiências armazenadas e verificar se existem casos semelhantes com as características do problema atual, podendo encontrar um ou mais casos, que serão utilizados para chegar a solução do problema. O sistema é capaz de localizar e encontrar partes de casos que não se adequem ao problema, criando um novo caso para uso posterior (FERNANDES, 2003).

Desta forma, um sistema RBC pode ser dividido em quatro elementos básicos (WANGENHEIM & WANGENHEIM, 2003):

- **Representação do Conhecimento:** em um sistema de RBC, o conhecimento é representado principalmente em forma de casos, que descrevem experiências concretas.
- **Medida de Similaridade:** define como será calculada a similaridade entre a situação atual e um determinado caso a base de casos, sendo aplicada repetidamente, par a par, para todos os casos, chegando a um valor de similaridade. Estes valores e casos são ordenados e os mais similares são sugeridos como solução potencial para o problema presente.
- **Adaptação:** situações passadas representadas como casos dificilmente serão idênticas às do problema atual. Sistemas de RBC avançados têm mecanismos e conhecimento para adaptar os casos recuperados completamente, para verificar se satisfazem às características da situação presente.
- **Aprendizado:** para que um sistema se mantenha atualizado e evolua continuamente, sempre que ele resolver um problema com sucesso deverá ser capaz de lembrar-se dessa situação no futuro como mais um novo caso.

Um caso é a forma de conhecimento contextualizado representando uma experiência que ensina uma lição útil. As lições úteis podem ser definidas como aquelas que têm o potencial para ajudar o raciocinador a alcançar uma meta ou um conjunto de metas ou advertem sobre a possibilidade de uma falha ou apontam para um problema futuro (FERNANDES, 2003).

Para que estes casos estejam a disposição para serem reutilizados, estes são organizados e armazenados em uma base de casos, formando um conjunto que geralmente contém experiências positivas descrevendo estratégias de solução que contribuíram com o



sucesso para resolver o problema descrito, de forma que possam ser reutilizadas. Experiências negativas, expressando tentativas frustradas de solução também podem ser armazenadas, com o objetivo de indicar problemas potenciais e prevenir a repetição de erros passados (WANGENHEIM & WANGENHEIM, 2003).

### **3. FERRAMENTAS DE DESENVOLVIMENTO DE RBC**

A literatura revela diferentes tipos de ferramentas voltadas ao desenvolvimento de aplicações de RBC, sendo que algumas já foram descontinuadas. As principais que estão disponíveis para uso atualmente são:

- CBRShell (CBR SHELL, 2009): Desenvolvido pelo grupo Artificial Intelligence Applications Institute School of Informatics da Universidade de Edinburgh. Possibilita realizar as definições através de uma interface gráfica, porém não apresenta uma seqüência lógica de passos para uso. Outra limitação é o uso de apenas arquivos texto para armazenar as informações do caso;
- MyCBR (MyCBR, 2009): Desenvolvida pelo Centro de Pesquisa Alemã de Inteligência Artificial (DFKI). Seu objetivo é minimizar o esforço para construir aplicações RBC que requerem alta intensidade cognitiva nas medidas de similaridade, tendo como principal deficiência constatada o uso apenas de arquivos XML para armazenar informações.
- JColibri (GAIA, 2009): é um framework orientado a objetos que facilita a construção de sistemas RBC desenvolvido pelo GAIA (Group for Artificial Intelligence Applications). Sua primeira versão disponibiliza um ambiente não muito usual para elaboração de sistema e sua segunda versão não possui interface gráfica para elaboração e testes do sistema RBC.

As soluções para desenvolvimento de RBC fornecem funções para elaboração de sistemas, onde analisando suas principais características e funcionalidades, é possível identificar problemas de usabilidade. Estas características e limitações serviram como base para estruturar a ferramenta aqui apresentada.

### **4. ARQUITETURA DA FERRAMENTA DE DESENVOLVIMENTO**

A ferramenta desenvolvida é um ambiente onde usuários podem entender o funcionamento da técnica de RBC, através dos tutoriais e também elaborar novos sistemas de RBC interagindo com o ambiente.

O diferencial da solução proposta em relação às existentes está na possibilidade de realizar as definições do sistema com ajudas contextualizadas e também visualizar cada etapa do ciclo RBC de forma interativa, além de possibilitar exportar a aplicação desenvolvida para ser utilizada em outros aplicativos ou desenvolver uma interface personalizada para o usuário.

O material de auxílio ao desenvolvedor é capaz de ajudar nas definições do sistema RBC, visando indicar os processos mais adequados para o problema que se pretende desenvolver, qual métrica de similaridade pode ser mais apropriada para cada tipo de informação, a forma com que as informações são armazenadas na base de casos e também os métodos de adaptação para novos casos. A Figura 1 apresenta a estrutura da ferramenta desenvolvida.

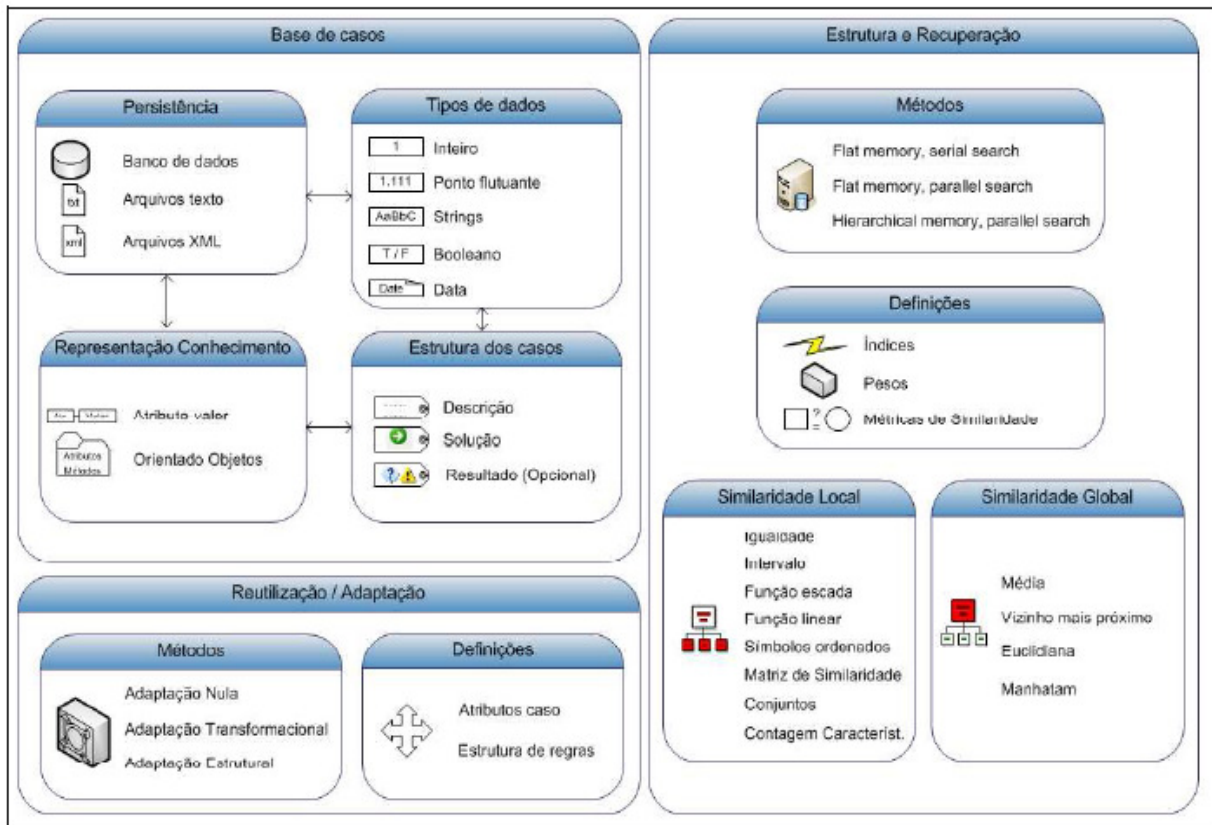


Figura 1: Arquitetura da Ferramenta Desenvolvida

#### 4.1. METODOLOGIA DE DESENVOLVIMENTO

Para o desenvolvimento da ferramenta foram observadas as soluções existentes buscando levantar suas características, onde contou-se que o framework jColibri 2 é portátil e robusto, porém não possui uma interface para elaboração dos sistemas de RBC e disponibiliza poucas métricas de similaridade. Desta forma optou-se por utilizar a estrutura deste framework que é desenvolvido com a linguagem de programação Java e é distribuído como software livre.

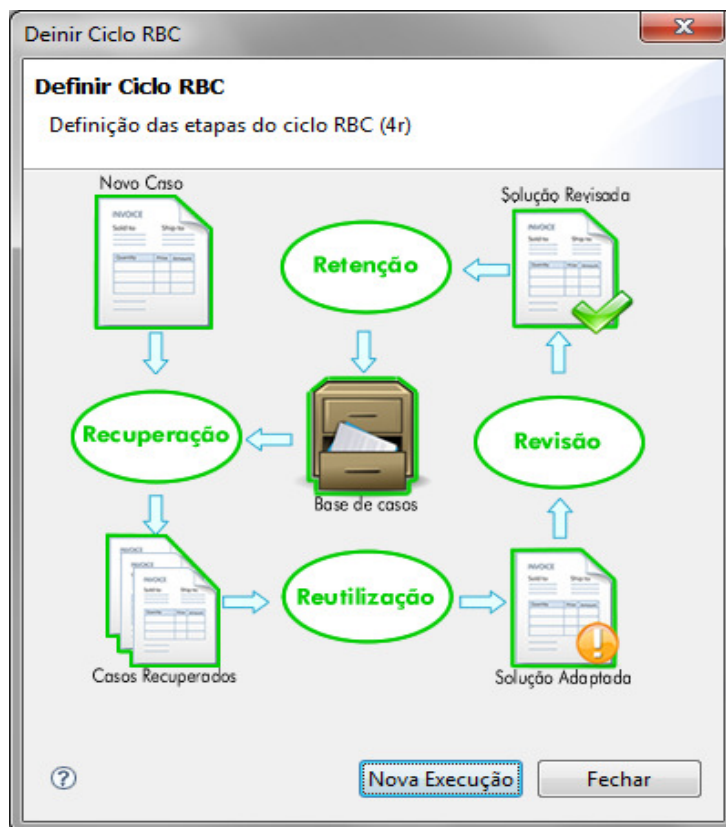
O desenvolvimento da ferramenta iniciou com o levantamento dos requisitos pertinentes a elaboração de sistemas RBC, onde foi possível identificar as definições necessárias para criação deste tipo de sistema, sendo identificadas as seguintes definições:

- Fonte de dados: definir qual a estrutura utilizada para armazenar as informações da base de casos, possibilitando o uso de arquivos texto ou bancos de dados relacional. Na realização dos exercícios esta etapa é abstraída do aluno, onde a ferramenta automaticamente define um banco de dados embarcado, para facilitar a realização do exercício.
- Estrutura do Caso: definir a estrutura que irá compor o caso, definindo através de atributos (nome e tipo de informação que irá armazenar) a descrição do problema, a descrição da solução e ao resultado.
- Base de Casos: definir o método como os casos são recuperados da base de casos e também realizar um filtro inicial nos dados, caso seja necessário.
- Métricas de Similaridade: nesta definição tem-se duas opções: similaridade local (determina como será realizado o cálculo da similaridade entre cada atributo do



caso) e similaridade global (determina como serão computados os valores de similaridade de todos os atributos do caso atual com os casos da base, além de definir os limiares de apresentação ao usuário). A ferramenta disponibiliza para cada tipo de similaridade uma série de métricas que podem ser selecionadas de acordo com o tipo de atributo do caso.

- Adaptação: definir em quais campos haverá adaptação (ajuste) das informações ao realizar o ciclo RBC, definindo os atributos e qual a forma que será utilizada.
- Execução do Ciclo RBC: A Figura 2 apresenta a tela de execução do ciclo RBC, onde o usuário pode visualizar e entender os resultados de cada etapa do ciclo.



**Figura 2:** Execução do Ciclo do RBC

O modelo de funcionamento de um sistema RBC mais aceito é o Ciclo de RBC, que engloba um ciclo de raciocínio contínuo composto por quatro tarefas (AAMODT & PLAZA, 1994), sendo que na tela de execução da ferramenta, o usuário pode visualizar estas tarefas (Recuperação, Reutilização, Revisão e Retenção). Clicando sobre as tarefas (elipses), na seqüência, estas são executadas e os demais ícones indicados após as setas permitem visualizar os resultados.

Para disponibilizar as métricas de similaridade na ferramenta foi realizada uma pesquisa buscando identificar as principais métricas (locais e globais) utilizadas em sistemas de RBC. As métricas de similaridade global disponibilizadas foram: a) Vizinho Mais Próximo Ponderado; b) Distância Euclidiana; c) Distância de Manhattan; d) Distância de ChebyChev; e) Distância de Casamento Simples. No sistema é possível definir uma métrica de similaridade global por projeto e parâmetros como número máximo de casos apresentados após o cálculo da similaridade e o percentual mínimo de similaridade.

As métricas de similaridade local disponibilizadas foram: a) Igualdade; b) Intervalo; c) Degrau; d) Função (Linear, Quadrática ou Normal); e) Matriz de Similaridade; f) Categorias

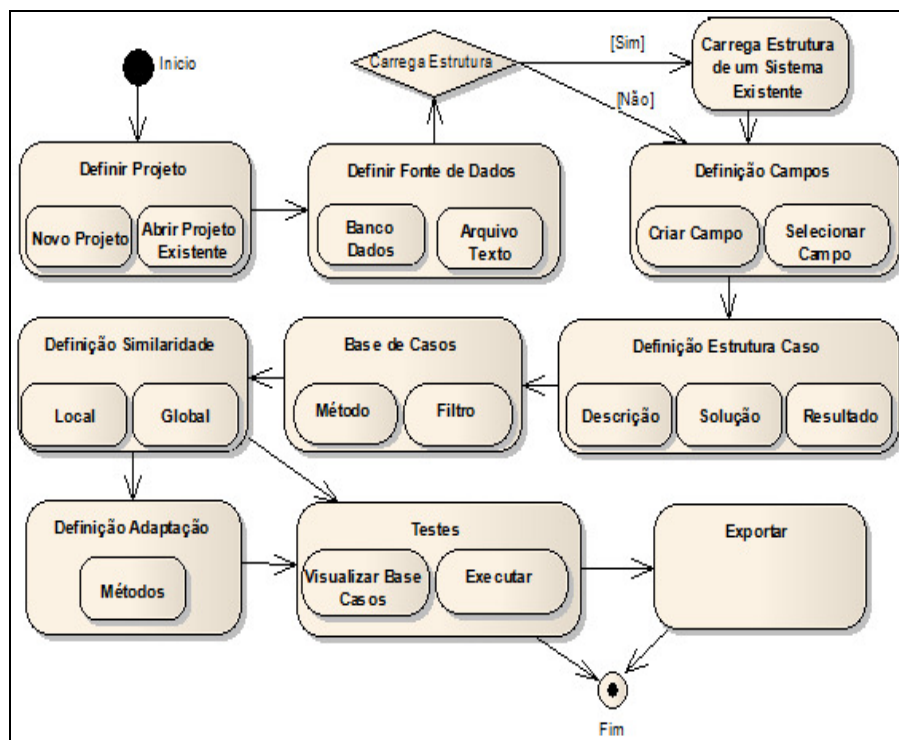


Ordenadas; g) Maior Sequência de Caracteres; e g) Textual. Para a métrica Textual foi utilizada a biblioteca extensível de métricas de similaridade SimMetrics (SIMMETRICS, 2009), que é disponibilizada como software livre. Da biblioteca foram incluídos os algoritmos (definido como parâmetro desta métrica): a) BlockDistance; b) CosineSimilarity; c) EuclideanDistance; d) JaccardSimilarity; e) Jaro; f) QGramsDistance; e g) NeedlemanWunch.

As métricas de similaridade local podem ser definidas para cada atributo do caso, além de possibilitar a definição dos pesos, que buscam destacar atributos com maior significância dentro do caso. Também pode-se definir os parâmetros específicos de cada métrica como por exemplo: valor do intervalo, tipo de função e valor das incógnitas, valores da matriz de similaridade e os respectivos pesos, entre outros.

Como métodos de adaptação foram utilizados apenas os gerados pelo framework jColibri2, sendo disponibilizados: a) Cópia do Atributo, b) Proporção Numérica Direta, e c) Adaptação Nula.

Para exemplificar o funcionamento da ferramenta foi criado um Diagrama de Atividades (Figura 3) sendo possível visualizar a ordem das etapas que devem ser realizadas para desenvolver um sistema de RBC, onde as circunferências indicam o início e término da atividade, as setas indicam o fluxo das atividades e as caixas retangulares indicam as atividades (internamente as opções disponíveis em cada atividade).



**Figura 3:** Diagrama de Atividades

Ao iniciar a elaboração de um sistema RBC, o desenvolvedor deve criar um novo projeto que é armazenado no formato XML (eXtensible Markup Language). Após a definição pode-se realizar as definições através da interface da Shell e conclui-se com os testes do sistema, executando o sistema através da tela de execução do ciclo RBC. Após a etapa de testes é possível exportar a aplicação, onde são geradas as classes Java com as definições realizadas, possibilitando executar sem o uso da ferramenta através do console Java ou ainda incluir importar as definições em outras aplicações.



#### 4.2. CARACTERÍSTICAS DA FERRAMENTA

A ferramenta visa disponibilizar um ambiente onde os usuários possam entender o funcionamento da técnica de RBC, através dos tutoriais e de um personagem que está disponível prestando informações sobre como realizar as definições, e também elaborar novos sistemas de RBC interagindo com o ambiente.

A interface com o usuário segue padrões de ergonomia e usabilidade de forma intuitiva, informando o usuário de todas as possibilidades que o ambiente dispõe e os próximos passos que devem ser realizados.

Na ferramenta o usuário pode recuperar as estruturas de representação do conhecimento já utilizadas em outras aplicações ou criar as próprias estruturas, como tabelas de bancos de dados relacional. Todas as configurações ou definições do sistema RBC podem ser realizadas através da interface gráfica, possibilitando consultar o tutorial, as ajudas contextualizadas e as mensagens apresentadas pelo personagem.

Os testes dos sistemas de RBC desenvolvidos são realizados dentro da própria ferramenta, onde se pode visualizar o ciclo RBC e inspecionar a execução de cada etapa do ciclo. Também é possível exportar o sistema desenvolvido, permitindo a execução em outros ambientes e possibilitando a geração de uma interface personalizada.

Para adequar a ferramenta à ambientes acadêmicos (ensino da técnica de RBC), para alunos de cursos de computação ou afins, foi disponibilizado à professores uma área específica para elaborarem exercícios que poderão ser resolvidos na própria ferramenta. Para realizar o exercício, o desenvolvedor (aluno) pode obter informações através da ajuda da ferramenta, para obter o embasamento teórico da técnica de RBC. Após realizar todas as definições e testar o sistema desenvolvido pode-se gerar um relatório, onde são apresentadas todas as definições e execuções realizadas para ser avaliado pelo professor.

### 5. RESULTADOS

Para avaliar o desempenho no ensino da técnica, foi disponibilizada a alunos do curso de graduação em Ciência da Computação, junto à disciplina de Inteligência Artificial, numa turma com 12 alunos.

Como definição do cenário de testes, foi solicitado pelo professor da disciplina que os alunos elaborassem problemas que pudessem ser resolvidos com o uso da técnica de RBC. Estes alunos não haviam desenvolvido nenhum sistema de RBC em atividades anteriores, apenas tinham o embasamento teórico básico sobre a técnica.

Assim, a descrição dos problemas elaborados pelos próprios alunos foi revisada pelo professor que selecionou as mais pertinentes para que fossem desenvolvidas pelos alunos. Desta forma, cada problema foi desenvolvido por 2 alunos. Os tipos de problemas levantados foram soluções para help desk, auxílio na identificação de pragas no cultivo da soja, diagnóstico de problemas em veículos, sistema de recomendação para compra de jogos, sistema de apoio à decisão em jurisprudência, sistema de recomendação para compra de computadores, e sistema de auxílio à escolha da profissão.

A ferramenta foi disponibilizada aos alunos, onde inicialmente foram apresentadas as principais funcionalidades disponíveis. A realização do exercício não foi supervisionada e os alunos ao fim responderam à um questionário de avaliação, que serviu como base para melhorias.

De maneira geral a avaliação conseguiu identificar os pontos de melhoria, sendo todos ajustados. Após esta avaliação, com auxílio do professor, todos os alunos apresentaram os sistemas de RBC desenvolvidos, onde do total de 12 alunos, apenas 02 precisam de auxílio





para concluir a atividade, por problemas de não conseguirem identificar qual a fonte de dados deveriam utilizar, sendo este problema resolvido com ajustes nas ajudas da ferramenta.

Outra avaliação realizada foi para verificar se os resultados das aplicações desenvolvidas estavam coerentes, quanto à definição dos parâmetros e métricas de similaridade. Desta forma foi desenvolvido um sistema de RBC proposto por (WANGENHEIM & WANGENHEIM 2003), um exemplo de aplicação de busca inteligente de catálogos para uma agência de viagens online.

Nesta aplicação os casos foram compostos por oito atributos (destino da viagem, data da partida, duração da viagem, meio de transporte, categoria da acomodação, preço, serviços adicionais e nome da companhia de viagem), sendo os seis primeiros atributos considerados discriminantes.

As métricas de similaridade (local e global) e pesos foram os mesmos apresentados no exemplo, sendo utilizada como métrica de similaridade global a do vizinho mais próximo ponderado por pesos.

Depois de realizadas as definições foram adicionados manualmente à base de casos seis casos, conforme citados no exemplo, para que fosse possível realizar uma comparação entre a aplicação desenvolvida e o exemplo. Foram realizados testes idênticos aos citados no exemplo, onde se obteve os mesmos valores de similaridade e casos recuperados da base.

Com este mesmo exemplo, após a comparação com o exemplo, o especialista (professor) realizou alterações nas métricas de similaridade e também aplicou métodos de adaptação, onde foi possível verificar o funcionamento do sistema. Os resultados obtidos com os testes foram adequados e coerentes, de acordo com as métricas de similaridade e métodos de adaptação utilizados.

Desta forma, a Shell mostrou-se adequada para o desenvolvimento, tornando-se disponível para professores, alunos e desenvolvedores de sistemas que desejam aplicar a técnica de RBC.

## **5. CONCLUSÕES**

Os avanços na área de Inteligência Artificial são significativos, onde métodos e técnicas buscam simular computacionalmente a capacidade de raciocinar dos seres humanos. A técnica de Raciocínio Baseada em Casos é uma destas técnicas, que pode ser aplicada em várias áreas de conhecimento, pois visa encontrar uma solução para o problema atual comparando-o com casos anteriormente armazenados. O RBC constitui um paradigma de aprendizado de máquina, pois é capaz de simular o raciocínio de um especialista, utilizando a experiência de casos passados na análise e resolução de uma situação atual.

A proposta deste trabalho foi facilitar o desenvolvimento deste tipo de sistema, disponibilizando a alunos de cursos de graduação e pós-graduação, a possibilidade de compreender o funcionamento da técnica e as definições envolvidas através da elaboração interativa de sistemas de RBC. Além de aplicações acadêmicas, também é possível realizar as definições na ferramenta e exportá-las para serem utilizadas em outras aplicações ou gerar interfaces personalizadas.

Na análise, o *framework* jColibri 2 mostrou-se como uma solução viável para ser utilizada como base para a ferramenta proposta, pois conforme descrito pelos desenvolvedores, não possuía uma interface gráfica (ferramenta de autoria) para definição das aplicações desenvolvidas e o conjunto de métricas de similaridade implementadas era pequeno.



A validação foi realizada em uma turma de Inteligência Artificial de um Curso de Ciência da Computação, onde um grupo de 12 alunos desenvolveu sistemas de RBC. Na validação a ferramenta foi disponibilizada aos alunos, que sem supervisão buscaram criar os sistemas de RBC que foram propostos. A validação foi concluída pelo especialista, que realizou um comparativo com sistemas de RBC descritos na literatura.

Buscando melhorar a ferramenta foi realizada a manutenção corretiva e evolutiva, que agora pode ser utilizada por professores no ensino da técnica de RBC. Na ferramenta foram disponibilizadas várias métricas de similaridade globais e locais que podem ser exploradas pelos usuários para cada tipo de aplicação em diferentes domínios de conhecimento.

## 6. REFERÊNCIAS

**AAMODT, A.; PLAZA, E.** Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. In: AICOM – Artificial Intelligence Communications. Vol. 7. Ed. IOS Press, EUA, 1994.

**CBR SHELL.** CBR Tools from AIAI. 2009. Disponível em <<http://www.aiai.ed.ac.uk/project/cbr/cbrtools.html>> . Acesso em 10/11/2009.

**FERNANDES, A. M. R.** Inteligência Artificial: noções gerais. Ed. Visual Books, Florianópolis, 2003.

**GAIA - Group of Artificial Intelligence Applications.** JColibri. 2009. Disponível em <<http://gaia.fdi.ucm.es/projects/jcolibri/>>. Acesso em 10/11/2009.

**JACZYNSKI, M.; TROUSSE, B.** An Object-Oriented Framework for the Design and Implementation of Case-Based Reasoners. In: German Workshop on Case-Based Reasoning, 6th, 1998, Berlin. Proceedings... Germany: German Workshop on Case-Based Reasoning, v. 1. p. 1-10, 1998.

**MYCBR.** myCBR. 2009. Disponível em <<http://mycbr-project.net/>>. Acesso em 10/11/2009.

**SIMMETRICS.** SimMetrics - open source Similarity Measure Library. 2009. Disponível em <http://www.dcs.shef.ac.uk/~sam/simmetrics.html>. Acessado em 01/11/2009.

**WANGENHEIM, C. G.; WANGENHEIM, A.** Raciocínio baseado em casos. Ed. Manole, Barueri, 2003.