Capítulo

5

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Case-Based Reasonable

**5.1 Introdução**

Este capítulo apresenta uma introdução dos **Sistemas de Raciocínio Baseados em Casos** (**CBR** – *Case-based Reasoning*). No entanto, segundo (Watson, 2003), os **CBR**s e a Gestão do Conhecimento estão relacionados. Portanto, antes de discutir os sistema de Raciocínio Baseados em Caso, discute-se o que é Gestão do Conhecimento.

**5.2 Gestão do Conhecimento**

A função da Gestão do Conhecimento, segundo (Watson, 2003), é permitir que as organizações possam alavancar suas informações e conhecimento, através de experiências. Conhecimento, e consequentemente, sua gestão, está sendo alardeado como base da futura competitividade econômica, por exemplo.

De acordo com Watson (2003), o conhecimento, agora, passa a ser vista como um ativo, a criação e o compartilhamento tem se tornado um importante fator dentro e entre as organizações. Embora, muitos autores levantam a questão em relação ao “paradoxo valor” quando considera a natureza do conhecimento, em particular sua intangibilidade e inadequação como um ativo e a dificuldade de avaliar e proteger seu valor.

**5.2.1 Definição de Gestão do Conhecimento**

Muitos autores abordam o assunto de diferentes perspectivas. Eles, portanto, têm diferentes definições. A maioria da literatura sobre gestão do conhecimento, tratam o conhecimento de forma ampla, e usa-o para cobrir tudo o que a empresa necessita para realizar suas funções. Isto pode envolver o conhecimento formalizado, patentes, leis, programas, e procedimentos, bem como o mais intangível know-how, habilidades, e experiências das pessoas. Ele também inclui a maneira como as organizações funcionam, comunica-se, analisa situações, desenvolve novas soluções para os problemas, e desenvolve novas formas de fazer negócio. Mais ainda, pode envolver questões culturais, costumes, e valores, bem como os relacionamentos entre fornecedores e clientes (Watson, 2003).

Por outro lado, Gestão inclui todas as maneiras que o conhecimento ativo de uma organização é colhido, armazenado, transmitido, aplicado, atualizado, ou gerado (Watson, 2003). Neste trabalho, centra-se na gestão do conhecimento, por meio da aplicação de uma metodologia para implementar a solução da gestão do conhecimento, nomeadamente, *case-based reasoning* (**CBR**).

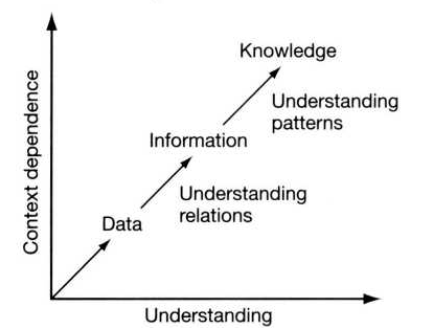
Portanto, a definição adotada é a apresentada por Watson (2003), que é “Gestão do Conhecimento envolve a aquisição, armazenamento, recuperação, aplicação, geração, e revisão do conhecimento ativo de uma organização de uma maneira controlada”.

**5.2.2 O que é Conhecimento?**

De acordo com Watson (2003), o conhecimento não existe isolado. Não é algo que pode ser pego e trancado em um cofre de uma empresa. Na verdade, alguns filósofos acreditam que o conhecimento é uma construção humana que não existe fora da mente das pessoas. Vale a pena considerar o relacionamento entre dado e informação. Os computadores têm por décadas sempre manipulado dados (em sistemas de banco de dados.

Dados, informações e conhecimento pode ser considerado, não como entidades discretas, mas como algo continuo, como ilustrado na Figura 5.1. Eles exibem uma relação com seu contexto e quantidade de entendimento quer é requerido ou impactar (Watson, 2003).

**Figura 5.1 A relação de contexto para a compreensão. Fonte:** Appling Knowledge Management: Techniques for Building Corporate Memories (Watson, 2003).



Uma noção importante a se observar na Figura 5.1 é que o conhecimento envolve o reconhecimento ou a compreensão de padrões. Isto envolve a criação de modelos mentais, exemplares, ou arquétipos (Watson, 2003).

De acordo com Watson (2003), quando existe padrão entre informação, o padrão tem o potencial de representado como conhecimento. Embora, os padrões representando conhecimento, devem ter um contexto. O contexto do padrão, proporciona um grau de previsibilidade de onde o padrão pode ser aplicado.

Existe o conhecimento explícito, que é aquele que você sabe expressá-lo ou codifica-lo. Mas, nem todo conhecimento é explícito; alguns são tácitos. Ele pode ser sentido e compreendido, mas não expresso. Mas, os sistemas de gestão do conhecimento devem lidar com ambos conhecimento explícito e tácito. Para muitos da comunidade da gestão do conhecimento, é errado tentar codificar todo conhecimento (isto é, torna-lo explícito), e a tentativa de fazê-lo, resultará em muito conhecimento tácito sendo perdido (Watson, 1997, 2003).

Dessa forma, a representação do conhecimento pelos sistemas de gestão de conhecimento deve ser flexível. O formalismo rígido de sistema especialistas baseados em regras (*rule-based expert systems*) dos anos 80 são restritivos para manipular conhecimento tácito. A representação mais discursiva de uma biblioteca de casos, tais como as usadas pelos sistema de raciocínio baseados em casos, pode ser mais capaz de lidar com o conhecimento tácito; mas você deve reconhecer que não existe formalismo que possa capturar adequadamente todo conhecimento tácito (Watson,1997, 2003).

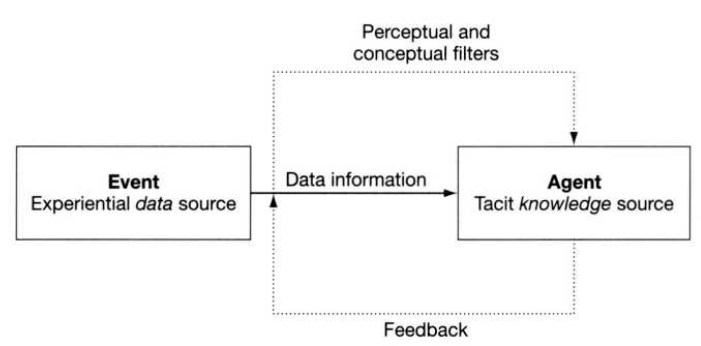
**5.2.3 Diferença entre Conhecimento, Informação e Dado**

Na literatura nem sempre fica claro o que é “gestão do conhecimento” e “Gestão da informação”. Informação e conhecimento são mais apropriadamente vistos em termos de uma dinâmica e interativa relação. A informação facilita o desenvolvimento do conhecimento, que cria mais informações que aprofundam o conhecimento, *ad infinitum*. A natureza dinâmica desta relação é ilustrada na Figura 5.2.

Olhando para a informação puramente em termos de como tenha sido processada, ou seja, olhando os dados, informação e conhecimento de forma continua, não se percebe a complexidade que existe entre os três valores. Note que o elemento feedback dentro da Figura 5.2, que ilustra a dinâmica e o relacionamento interativo entre informação e conhecimento, é um laço (Watson, 2003).

Dado é a discriminação de estado-por exemplo, preto, branco, pesado, leve-que pode ou não transmitir informação para uma pessoa, dependendo da prioridade da pessoa ou do contexto.

**Figura 5.2** **Dado, Informação e Conhecimento**. **Fonte**: Appling Knowledge Management: Techniques for Building Corporate Memories (Watson, 2003).



Pode-se caracterizar dado com uma propriedade de coisas e conhecimento com uma propriedade de pessoas, que o predispõe a agir em circunstancias particulares. Informação é um subconjunto de dados que residem em ciosas que levam uma pessoa a agir; que filtrada a partir dos dados por aparato perceptual ou conceitual da pessoa (Watson, 2003).

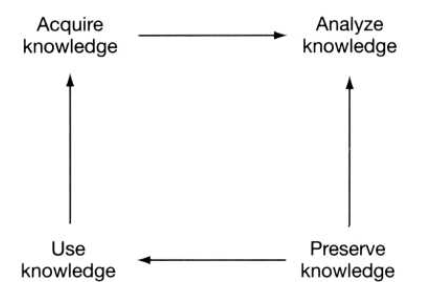
**5.2.4 Atividades da Gestão do Conhecimento**

De acordo com Watson (2003), o ato de gerenciar conhecimento pode ser caracterizado por quatro atividades:

1. Adquirir conhecimento (aprendendo, criando ou identificando);
2. Analisando conhecimento (avaliar, validar ou valorar);
3. Preservar conhecimento (organizar, representar ou manter); e
4. Usar conhecimento (aplicar, transferir ou compartilhar).

Estas atividades não existem em isolado. Em vez disso, pode-se considera-las como um ciclo, como mostra a Figura 5.3. Você pode ver esta gestão do conhecimento ciclo (***KM-cycle***) com uma simplificação do mais detalhado Case-based reasoning-cycly (CBR-cycle) que será discutido em sessões seguintes (Watson, 2003). O elemento que liga o ciclo é o uso do conhecimento, uma vez que é provável que quando o conhecimento é usado, uma nova visão sobre o conhecimento pode ser criado. Este novo conhecimento, por sua vez, deve ser adquirido, analisado e preservado para uso futuro (Watson, 2003).

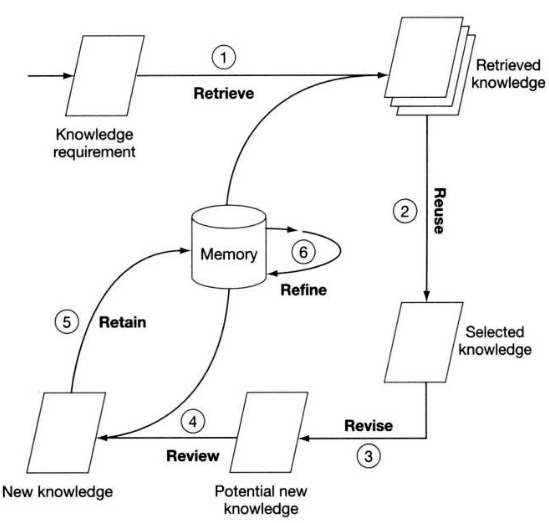
**Figura 5.3 O KM-cycle**. **Fonte**: Appling Knowledge Management: Techniques for Building Corporate Memories (Watson, 2003).



* + 1. **Uma Metodologia para o Conhecimento**

Em recente workshop realizado na Universidade de Cambridge, na Inglaterra, um grupo de pessoas ativas gestão do conhecimento e Inteligência Artificial (IA) identificou as principais atividades necessárias por um conhecimento (Watson, 2003). Essa atividades estão ilustradas na Figura 5.4.

**Figura 5.4 O funcionamento do Raciocínio Baseado em Casos** (*CBR-cycle*). **Fonte**: Appling Knowledge Management: Techniques for Building Corporate Memories (Watson, 2003).



1. O processo de recuperação, reuso, e revisão suporta a aquisição de conhecimento.
2. O processo de revisão e refinamento suporta a análise do conhecimento.
3. A proporia memória (juntamente com a recuperação e o refinamento) suportam a preservação do conhecimento.
4. Finalmente, a recuperação, reuso, e revisão suportam o uso do conhecimento.

Em sessões seguintes este processo será mais detalhado e ilustrado com estudo de casos.

Os pontos chaves aqui discutidos foram (Watson, 2003):

* O conhecimento não é estático; isto é, um sistema de gestão do conhecimento deve poder suportar a aquisição, análise, preservação, e reuso do conhecimento como um processo continuo e cíclico.
* O conhecimento existe em duas formas: conhecimento explícito, que pode ser codificado e o conhecimento tácito, que nem sempre pode ser codificado. Se a representação do conhecimento for muito formalizado, muito conhecimento tácito pode ser perdido. Assim, a representação do conhecimento em sistemas de gestão do conhecimento, deve ser flexível e discursiva.
  1. **Raciocínio Baseado em Casos (***Case-Based Reasoning* **– CBR)**

Na sessão anterior foi introduzido o ***CBR-cycle*** e como ele satisfaz os requisitos de um sistema de gestão de conhecimento. Nesta sessão será detalhado cada processo do ***CBR-cycle***.

De acordo com Watson (2003), o **CBR** usa o conceito de similaridade para recuperar coisas (casos) de uma biblioteca (uma base de casos). Casos são usados em muitas situações; por exemplo, para fornecer informações de produtos para um cliente, resolver problemas em uma central de informações ao clientes, configurar equipamentos de manufatura, ou resolver problemas financeiros complexos.

De acordo com Watson (2003), nós resolvemos problemas usando experiências adquiridas e que podemos aprender novas experiências. O Raciocínio baseado em Casos pode ser descrito por seis atividades ocorrendo em ciclo, como discutido na sessão anterior.

Este ciclo é constituído de seis processos:

1. Recuperar
2. Reusar
3. Revisar
4. Avaliar
5. Manter
6. Refinar

**5.3.1 Definição**

Para Watson (2003), a ideia básica em um sistema CBR é que, para um domínio particular, os problemas a serem resolvidos tendem a ser recorrentes e repetir-se com pequenas alterações em relação a sua versão original. Dessa forma, soluções anteriores podem ser reutilizadas também com pequenas alterações.

Riesbeck e Schank (1996), definem **CBR** como “Um sistema de CBR resolve problemas por adaptar soluções que foram utilizadas para resolver problemas anteriores”.

Em seguida será detalhado cada um desses processos, mas primeiro é necessário entender o que é recuperar, reusar, e revisar, e assim, casos.

**5.3.2 Representação de Casos**

De acordo com Watson (2003), casos são registros de experiências que contém conhecimento, que pode ser ambos explicito e tácito. Por exemplo, ele pode ser casos de históricos de pacientes no sentido médico, detalhes de empréstimos bancários, ou descrição de situações de erros de equipamentos. Cada um desses registros de casos compreende:

* Uma descrição
* O respectivo resultado ou solução

Assim, um caso tipicamente compreende um par problema e solução. Uma coleção de casos é chamado de uma base de casos, justamente como uma base de registros é chamado de banco de dados (Watson, 2003).

Uma forma de visualizar é em termos de espeço do problema e espaço de solução. Na Figura 5.5 vê-se que um caso individual é composto de dois componentes: uma descrição do problema e o armazenamento da solução. Estes residem respectivamente no espaço do problema e no espeço de solução. A descrição do problema a ser resolvido é colocado no espaço de problema. Recupera-se o caso mais similar a descrição do problema, e sua solução, armazenada encontrada. Se necessário, ocorrem adaptações, e uma nova solução é armazenada. Este modelo conceitual de CBR, assume que há um mapeamento direto de um-para-um entre o problema e os espaços de solução. (Watson, 1999).

Bases de Casos divide-se em duas grandes categorias (Watson, 2003):

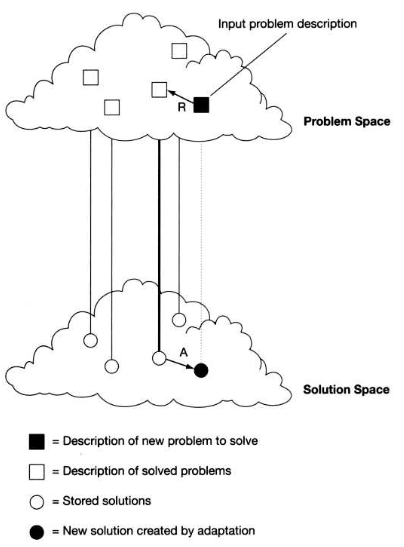
* **Em bases de Casos homogêneas**: todos os casos compartilham os mesmos dados ou estrutura de registros; isto é, casos têm os mesmos atributos mas variando os valores.
* **Em Bases de Casos Heterogêneas**: casos têm estrutura de registros variados; isto é, casos podem ter diferentes atributos e valores variados.

Um exemplo de caso homogêneo pode ser o caso de venda de casa, onde na base de casos de casas tem os mesmos atributos que são suficientes para descrever uma casa. Então, um corretor, que tenha acesso a essa base de casos, pode assumir que já tenha todas as informações para realizar a transação. Embora, se o corretor ainda não tiver esta propriedade na base de casos, ele pode facilmente criar um registro nessa base de casos.

Um exemplo de base de casos heterogênea poderia ser uma base de casos de diagnósticos de pacientes. Registros de pacientes contêm um lote de informações em comum, tais como idade, tipo sanguíneo, pressão sanguínea, mas também muitas informações que são únicas para cada paciente, por exemplo, histórico médico, tratamento, e prognósticos.

Quando desenvolve-se uma base de casos heterogênea, os desenvolvedores nunca pode assegurar que ele tenha um conjunto completo de características (Watson, 2003). Por exemplo, uma base de dados de diagnósticos de pacientes, os desenvolvedores poderia não listar todas as possíveis condições médicas, sintomas, e testes que uma pessoa poderia ter.

**Figura 5.5 Os espaços de problema e de solução**. **Fonte**: Appling Knowledge Management: Techniques for Building Corporate Memories (Watson, 2003).



Dentro de um **Caso** pode-se armazenar muitos tipos de dados, tais como nomes, identificação de produto, valores como custo, temperatura, e notas textuais. Algumas ferramentas de CBR também suportam dados com características de multimídia, tais com imagens, sons e vídeo (Watson, 1999, 2003).

Não há um consenso por parte da comunidade de CBR, que informações exatamente, poderia ser um Caso. Embora, duas medidas pragmáticas poderia ser tomadas para se decidir o que poderia ser representada em Casos: a funcionalidade da informação e a facilidade de aquisição da informação (Watson, 1999).

**5.3.3 Indexação**

Muitos sistemas de Banco de Dados utilizam-se de índices para agilizar a recuperação de dados. Um índice é computacionalmente, uma estrutura de dados que pode ser realizada em memória, tornando a localização da informação, muito rápida, sem ter que fazer a busca do(s) registro(s) no disco. O CBR também faz uso de índice para agilizar a recuperação de Casos. A informação dentro de um Caso, é de dois tipos (Watson, 1999):

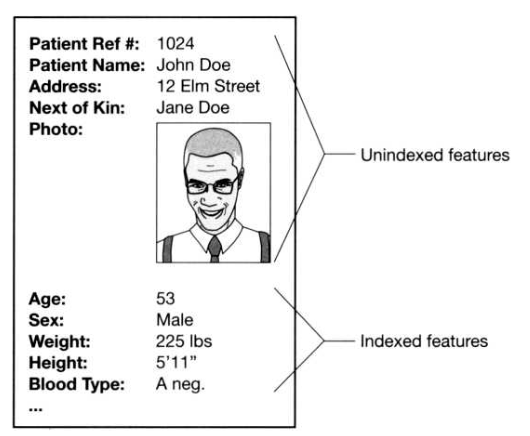
1. Informação Indexadas, que á usada para recuperar um Caso.
2. Informação não indexada, que fornecem informações contextuais para o usuário, mas que não são usadas diretamente para recuperação de Caso.

Por exemplo, em um sistema médico, pode-se usar as informações do paciente, tais como idade, sexo, altura, e peso como características a serem indexadas, que pode ser usada para recuperação do Caso e, outras informações, tais como nome, endereço e fotografia como informações contextuais, ou seja, não indexadas, que não podem ser usadas para recuperação de Casos. A Figura 5.6 ilustra este exemplo.

Como diretrizes, os índices devem:

* Ser preditivo.
* Indicar o propósito em que o Caso será usado.
* Ser abstrato o suficiente para permitir ampliar a base de casos e seu uso no futuro.
* Ser concreto o suficiente para ser reconhecido em futuras situações.

**Figura 5.6 Informações indexadas e não-indexadas**. **Fonte**: Appling Knowledge Management: Techniques for Building Corporate Memories (Watson, 2003).



Se tomarmos como exemplo um sistema bancário. Informações dos clientes, tais como nome e telefone, são claramente não preditivas, você não decidir emprestar dinheiro a um cliente como base em seu nome e telefone. No entanto, informações tais como renda, e seus compromissos financeiros, tais como empréstimos habitacionais, pagamentos de carros, e seguro de vida, e assim por diante, são claramente preditivos. Dessa forma, informações como renda e compromissos financeiros podem ser escolhidos como índices e nome e telefone como informações contextuais (Watson, 1999).

A escolha do índice, tanto pode ser manual com automatizada. Escolher um índice manualmente, envolve decidir o propósito do Caso em respeito ao objetivo do sistema e decidir em que circunstancias o Caso vai ser útil (Watson, 1999).

Existem um número crescente de métodos de indexação automática na literatura, incluindo: MEDIATOR, CHEF e CYRUS, etc.

Várias ferramentas de CBR presentes no mercado suportam a identificação de índices de casos automaticamente, para aplicações práticas, índices pode ser escolhido automaticamente, manualmente ou ambas técnicas (Watson, 1999, 2003).

**5.3.4 Aquisição (Storage)**

A representação do Caso é um importante aspecto no projeto de sistemas CBR, no que se refere a visão conceitual do que é representado no Caso e levando-se em conta os índices que caracterizam os casos. A base de Casos poderia ser organizada em uma estrutura gerenciável que suporte pesquisas e métodos de recuperação eficientes. Deve-se ser encontrado um equilíbrio entre os métodos de armazenamento que preserve a riqueza de casos e seus índices e métodos que simplifique o acesso e recuperação de casos relevantes (Watson, 1999). Este métodos são usualmente chamados de modelos de memória de casos (*case-memory models*). Os dois modelos de memória de casos mais influentes na academia são o modelo de memória dinâmica de Schank e Kolodner, e o modelo categoria exemplares de Porter e Bareiss. Estas técnicas ainda são bastantes utilizadas pela comunidade de Ciência Cognitiva, mas nenhuma ferramenta comercial de CBR usam essas técnicas. Ao invés disso, muitas base de casos utilizam estruturas simples de arquivos planos (flat files), ou estruturas de bancos de dados relacionais e, usa índices para se referir aos casos (Watson, 1999).

**5.3.4.1 Modelos de Memória Dinâmica**

O modelo de memória dinâmica é composto principalmente de pacotes de organização de memória (**MOP**s), que são frames que compõem uma unidade básica da memória dinâmica. Eles representam conhecimento sobre classes de eventos de duas maneiras:

* Instâncias, que representam casos, eventos ou objetos.
* Abstrações, que representam versões generalizadas de instâncias ou de outras abstrações.

O desenvolvimento da teoria mais geral de Schank levou aos pacotes de organização de memória episódicos (**E-MOP**s), implementados no sistema **CYRUS** (Kolodner, 1993). A ideia básica é organizar casos específicos que partilham propriedades similares sob uma estrutura mais geral, ou seja, um **E-MOP**. Uma **E-MOP** contém os casos, as propriedades comuns entre eles e as características que os diferenciam.

**5.3.4.2 Modelos de Categoria Exemplares**

Esse modelo considera que os casos no mundo real podem ser vistos como exemplares de acontecimentos. Neste caso, uma memória de casos é uma rede semântica de categorias de casos ligados por relações semânticas de hierarquias, de semelhança ou de diferenças. Cada caso é associado a uma categoria e suas características têm importância distintas para enquadrá-lo ou não na categoria. Características similares de um caso apontam para as de outro caso ou categoria. Dessa forma, compõe-se uma rede de conhecimento genérico do domínio que permite alguma recuperação do raciocínio do sistema para gerar explicações. Para gerar um novo caso, é buscado um caso semelhante no banco de casos. Se houver pequenas diferenças entre os dois, apenas um deles é retido, ou é armazenado uma única combinação dos dois (Watson, 2003).

**5.3.5 Recuperação**

Dada uma descrição de um problema, um algoritmo de recuperação deveria encontrar os casos mais similares à situação atual, utilizando-se dos índices da memória de casos. Os algoritmo baseiam-se nos índices e na organização de memória para guiar a busca dos casos potencialmente úteis.

De acordo com Watson (2003), a recuperação de casos está diretamente relacionado e dependente ao método de indexação usado. Em geral, duas técnicas são correntemente usadas pelas ferramentas de **CBR** comerciais: algoritmo de vizinhança (***Nearest-Neighbor***) e Indutivo.

**5.3.5.1 Algoritmo de Vizinhança**

Esse método, segundo Watson (2003), baseia-se na comparação entre um novo caso e aqueles armazenados na base de casos, utilizando uma soma ponderada das suas características. Para isso é necessário atribuir um peso a cada uma das características que descrevem o caso e que serão utilizadas na recuperação.

Na prática, a similaridade (isto é, a proximidade) do caso destino para o caso fonte para cada atributo é determinado. Esta medida é multiplicado por um fator peso. Então a soma da similaridade de todos os atributos é calculada. Esta pode ser representada por uma equação relativamente simples

**(5.1)**

Onde

T é o caso destino

S é o caso fonte

n é o número de atributos em cada caso

i é um atributo individual de 1 até n

é a função de similaridade para atributo i nos casos T e S

w é o peso do atributo i

Algoritmos de similares a este são usados por muitas ferramentas CBR para realizar recuperação do caso mais similar. Similaridade são normalmente para cair dentro da faixa de 0 para 1 (onde o significa totalmente dissimilar e 1 exatamente similar) ou usando um percentual, onde 100% é totalmente similar (Watson, 1999; 2003).

**5.3.5.2 Algoritmo de Indução**

Indução é uma técnica desenvolvida por pesquisadores de Aprendizado de Máquinas para extrair regras ou construir de dados passados. Em sistema CBR, a base de casos é analisada por algoritmo de indução para produzir uma árvore de decisão que classifica (ou indexa) os casos. O algoritmo de indução foi amplamente usado pela ferramenta CBR chamada ID3.

**5.3.6 Adaptação**

A tarefa final do Sistema CBR é adaptar a solução associada a um caso recuperado para as necessidades do problema corrente. Quando uma situação é fornecida, o algoritmo de recuperação traz o melhor caso que ele encontrar para a memória. Normalmente, o caso selecionado não atende perfeitamente coma descrição do problema do usuário. Ou seja, existem diferenças entre o problema do usuário e o caso contido no banco de casos que devem ser levadas em conta. Então, o processo de adaptação procura por diferenças salientes entre as duas descrições e aplica regras de forma a compensá-las. Em geral, existem dois tipos de adaptação em CBR (Watson, 2003):

* **Adaptação Estrutural** - as regras de adaptação são aplicadas sobre a solução armazenada junto aos casos.
* **Adaptação Derivacional** – o algoritmo reusa os algoritmos, métodos ou regras que geraram a solução que consta no banco de casos para gerar uma nova solução para o problema corrente. Neste método, a sequência que construiu a solução original deve ser armazenada juntamente com o caso na memória de casos. O algoritmo de adaptação derivacional exige uma perfeita compreensão dos casos armazenados e da forma como as soluções foram geradas.

Segundo Watson (2003), várias técnicas tem sido usadas em sistema CBR. Incluindo as seguintes:

* **Adaptação nula** – ele simplesmente aplica a solução recuperada ao problema corrente sem modificação. Adaptação nula é útil para problemas envolvendo raciocínio complexo mais com solução simples. Por exemplo, em um sistema para concessão de crédito, embora seja necessário coletar muitas informações do cliente, a solução final de conceder ou rejeitar o crédito é direta.
* **Ajuste por parâmetros** – é uma técnica de adaptação estrutural que compara parâmetros específicos entre o caso recuperado e o novo para modificar a solução armazenada na direção apropriada. Esta técnica foi usada no sistema CBR chamado JUDGE, que recomenda sentenças mais curtas para crimes menos violentos.
* **Reinstanciação** – instancia uma nova solução para um caso recuperado do banco de casos com novas características adequadas ao problema do usuário. Por exemplo, o sistema CBR CHEF, que a partir de uma receita existente, criar uma nova receita.
* **Substituição derivacional** – repete o método, ou parte do método que gerou uma solução armazenada em um caso similar de forma a obter a solução para o novo caso, substituindo os atributos distintos. Como no sistema BOGART que reaplica os planos de geração de projetos para novos problemas.
* **Repara guiado por modelos** – utiliza um modelo casual para adaptar as soluções armazenadas ao problemas do usuário. O sistema CBR, CELIA, utiliza-o para aprendizado e diagnóstico de problemas mecânicos de automóveis.

Finalizando, de acordo com Watson (2003), a adaptação é útil em muitas situações. Mais não significa que seja essencial. Muitos dos sistema CBR comerciais não usam adaptação para tudo. Eles simplesmente reusam a solução sugerida para o melhor caso correspondente (i.e., adaptação nula) ou eles deixam a adaptação para as pessoas.

* 1. **Conclusões**

Pode-se concluir que, O **Raciocínio Baseado em Casos** é um método em que problemas novos são resolvidos através de soluções adaptadas que foram usadas para resolver problemas mais antigos.

Um **Caso** é uma pedaço contextualizado de conhecimento que representa uma experiência. Ao se analisar cada caso, se tem a descrição do problema e a solução armazenada. Caso já exista um problema semelhante já anteriormente armazenado no banco de dados, a solução será recuperada. Porém, se não existir um caso similar, a descrição desse novo problema será enviado ao espaço de problemas, recuperando o caso com o problema mais similar possível, criando uma nova solução (Watson,1997).

Devido a essas características, o **RBC** é indicado para tarefas de segmentação e categorização.

A recuperação de casos é profundamente relacionada e dependente do método de categorização utilizado. As duas técnicas utilizadas atualmente são a de **recuperação por vizinho mais próximo** e **recuperação indutiva**. Na recuperação por vizinho mais próximo, deverão ser definidos os índices dos casos e os seus respectivos pesos, dependendo do problema a ser resolvido. Esses índices deverão ser previsíveis, identificar o propósito em que os casos serão utilizados, serem abstratos o bastante para permitir o uso posterior da base de dados, e serem concretos o bastante para serem reconhecidos no futuro (Watson, 1997). Como explicitado anteriormente, o caso similar será utilizado como solução do problema. No caso de recuperação por indução, uma árvore de decisão é utilizada, classificando os casos.