

UNIVERSIDADE FEDERAL DE ALAGOAS
PROGRAMA MULTIDISCIPLINAR DE PÓS-GRADUAÇÃO EM
MESTRADO EM MODELAGEM COMPUTACIONAL DO
CONHECIMENTO



Dissertação de Mestrado

Um Sistema Evolutivo de Apoio a Tomada de Decisão: Um Estudo de Caso em Compra Eletrônica

Genildo Nascimento dos Santos
genildo16@gmail.com

Orientadora:
Roberta Vilhena Vieira Lopes

Maceió, novembro 2009.

Genildo Nascimento dos Santos

Um Sistema Evolutivo de Apoio a Tomada de Decisão: Um Estudo de Caso em Compra Eletrônica

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Curso de Mestrado em Modelagem Computacional de Conhecimento do Instituto de Computação da Universidade Federal de Alagoas.

Orientadora:

Roberta Vilhena Vieira Lopes

Maceió, novembro 2009.


Dissertação apresentada como requisito parcial para a obtenção do grau de Mestre em Modelagem Computacional de Conhecimento pelo Programa Multidisciplinar de Pós-Graduação em Modelagem Computacional de Conhecimento, da Universidade Federal de Alagoas, aprovada pela comissão examinadora que abaixo assina:



Profa. Dra. Roberta Vilhena Vieira Lopes

UFAL – Instituto de Computação

Orientadora



Prof. Dr. Evandro de Barros Costa

UFAL – Instituto de Computação

Co-orientador



Prof. Dr. Luis Cláudius Coradine

UFAL – Instituto de Computação

Examinador



Prof. Dr. Robson do Nascimento Fidalgo

UFPE – Centro de Informática

Examinador

Maceió, novembro de 2009.

Agradecimentos

Em primeiro lugar, agradeço a Deus por ter permitido que, apesar dos obstáculos pessoais pelos quais passei, eu concluísse este trabalho.

Agradeço, e muito, a minha esposa por sempre acreditar no meu potencial, me ajudando em todos os momentos difíceis.

A minha orientadora Dra. Roberta Vilhena Vieira pela paciência, orientação e acompanhamento das tarefas deste estudo.

Ao professor Dr. Evandro de Barros Costa pela ajuda e co-orientação de diversas atividades.

A todos os professores que me incentivaram e me ensinaram a lutar e ultrapassar barreiras.

Em destaque para meu grande amigo Hélio pelo apoio dado dentro e fora da universidade durante todo momento que precisei de ajuda, e principalmente, pela amizade dada durante anos.

Sumário

Lista de Abreviaturas e Siglas	3
Lista de Figuras	5
Lista de Tabelas	7
Resumo	8
Abstract.....	9
1. Introdução.....	10
1.1 Sistema de Recomendação em um ambiente E-Commerce.....	10
1.2 Problemática e Motivação	11
1.3 Objetivos.....	13
1.4 Organização da Dissertação	14
2. Fundamentação Teórica.....	16
2.1 Sistemas de Informação	16
2.1.1 Sistemas de Informação Baseados em Computadores	18
2.1.2 Sistemas de Apoio à Decisão.....	20
2.1.3 Sistemas de IA	22
2.1.4 Sistemas de Recomendação.....	24
2.2 Computação Evolutiva.....	25
2.3 Algoritmos Genéticos	28
2.4 Algoritmo Genético de Holland.....	29
2.5 Manutenção da Base de Conhecimento	35
3. O Algoritmo Genético através de Tipos Abstratos de Dados.....	37
3.1 Algoritmo Genético via Tipo de Dados Abstrato	37
3.1.1 Gene.....	38
3.1.2 Cromossomo.....	38
3.1.3 População	39

3.1.4	Grau de Adaptação	39
3.1.5	Operadores.....	39
3.2	Algoritmo Genético pelo GAADT	41
3.3	Exemplo	41
4.	Modelagem do Sistema de Apoio à Compra Eletrônica	45
4.1	Esquema Relacional.....	45
4.2	Instanciação pelo GAADT.....	46
4.3	Resultados	53
4.3.1	Desempenho	54
4.3.2	Trabalhos Relacionados.....	55
5.	Conclusões.....	59
5.1	Contribuições e Relevâncias	59
5.2	Dificuldades Encontradas	60
5.3	Trabalhos Futuros	61
Apêndice A: Análise do Sistema de Apoio à Compra Eletrônica		62
Apêndice B: Fase de projeto do Sistema		71
Referências Bibliográficas		79

Lista de Abreviaturas e Siglas

AA	Ambiente Aplicativo
AE	Algoritmo Evolutivo
AFC	Axiomas de Formação do Cromossomo
AFG	Axiomas de Formação de Genes
AO	Ambiente Operacional
BD	Bancos de Dados
CBR	Case-Based Reasoning
CDI	Ciclo de Desenvolvimento Iterativo
DSS	Decision Support Systems
ER	Entidade-Relacionamento
FAD	Ferramentas de Apoio à Decisão
GAADT	Genetic Algorithm Abstract Data Types
IA	Inteligência Artificial
LAN	Local Area Network
ODMG	Object Data Management Group
OO	Orientação a Objetos
RBC	Raciocínio Baseado em Casos
SAD	Sistemas de Apoio à Decisão
SE	Sistema Especialista
SGBD	Sistema Gerenciador de Banco de Dados
SIBC	Sistema de Informação Baseado em Computador
SIG	Sistema de Informação Gerencial
SIT	Sistemas de Informação Transacionais
SQL	Structured Query Language
SR	Sistema de Recomendação

URL	Uniform Resource Locator
UML	Unified Modeling Language
VAN	Value-Added Networks
WWW	World Wide Web
XML	eXtensible Markup Language
SOM	Self-Organizing Map

Lista de Figuras

Figura 2.1 - Elementos de um SIBC.....	18
Figura 2.2 - Ambiente de apoio à decisão.....	21
Figura 2.3 - Uma visão conceitual dos sistemas de Inteligência Artificial.....	23
Figura 2.4 - Estrutura do Sistema de Recomendação.....	24
Figura 2.5 - Esquema geral dos algoritmos evolutivos.....	26
Figura 2.6 - Pseudocódigo para algoritmo evolutivo.....	26
Figura 2.7 - Evolução de uma população durante uma geração conforme (Zuben, 2000).....	27
Figura 2.8 - Ilustração de um AG através de diagrama de estado.....	30
Figura 2.9 - Pseudocódigo do algoritmo R.....	31
Figura 2.10 - Exemplo do operador de cruzamento de Holland.....	32
Figura 2.11 - Exemplo do operador de inversão de Holland.....	33
Figura 2.12 - Exemplo do operador de mutação de Holland.....	34
Figura 3.1 - Exemplo de genes conforme problema em (Furtado, 2006).....	42
Figura 3.2 - Exemplo de cromossomos conforme problema em (Furtado, 2006).....	43
Figura 3.3 - Exemplo de população conforme problema em (Furtado, 2006).....	43
Figura 4.1 - Esquema relacional proposto.....	46
Figura 4.2 - Tela de extração através do site Buscapé.....	47
Figura 4.3 - Pseudocódigo da função $AG(P_n, t, k)$	52
Figura 4.4 - Tela de compactação do software protótipo.....	53
Figura A.1 - Diagrama de caso de uso do sistema.....	64
Figura A.2 - Esquema conceitual do sistema.....	64
Figura A.3 - Diagrama de sequência solicita extração.....	65
Figura A.4 - Diagrama de sequência solicita recomendação.....	66
Figura A.5 - Diagrama de sequência retira relatório.....	66
Figura A.6 - Diagrama de sequência cadastra experiência.....	67

Figura A.7 - Diagrama de sequência solicita manutenção.....	67
Figura A.8 - Diagrama de estados para a classe extração.....	68
Figura A.9 - Diagrama de estados para a classe recomendação.....	69
Figura A.10 - Diagrama de estados para a classe relatório.....	70
Figura B.1 - Arquitetura do sistema.....	74
Figura B.2 - Diagrama de colaboração de uma solicitação de extração.....	75
Figura B.3 - Diagrama de colaboração de uma solicitação de recomendação.....	75
Figura B.4 - Diagrama de colaboração de uma solicitação de manutenção.....	76
Figura B.5 - Diagrama de colaboração de uma impressão de relatório.....	76
Figura B.6 - Diagrama de classe do sistema.....	77
Figura B.7 - Diagrama de implantação do sistema.....	78

Lista de Tabelas

Tabela 2.1 - Descrição dos principais fundamentos de um sistema de informação.....	17
Tabela 2.2 - Descrição dos elementos de um SIBC.....	18
Tabela 2.3 - Descrição dos principais vocábulos biológicos para o computacional.....	29
Tabela 4.1 - Descrição das entidades existentes no modelo relacional proposto.....	46
Tabela 4.2 - Exemplo de ocorrências armazenadas na base de dados.....	47
Tabela 4.3 - Modelo da base B do ambiente.....	48
Tabela 4.4 - Modelo da função $valor(v)$	49
Tabela 4.5 - Modelo do AFG do ambiente.....	49
Tabela 4.6 - Modelo do cromossomo C do ambiente.....	49
Tabela 4.7 - Modelo do AFC do ambiente.....	49
Tabela 4.8 - Resultado da implementação do cromossomo para a ocorrência 1.....	50
Tabela 4.9 - Modelo da função $grau(g)$	50
Tabela 4.10 - Modelo da função $domi(g_1, g_2)$	50
Tabela 4.11 - Modelo da função $tipo(g)$	51
Tabela 4.12 - Modelo da função $AG(P_n, t, k)$	51
Tabela 4.13 - Nova base de dados, com cardinalidade menor que a base de dados inicial.....	52
Tabela 4.14 - Tempo de resposta.....	54
Tabela 4.15 - Resumo do desempenho de outros sistemas de aconselhamento.....	56
Tabela 4.16 - Pergunta chave de cada sistema de aconselhamento.....	57
Tabela 4.17 - Comparação do protótipo GAADT com outros sistemas.....	58
Tabela B.1 - Funções básicas.....	71

Resumo

O desempenho da manipulação dos dados por um sistema de apoio à decisão, em um ambiente de comércio eletrônico, pode ser considerado um processo complexo devido à grande quantidade de informação proveniente do mecanismo de busca adotado para aquisição de equipamentos eletroeletrônicos. Este processo poderá aumentar exponencialmente no decorrer do tempo devido ao crescimento que influenciará no tempo de resposta da execução das tarefas do mecanismo de busca para o usuário. Este trabalho propõe uma abordagem híbrida utilizando banco de dados e algoritmo genético baseado em tipos abstratos de dados, denominado GAADT, para reduzir a dimensionalidade dos dados providos em uma base de conhecimento e auxiliar nas negociações da obtenção de produtos.

Palavras-chave: Sistemas Baseado em Casos, Inteligência Artificial, Algoritmo Genético.

Abstract

The performance of data manipulation through a search engine in e-commerce systems using decision support systems can be considered a complex process due to lots of information and may grow exponentially, according to research performed by the user for acquisition of equipment electronics. The import information collected by the system may influence the time to answer during the course of the search engine in accordance with the user's request. This works proposes a hybrid approach using database and genetic algorithm based on abstract data types (GAADT), to reduce the data of database dimensionality and helpful in the selection of products.

Keywords: Case-Based Reasoning Systems, Artificial Intelligence, Genetic Algorithm.

Capítulo 1

Introdução

1.1 Sistema de Recomendação em um ambiente E-Commerce

Um grande número de pessoas tem acessado cada vez mais à *Internet* e a tendência é que esta quantidade aumente (Costin, et al., 2007). E com este grande número de acesso também cresce o número de serviços oferecidos (Wei, et al., 2002). Basta o usuário entrar na *Internet* para começar a confrontar com vários termos, a maioria deles com a nomenclatura *e* (*e-business*, *e-commerce*, *e-mail*, por exemplo). Este tipo de nomenclatura sempre aparece diante de palavras, chamadas de *buzzwords*, que tem alguma relação com o termo em inglês *electronic* (Turban, et al., 2004). Essas nomenclaturas são rótulos que foram dados a certas atividades no mundo dos negócios na *Internet*. Apesar de estas expressões serem normalmente em inglês, com o tempo passaram a ser palavras comuns em todo mercado mundial da *Internet* (Turban, et al., 2004).

Considerada no mundo moderno uma ferramenta indispensável, a *Internet* fascina as pessoas devido aos seus recursos, funções e aplicabilidades. Tornando possível aos usuários saber seu extrato bancário, fazer aplicações, transferências, adquirir remédios, informações das mais diversas, programar suas férias, contratar serviços e até mesmo comprar um carro sem sair de casa (Cortez, 2004). A *Internet* veio para causar notável mudança no mundo atual com suas possibilidades de serviços em todas as áreas. Segundo Turban (2004), as empresas estão interessadas em atuar no comércio eletrônico, também conhecido como *e-business*, por ser o mais atraente e promissor dos mercados econômicos, onde as informações sobre produtos e serviços são adquiridas em frações de segundos. Assim, o interesse comercial

passou de propagandas para sites de comércio eletrônico. Pois, a *Web* trouxe ao usuário o recurso da transparência de transações executadas no servidor, a possibilidade de atualizar a tecnologia empregada sem prejuízo à aplicação original, além de fornecer ao usuário final a facilidade de acessar o sistema a qualquer momento, bastando ter um computador conectado à *Internet*.

O foco deste trabalho está na contribuição que a inteligência artificial pode oferecer na modelagem computacional do processo de apoio à decisão. Mostrando ser possível melhorar o processo de tomada de decisões no ambiente *e-commerce* através de um sistema inteligente usando algoritmo genético como principal mecanismo de inferência. Facilitando a vida dos usuários que usam a *Internet* como meio de compra, venda ou consulta para tomada de decisão. A seguir serão apresentados o contexto e a problemática do enfoque deste trabalho, juntamente com as motivações que levaram ao seu desenvolvimento, descrição dos principais objetivos e, por último, apresentação da organização do texto.

1.2 Problemática e Motivação

No comércio eletrônico, todas as transações de negócio ocorrem através de redes. Onde temos o processo de comprar e vender eletronicamente bens, serviços e informações (Cortez, 2004). Esta atividade ocorre normalmente na *Internet*, mas também pode ser realizado através de redes particulares chamadas de VANs¹ (*Value-Added Networks*). Estas redes adicionam serviços de comunicação as redes transmissoras comuns preexistentes de LANs (*Local Area Networks*) ou em uma máquina computadorizada (Turban, et al., 2004). O comércio eletrônico pode ser definido como a realização de toda cadeia de valor dos processos de negócios num ambiente eletrônico, por meio da aplicação intensa das tecnologias de comunicação e de informação, atendendo aos objetivos de negócios. Os processos podem ser realizados de forma completa ou parcial, incluindo as transações negócio a negócio, negócio a consumidor e intra-organizacional, numa infra-estrutura predominantemente pública de fácil e livre acesso e baixo custo.

Com as várias possibilidades de negócio aumentou o uso de computadores e, conseqüentemente, por conta do progresso da *Internet*, cresceu a demanda e uso de sistemas com banco de dados para coleta e análise dos dados gerados no ambiente *e-commerce*. Assim

¹ São redes de comunicação de dados oferecidas por empresas públicas ou privadas e que oferecem vários serviços de conexão, com ênfase em padrões abertos.

a tecnologia de banco de dados, inicialmente usadas por cientistas e técnicos, também passa a ser usada por pessoas comuns que serão clientes de produtos ou serviços, como por exemplo, fazendo compras, controlando suas contas bancárias e reservando passagens ou hotéis sem sair de casa.

Normalmente, em ambientes de *e-commerce* é disponibilizado um mecanismo de busca para agilizar a aquisição do produto ou serviço desejado para um cliente, tendo como retorno uma lista dos produtos pesquisados. Um dos desafios deste ambiente está relacionado ao processo de aquisição de produtos de boa qualidade com preço baixo. Os mecanismos de buscas tradicionais poderão inibir o cliente a comprar um produto de qualidade com bom preço devido à grande quantidade de informação na consulta gerada. O emprego de um sistema de apoio à decisão irá utilizar-se dessa busca para selecionar as informações de maior relevância (aqueles de preço baixo e de boa qualidade), gerando uma base de dados de conhecimento. Quanto maior for a coleta de dados para o sistema de apoio à decisão, maior será a base de conhecimento. Ou seja, a base de conhecimento poderá crescer exponencialmente podendo influenciar no processo de análise.

Por conta dessas possibilidades, quanto maior o número de transações ocorridas, maior o volume de dados envolvidos e, conseqüentemente, um aumento no tamanho físico do banco de dados de um sistema de apoio a essas transações.

Um banco de dados é uma coleção de dados relacionados (Elmasri, et al., 2000), em que se referem às características de pessoas, fatos, objetos ou eventos. O estudo de banco de dados (BD) relacionais permitiu a construção de sistemas gerenciadores de banco de dados (SGBD) capazes de oferecer acesso e transformar estes dados em informações. Permitindo, em diversos contextos, criar, atualizar e extrair informações de suas bases de dados.

Com essa grande massa de informação armazenada em um banco de dados, se fez necessária a criação de sistemas inteligentes capazes de extrair conhecimento para auxiliar ao usuário a tomadas de decisão. Atualmente, existem vários bancos de dados com o propósito de fornecer dados para extração de conhecimento. Cada um dos bancos de dados armazena um aspecto particular em um determinado contexto.

Um sistema de apoio à decisão utilizando-se de técnicas simples, como eliminar tuplas redundantes, efetuar filtragem por período ou eliminar dados antigos, não terá uma garantia de melhoria do processo de tomada de decisão para maximizar o desempenho do processo de resposta do sistema. Pois, tendo um banco de dados sem redundância em suas tuplas, fica complexa a escolha do ponto de seleção de tuplas para o filtro ou a exclusão dos dados. E,

ainda assim, teríamos o risco de ter um banco de dados com grande volume para ser interpretado por algum mecanismo de inferência.

Segundo Wanderley (1998), um processo de apoio à decisão, em negociação², terá que constar:

- A descrição da situação de forma adequada, procurando fatos e coletando informações, necessárias para se definir adequadamente a situação problema, ou seja, o que se está negociando;
- Optar por uma alternativa de solução onde seja possível formular um número adequado de possibilidades do que se está negociando. Onde a melhor solução de que se dispõe é tão boa quanto a melhor alternativa formulada;
- Levar em conta soluções com pontos negativos;
- Planejar adequadamente a implantação da decisão tomada, estabelecendo mecanismos de controle.

Sendo assim, a principal motivação desta dissertação é a criação de um modelo de apoio à negociação e decisão que forneça uma boa barganha³ baseada em negociações passadas e uma redução no volume dos dados do próprio sistema, não utilizando de estratégias de eliminação de tuplas redundantes ou filtragem por período.

1.3 Objetivos

Esta dissertação trata da análise, especificação e implementação de um sistema de apoio à decisão, utilizando-se das tecnologias de Inteligência Artificial (IA), Orientação a Objetos (OO) e Linguagem de Modelagem Unificada (do inglês *Unified Modeling Language*, UML). Apresentaremos a especificação e detalhes do desenvolvimento de um protótipo de um sistema evolutivo de apoio à compra eletrônica, com dados vinculados a *Web*, utilizando-se de técnicas de banco de dados relacionais e algoritmo genético baseado em tipo de dados abstratos (GAADT).

² É o processo de se buscar um acordo proveito de uma consequência de decisões entre as partes (Kraus, 2001).

³ Acordo ou processo que satisfaz aos interesses de um ou ambos participantes da negociação (Wanderley, 1998).

Esse trabalho tem como objetivos principais:

- Especificar uma instância do GAADT para apoio à tomada de decisão em domínio *e-commerce*. A especificação descreve tanto as estruturas de tipos quanto os operadores genéticos de cruzamento e mutação;
- Especificar um modelo utilizando uma linguagem orientada a objetos, desenvolvendo um protótipo que ilustre o uso e funcionamento do modelo criado;
- Verificar a capacidade do modelo em realizar o apoio à decisão dentro de um domínio de compras *on-line*;
- Realizar a descrição de um esquema relacional que atenda a representação das informações capturadas em um ambiente *Web*.

1.4 Organização da Dissertação

Para cumprir os objetivos desta dissertação os capítulos foram organizados em cinco, incluindo esta introdução, e dois apêndices.

O capítulo 2 busca elaborar um levantamento geral e teórico sobre os sistemas de informação e sua utilidade para a implementação, gerenciamento de dados e informações, abordado no comércio eletrônico que é o meio de comunicação em que as técnicas e o protótipo de um sistema de apoio à decisão estarão sendo divulgados. Além disso, neste capítulo é feita uma introdução à Inteligência Artificial e a Computação Evolutiva, um esboço teórico sobre o primeiro método computacional desenvolvido por John H. Holland e uma abordagem sobre manutenção de uma base de conhecimento, situando o leitor na linha desta dissertação.

No capítulo 3, é feita uma abordagem sobre algoritmos evolucionários, especificamente sobre o algoritmo genético através de tipos abstratos de dados.

No capítulo 4, temos a especificação e descrição de um protótipo de sistema criado a partir das técnicas estudadas nesta dissertação, bem como os detalhes de implementação e testes.

No capítulo 5, são apresentadas as conclusões finais, contribuições da dissertação e potenciais melhorias que podem ser aplicadas em trabalhos futuros.

E nos apêndices, detalhamos a análise e projeto do protótipo do sistema, fazendo uso do paradigma de Orientação a Objetos, especificamente sobre o desenvolvimento orientado a objetos, utilizando a linguagem de modelagem UML com a técnica de desenvolvimento CDI⁴.

⁴ É utilizado quando se quer organizar as tarefas, diminuir a complexidade, acrescentar detalhes ou outros recursos, com rapidez e sem comprometer o desenvolvimento do sistema (Furlan, 1998).

Capítulo 2

Fundamentação Teórica

Será apresentada a base teórica para compreensão do presente trabalho. Primeiramente iremos discutir o tema sistemas de informação e sistemas baseados em computadores, definiremos os termos e as formas de manipulação da informação. Em seguida, discutiremos o tema sistemas inteligente, definindo processos que podem atuar no comércio eletrônico, enfocando o tópico aprendizagem de máquina com o uso de técnicas de algoritmo genéticos. Logo após, discutiremos o tema computação evolutiva e manutenção de base de conhecimento.

2.1 Sistemas de Informação

Normalmente, a palavra sistema é usada para descrever elementos de um todo, conjunto ou reunião de elementos, podendo constituir-se de muitos significados. Segundo Silver (1999), um sistema é um grupo regularmente interativo e interdependente de elementos formando um todo unificado. Pressman (1995) define sistema como um conjunto de fatos, princípios ou regras classificados e arranjados ordenadamente para mostrar um plano lógico unindo as várias partes. Deste modo, um sistema é um conjunto de partes relacionadas consideradas como uma unidade onde seus componentes interagem.

Um atributo principal dos sistemas é que eles possuem qualidades e capacidades não encontradas em elementos individuais. O mesmo acontece em relação a um sistema de negócios. Os computadores, a equipe de pessoas, os processos, as faturas e os relatórios funcionando como um sistema integrado pode ter um valor maior do que as partes sozinhas.

Um dos componentes mais importantes das atividades de negócio são o gerenciamento e o processamento de informação. Um sistema de informação é constituído por um conjunto

de procedimentos, equipamentos, programas e métodos e estes conjuntos são responsáveis pelo processamento de dados tornando-os disponíveis para a gestão de tomadas de decisão (Lucas, 1990). Os dados podem ser definidos como a matéria prima que é processada e refinada para gerar informação (Tsai, et al., 2007). Informação é o que resulta do processo e manipulação dos dados.

Um sistema de informação é normalmente composto de muitas partes ou elementos chamados subsistemas. Um subsistema desempenha uma tarefa específica que deve ser compatível com os objetivos do sistema maior do qual faz parte (Pressman, 1995).

Computadores são aptos para processar vasta quantidade de informação de forma rápida e econômica, nos dias de hoje, sendo uma das figuras centrais nos modernos sistemas de processamento de informação. Silver (1999) enuncia termos, descritos na Tabela 2.1, que fornecem um ponto de vista base para a compreensão dos fundamentos de um sistema de informação, onde sistemas são vias que operam com um ambiente. Os sistemas são separados do ambiente por uma fronteira. Todos os sistemas possuem entradas (ou termo em inglês, *inputs*), um elemento do processamento de dados e resultados (ou termo em inglês, *outputs*).

Tabela 2.1 - Descrição dos principais fundamentos de um sistema de informação.

Termo	Descrição
Ambiente	Composto de pessoas, facilidades, regras, políticas e leis que circunscrevem o sistema. Sistemas existem em função de outros sistemas e do mundo exterior. Sistemas existem em função de um ambiente. Um sistema de negócios não somente deve interagir com os clientes, mas também deve lidar com muitas outras pessoas e sistemas do mesmo modo. O ambiente de um sistema pode incluir leis estaduais, federais, tendências econômicas locais, regionais ou nacionais e uma demanda de intercâmbio e suprimento monetário. Para analisar sistemas devemos considerar o contexto em que eles funcionam.
Fronteira	Fronteira pode ser definida como o limite de demarcação entre o sistema e o ambiente. Ela ajuda a distinguir entre os elementos que formam o sistema e o mundo exterior com o qual interage. Algumas vezes, as fronteiras estão claramente definidas, enquanto outros sistemas podem ter fronteiras confusas. Portanto, um estudo minucioso deve ser feito na tentativa de delimitar as linhas de demarcação dos sistemas.
Entrada	Todo sistema recebe algum tipo de informação ou dados que é processado, gerando resultados. Entradas são aqueles itens que adentram as fronteiras do sistema a partir do ambiente e são manipulados pelo sistema.

Processamento	O processamento é uma conversão de entradas em resultados. As atividades do processamento de dados incluem classificação, pesquisa, unificação, sumarização, cálculo e operações semelhantes e possibilitando uma organização alcançar seus objetivos.
Resultado	Resultado é o produto do processamento. Os resultados em um pequeno negócio podem consistir no lucro de seu proprietário ou dos acionistas.

2.1.1 Sistemas de Informação Baseados em Computadores

Segundo Pressman (1995), sistemas de informação baseados em computadores (SIBC) utiliza *hardware*, *software*, documentação, bancos de dados, pessoas e procedimentos para coletar, armazenar e transformar dados em informações, como ilustrado na Figura 2.1 e descritos na Tabela 2.2.

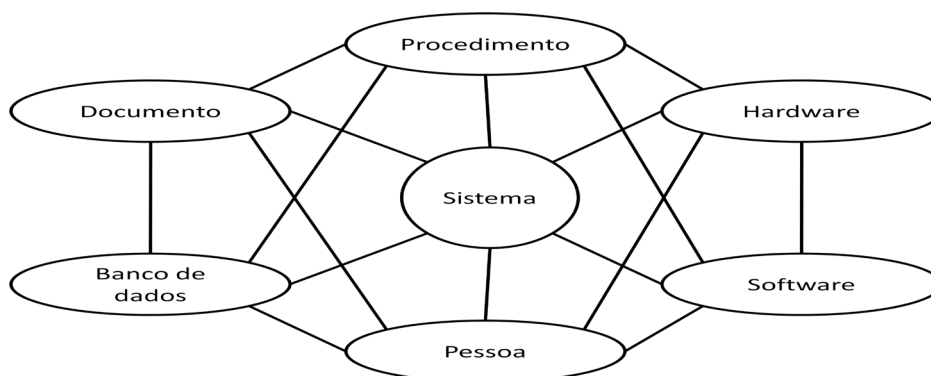


Figura 2.1 - Elementos de um SIBC.

Uma característica complicadora dos sistemas baseados em computador é que os elementos que formam um sistema podem representar um macro elemento de um sistema ainda maior. Podendo torna-se um fator negativo e sendo necessário definir os elementos para um sistema específico baseado em computador no contexto da hierarquia global dos sistemas.

Tabela 2.2 - Descrição dos elementos de um SIBC.

Elemento	Descrição
<i>Hardware</i>	Dispositivos eletrônicos que fornecem capacidade ao computador, e dispositivos eletromecânicos que oferecem funções ao mundo externo.
<i>Software</i>	Programas de computador, estruturas de dados e documentação correlata que servem para efetivar o método, processo ou controle lógico necessário.

Documentação	Manuais, formulários e outras informações descritivas que retratam o uso ou operação do sistema.
Banco de dados	Grande e organizada coleção de dados a que se tem acesso pelo software e faz parte integrante da função do sistema.
Pessoa	Usuários e operadores de hardware e software.
Procedimento	Passos que definem o uso específico de cada elemento do sistema ou o contexto processual em que o sistema reside.

Sistemas de informação podem ser classificados em dois grupos principais: sistemas de apoio às operações e sistemas de apoio à gestão. Segundo Pressman (1995), sistemas de apoio às operações são redes de procedimentos rotineiros que servem para o processamento de informações recorrentes com grandes volumes de dados. Como por exemplo, sistemas de informações transacionais (do inglês *Transaction Processing Systems*) que são sistemas que processam transações com precisão e segurança.

Sistemas de apoio à gestão existem especificamente para auxiliar processos decisórios (Vidal, 1996) e não são orientados para o processamento de transações rotineiras. Como exemplo, temos os sistemas de informações gerenciais que são sistemas que apóiam o planejamento, fazem acompanhamento e controle das operações na organização. Compreendem sistemas de estimação de vendas ou compras, análises financeiras e os sistemas voltados à elaboração e controle das operações. Os sistemas de apoio à gestão que são mais difíceis de serem construídos e avaliados, porque suportam decisões nos níveis superiores da hierarquia das empresas (Pressman, 1995).

Sistemas de informações gerenciais devem ser vistos como uma ferramenta administrativa e representa um auxílio importante para os executivos das empresas. Estes sistemas são representados pelo conjunto de subsistemas, visualizados de forma integrada e capaz de gerar informações necessárias ao processo decisório (Silver, et al., 1999). Algumas vantagens que um sistema de informação gerencial pode trazer para as empresas:

- Diminuição nos custos das operações;
- Progresso no acesso às informações (agilidade na obtenção de informações);
- Melhoria na prestação de serviços aos clientes (internos e externos);
- Tomada de decisões melhores e mais ágeis;
- Níveis hierárquicos reduzidos.

2.1.2 Sistemas de Apoio à Decisão

A necessidade dos sistemas de apoio à decisão (SAD), ou DSS (do termo em inglês, *Decision Support Systems*), surgiu na década de 70 (Lucas, 1990), em decorrência de diversos fatores, como, por exemplo, os seguintes:

- Competição cada vez maior entre as organizações;
- Necessidade de informações rápidas para auxiliar no processo de tomada de decisão;
- Disponibilidade de tecnologia para armazenar e buscar rapidamente as informações;
- Possibilidade de armazenar o conhecimento e as experiências de especialistas em bases de conhecimentos;
- Necessidade de a informática apoiar o processo de planejamento estratégico empresarial.

Estes itens foram de fundamental importância no processo de desenvolvimento de sistemas capazes de fornecer informações e auxiliar no processo de tomada de decisão.

Dentre as definições consultadas, serão ilustradas três:

- “SAD é um sistema de informação que apóia qualquer processo de tomada de decisão em áreas de planejamento estratégico, controle gerencial e controle operacional” (Sprague, et al., 1991);
- “SAD é um sistema baseado em computador que auxilia o processo de tomada de decisão utilizando dados e modelos para resolver problemas não estruturados” (Lucas, 1990);
- “SAD é uma estratégia de implementação que torna o computador útil ao gerente” (Rockart, et al., 1986).

Sistema de apoio à decisão pode ser definido como um conjunto de procedimentos para o processamento de dados e seu julgamento, que apóiam o usuário em sua tomada de decisão (Sprague, et al., 1991). São sistemas mais complexos que permitem total acesso à base de dados corporativos, modelagem de problemas e simulações, auxiliando um executivo em todas as fases de tomada de decisão, principalmente, nas etapas de desenvolvimento,

comparação e classificação dos riscos, além de fornecer subsídios para a escolha de uma boa alternativa.

É através da interação do usuário com o ambiente de apoio à decisão que o processo de tomada de decisão se realiza. Esse ambiente, representado na Figura 2.2, é constituído por:

- Bancos de Dados (BD): São formados por dados internos e externos à organização, por conhecimentos e experiências de especialistas e por informações históricas acerca das decisões tomadas;
- Sistema Gerenciador de Banco de Dados (SGDB): Após os dados estarem instalados no BD, o SGDB deve possibilitar o acesso às informações e a sua atualização, garantindo a segurança e a integridade do BD;
- Ferramentas de Apoio à Decisão (FAD): São *softwares* que auxiliam na simulação de situações e na representação gráfica das informações;
- Ambiente Aplicativo (AA): São sistemas aplicativos ou funções acrescentadas aos sistemas existentes que fazem análise de alternativas e fornecem soluções de problemas;
- Ambiente Operacional (AO): É composto por *hardwares* e *softwares* que permitem que todos os componentes do ambiente sejam integrados.

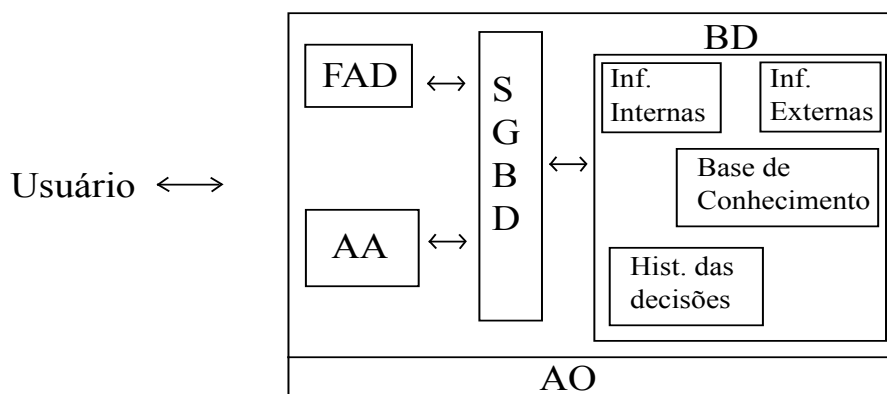


Figura 2.2 - Ambiente de apoio à decisão.

Tal ambiente, para ser eficiente e eficaz, deve ser simples, robusto, fácil de operar, adaptativo, completo para as questões em estudo e amigável.

É através da escolha de alternativas existentes que ocorre a tomada de decisão seguindo determinados passos anteriormente estabelecidos e resultando na resolução de um problema de modo satisfatório ou não, envolvendo análise, identificação, desenvolvimento, comparação, classificação dos riscos, execução, avaliação e escolha da alternativa ideal.

Principais características de um sistema de apoio à decisão:

- Utilizados para resolução de problemas mais complexos e menos estruturados que os demais;
- Tentam combinar modelos ou técnicas analíticas com as funções tradicionais de processamento de dados, como acesso e a recuperação de informações;
- Devem ser interativos, fáceis de usar e ter interface extremamente amigável;
- Os sistemas de apoio à decisão devem acompanhar as tendências, sendo mais flexíveis e adaptáveis a mudanças;
- Devem fornecer subsídios para um rápido encaminhamento e implementação dos resultados obtidos a partir da tomada de decisão;
- Suporte à decisão é necessário em todos os níveis de gerenciamento da empresa.

2.1.3 Sistemas de IA

O termo inteligência artificial nasceu em 1956 no encontro de Dartmouth. Dentre os presentes a este encontro incluíam-se Allen Newell, Herbert Simon, Marvin Minsky, Oliver Selfridge e John McCarthy. No final dos anos 50 e início dos anos 60, os cientistas Newell, Simon, e J. C. Shaw introduziram o processamento simbólico. Ao invés de construir sistemas baseados em números, eles tentaram construir sistemas que manipulassem símbolos (Minsky, 1969). Essa investida teve grande influência e foi importante para muitos trabalhos posteriores. A partir daí, os variados estudos e pensamento em IA têm avaliado formas de constituir comportamentos inteligentes nas máquinas.

Um sistema de inteligência artificial terá que ser apto de realizar aquisição, representação e manipulação de conhecimento (Pal, et al., 2004). A esta manipulação também compreende a competência de deduzir (ou inferir) novos conhecimentos, ou seja, novas relações sobre fatos e conceitos a partir do conhecimento prévio (Russell, et al., 2004). Portanto, problemas complexos podem ser solucionados por um sistema IA que, para tal, utiliza métodos de representação e manipulação de conhecimento.

As pesquisas em IA demonstram que o conhecimento declarativo (fatos e regras) pode ser representado separadamente do conhecimento procedimental (instruções de decisão) ocasionando mudanças na abordagem do problema quanto nas técnicas para produzir sistemas inteligentes. No desenvolvimento de sistemas inteligentes é importante a construção de um

processo para obtenção e codificação de regras e fatos que sejam suficientes para um determinado domínio do problema. Como também, se faz necessário, a construção de uma máquina de inferência onde irá ocorrer a dedução e obtenção do conhecimento declarativo, onde novos fatos ou regras poderão ser gerados a partir do conhecimento armazenado. A Figura 2.3 retrata estas questões, mostrando a relação entre os componentes de um sistema clássico de IA (Schutzer, 1987).

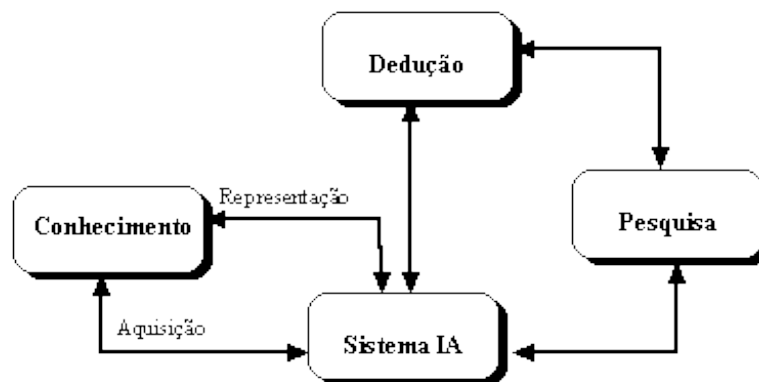


Figura 2.3 - Uma visão conceitual dos sistemas de Inteligência Artificial.

As pesquisas em IA estão relacionadas com áreas de aplicação que envolve o raciocínio humano (Savory, 1988). Estas áreas de aplicação podem ser de compreensão e tradução de linguagem natural, compreensão e geração de voz, análise de imagem e cena em tempo real, programação automática, sistemas especialistas ou sistemas baseados em conhecimento, sistemas evolutivos e sistemas de aprendizagem.

Segundo Russel (2004), os sistemas de inteligência artificial podem está classificados em um dos três grupos principais na computação: sistemas de inteligência computacional cognitiva, sistemas de inteligência computacional neural e sistemas de inteligência computacional evolucionária. A inteligência computacional cognitiva procura representar, através de símbolos, as regras formais do conhecimento humano (Russell, et al., 2004). Exemplos de uso de inteligência computacional cognitiva são sistemas baseado em regras e sistemas baseados em casos⁵. Na inteligência computacional neural e evolucionária procuram representar os modelos formais do comportamento humano (Kovacs, 2002) (Nascimento, 2000), podendo tais formalismos ser manipulados por um computador. Um exemplo de uso

⁵ Raciocínio baseado em casos (RBC, ou do inglês *Case-Based Reasoning* - CBR) é uma fundamentação teórica relacionada às técnicas que representam o aprendizado com experiência baseado no raciocínio (Wangenheim, et al., 2003). Obras recentes como (Bichindaritz, et al., 2007), (Montani, 2006) tem aplicado RBC na área médica e em Tsai (2007), utiliza sistemas RBC para inferir principais parâmetros de um processo produtivo aplicado a fabricação de placa de circuito impresso.

inteligência computacional neural é o modelo de mapas auto-organizáveis⁶ que são utilizados em problemas de classificação de padrões e baseia-se em aprendizagem competitiva e não supervisionada (Braga, et al., 2000). A inteligência evolucionária é baseada em uma metáfora apresentada por Holland (Qi, et al., 1994) entre o processo de evolução das espécies e o processo de resolução de problemas, os algoritmos evolucionários trabalham com operadores de seleção, mutação e recombinação em uma população de soluções para o problema de modo concorrente (Raymer, et al., 2000). Um exemplo de uso de inteligência computacional evolucionária é o estudo de modelos de algoritmo genético que é área de estudo deste trabalho e que será tratado a seguir.

2.1.4 Sistemas de Recomendação

O aparecimento do comércio eletrônico proporcionou um ambiente ideal para a aplicação de sistemas de recomendação (SR). O objetivo dos sistemas de recomendação é personalizar um determinado usuário ou grupo, possibilitando aos estabelecimentos relacionar os produtos ao perfil de cada grupo de clientes (Wei, et al., 2002). A arquitetura de um sistema de recomendação está ilustrada na Figura 2.4, abaixo.

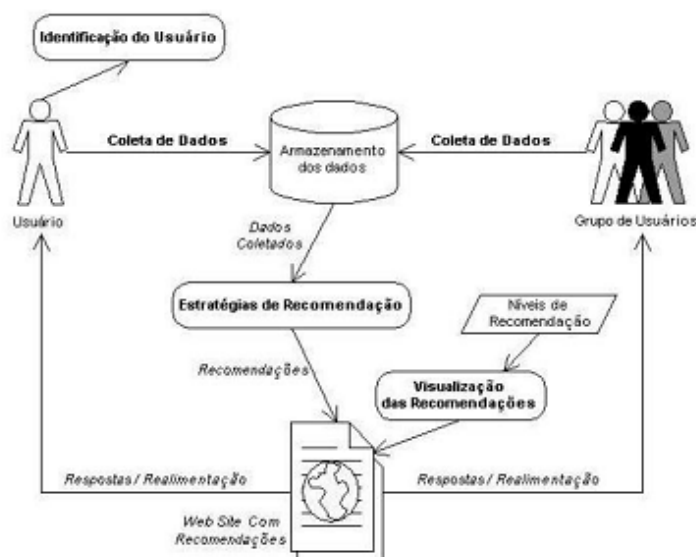


Figura 2.4 - Estrutura do Sistema de Recomendação (Barcellos, et al., 2007).

⁶ Do inglês *Self-Organizing Map* (SOM), foi inicialmente desenvolvido por Kohonen a partir de 1982 através do estudo do cérebro humano (entradas sensoriais como a visual, a auditiva, a motora, entre outras), onde as entradas sensoriais são mapeadas para áreas correspondentes do córtex cerebral de forma ordenada (Haykin, 2001). Sua arquitetura consiste em neurônios sobre uma grade unidimensional ou bidimensional, com os vários estímulos de entrada, os neurônios se tornam sintonizados no decorrer do processo de aprendizagem.

Na visão de Schafer (1999), os sistemas de recomendação são divididos em: identificação do usuário, coleta de informações, estratégias de recomendação e visualização das recomendações. Na estrutura da Figura 2.4, a identificação do usuário é opcional, porém, uma coleção de dados reais e confiáveis se faz necessário para uma boa estratégia do SR (Wei, et al., 2002). Segundo (Telles, et al., 2006), existem duas técnicas usadas de forma individual ou juntas para identificar o perfil do usuário. São elas:

- Extração implícita: As informações são coletadas de forma transparente através do uso do sistema;
- Extração explícita: O próprio usuário deve informar seus interesses que podem ser extraídos através de questionário.

O sistema deve gerar as recomendações depois da identificação e coleta de informações do usuário podendo usar para tal duas técnicas (Barcellos, et al., 2007):

- Filtragem colaborativa: Emprega a similaridade entre os usuários possibilitando a ajuda a outras pessoas que, por sua vez, são agrupadas por terem perfis semelhantes;
- Filtragem baseada em conteúdo: Utiliza a similaridade de produtos ou serviços adquiridos anteriormente para a recomendação de itens próximos aos adquiridos.

2.2 Computação Evolutiva

A computação evolutiva soluciona problemas tomando como base o processo biológico de seleção natural e propagação genética de Darwin, baseando-se em uma população de indivíduos⁷ em um dado ambiente em que os mais aptos sobrevivem e conseqüentemente reproduzem-se. As características destes indivíduos, no Algoritmo Evolutivo (AE), são codificadas no cromossomo. Portanto, de modo geral, nos algoritmos evolutivos os indivíduos competem entre si, sendo sua sobrevivência e reprodução determinadas pelo meio através da seleção dos mais aptos (Qi, et al., 1994). É através de funções de custo que a medida de aptidão de cada indivíduo pode ser determinada diretamente ou indiretamente (Zuben, 2000).

⁷ São possíveis soluções do problema que combinado com outros gerão filhos (novos indivíduos).

Algoritmos evolucionários trabalham sobre uma população P de indivíduos, que evoluem através da ação dos operadores genéticos, com o intuito de gerar outra população P' tal que, os indivíduos P' sejam mais adaptados do que os indivíduos de P . Na Figura 2.5 temos um esquema geral de um AE.

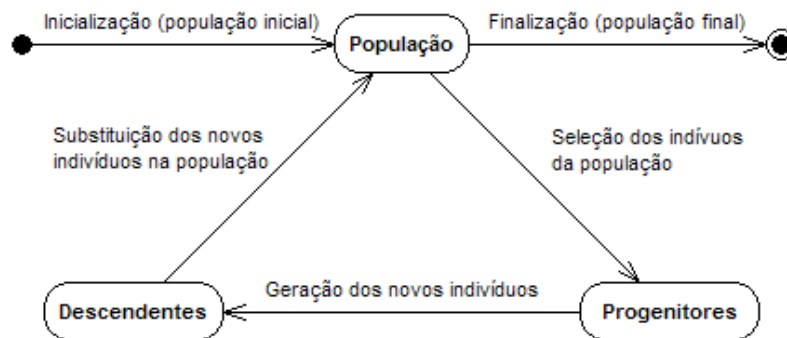


Figura 2.5 - Esquema geral dos algoritmos evolutivos.

No esquema da Figura 2.5 temos a inicialização da população com a entrada de um conjunto de indivíduos aleatórios e cada indivíduo é avaliado. Enquanto a população atual não satisfazer uma condição de parada, uma seleção de indivíduos será feita. Os indivíduos selecionados serão combinados em pares gerando indivíduos descendentes. Esses novos indivíduos serão avaliados e substituídos na população.

O AE possui, como importante aspecto, uma diversidade genotípica garantida pela formação de novos indivíduos que se utilizam do genótipo dos seus progenitores, através de operadores genéticos não determinísticos (como cruzamento ou mutação), objetivando a obtenção do genótipo do novo indivíduo (também chamado de filho) (Rudolph, 1994). Na Figura 2.6 é ilustrado pseudocódigo de um AE.

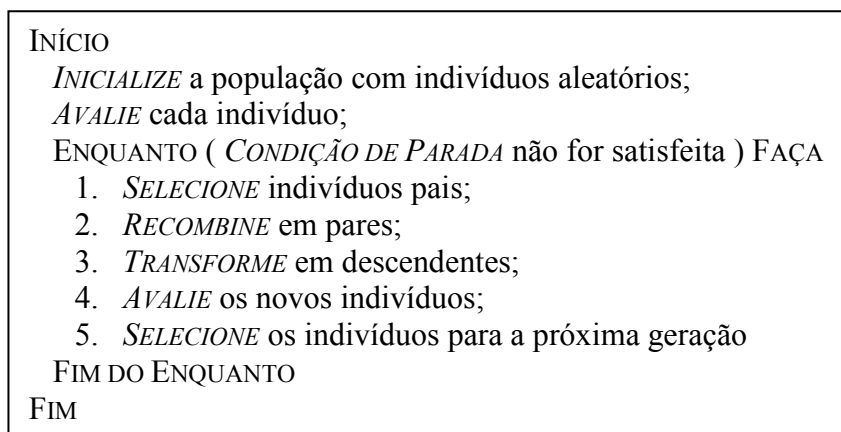


Figura 2.6 - Pseudocódigo para algoritmo evolutivo.

O processo de exploração do espaço de busca é formado por parâmetros codificados nos cromossomos e a adaptabilidade do indivíduo. Outra forma de garantir a diversidade genotípica é através dos mecanismos de seleção, que podem escolher indivíduos de forma não determinística (Qi, et al., 1994). A escolha de parâmetros é um fator importante no processo evolutivo. Eles influenciam na adaptação do indivíduo, podendo tornar o processo evolutivo lento (Rockett, 1998). Na Figura 2.7 temos uma representação do processo de geração do espaço genótipo G , fenótipo F e um ambiente I . Segundo Zuben (Zuben, 2000), é no espaço genótipo (ou código genético) que o AE realiza a busca e é no espaço fenotípico que a pressão seletiva incide.

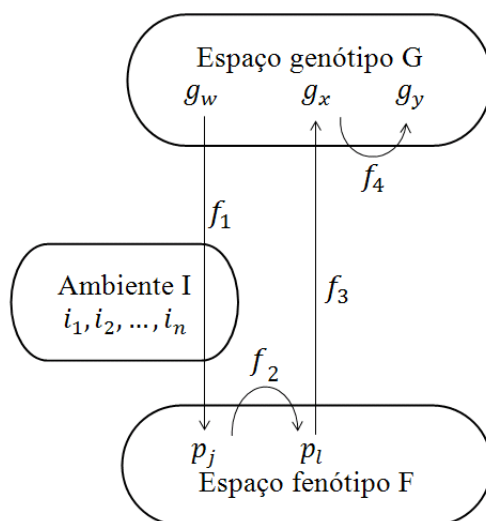


Figura 2.7 - Evolução de uma população durante uma geração conforme (Zuben, 2000).

A Figura 2.7 ilustra o processo de evolução de uma população P , onde G são os estados genótipos de um conjunto de entradas providos de um ambiente I e F o espaço de fenótipo deste ambiente. Para cada geração do processo de evolução de P irá ocorrer os seguintes mapeamentos: $f_1: I \times G \rightarrow F$ que determina as características dos indivíduos a partir do seu genótipo g_k ; $f_2: F \rightarrow F$ que determina a seleção e migração de indivíduos sobre o espaço fenótipo F ; $f_3: F \rightarrow G$ que determina os efeitos dos processos de f_2 ; E $f_4: G \rightarrow G$ que determina a recombinação e mutação de indivíduos e abrange todas as alterações genéticas.

Embora a computação evolutiva seja baseada em métodos estocásticos (principalmente nos operadores de seleção, cruzamento e mutação, onde o fator probabilístico está presente) as operações da computação evolutiva podem ser obtidas com um direcionamento ou uma heurística.

Historicamente, dentro da computação evolutiva, diferentes modelos de soluções foram associadas, podendo ser classificadas em quatro áreas: algoritmo genético, estratégia evolutiva, programação evolutiva e programação genética. Os algoritmos genéticos têm o objetivo de formalizar matematicamente os processos de adaptação e criação de aplicações de computador empregando, principalmente, os operadores de cruzamento e mutação nas entradas representadas por uma cadeia de símbolos (Rudolph, 1994). A estratégia evolutiva é utilizada principalmente para resolver problemas de aperfeiçoamento de parâmetros e funções de um programa existente empregando apenas o operador de mutação (Arruda, et al., 2009). Na programação evolutiva, semelhante à estratégia evolutiva no que diz respeito à aplicação de problemas de otimização e uso do operador de mutação, procura representar os modelos formais em máquinas de estados finitos (Qi, et al., 1994). Na programação genética os indivíduos são módulos ou programas representados em árvores, onde cada programa será candidato a solução do problema (Maia, et al., 2001).

2.3 Algoritmos Genéticos

Em meados da década de setenta surgiu os algoritmos genéticos através do método computacional para abordar fenômenos gerados por sistemas adaptativos complexos (Qi, et al., 1994). Os fenômenos gerados por sistemas adaptativos complexos são aqueles cujo resultado depende das interações não lineares entre vários agentes adaptativos. Durante as pesquisas de Holland foi constatado que existia uma nítida semelhança entre os fenômenos que estudava e o processo de evolução das espécies, pois assim como a interação entre os agentes adaptativos determinava o resultado dos fenômenos investigados por ele, a interação entre os fatores ambientais determinava a próxima população de uma determinada região. Essa constatação levou a desenvolver um método computacional que simulasse o processo de evolução das espécies que passou a se chamar de algoritmo genético (Rockett, 1998). Na Tabela 2.3 será apresentada a correspondência entre alguns vocábulos da linguagem usada na biologia e na computação.

Tabela 2.3 - Descrição dos principais vocábulos biológicos para o computacional.

Termo da biologia	Descrição computacional
Cromossomo	Indivíduo de uma população.
Gene	Caractere ou atributo de um cromossomo (indivíduo).
Genótipo	Vetor de caracteres que representa o indivíduo.
Fenótipo	Interpretação do vetor de caracteres.

De forma simplificada, os passos básicos do processo de execução de um algoritmo genético trabalham sobre uma população P de indivíduos, que evoluem através da ação dos operadores genéticos (cruzamento e mutação), com o intuito de gerar outra população P' a partir do material genético da população P . Tal que, os indivíduos P' sejam mais adaptados do que os indivíduos de P .

Os algoritmos genéticos têm sido fortemente aplicados em problemas de otimização e tomada de decisão, apesar de não ter sido este o propósito original que levou ao seu desenvolvimento. O trabalho de Holland é considerado um dos principais precursores do AE (Qi, et al., 1994), pois teve a finalidade de compreender os princípios dos sistemas adaptativos complexos⁸. Na próxima sessão, será descrito um resumo do algoritmo de Holland que está relacionado com o esboço de modelos de algoritmo genético que é área de estudo deste trabalho.

2.4 Algoritmo Genético de Holland

Na sua pesquisa, Holland tentava desenvolver um método computacional que se prestasse para abordar fenômenos gerados por sistemas adaptativos. No decorrer do seu estudo, percebeu que existia uma nítida semelhança entre os fenômenos que estudava e o processo de evolução das espécies, pois assim como a interação entre os agentes adaptativos determinava o resultado dos fenômenos investigados por ele, a interação entre os fatores ambientais determinava a próxima população de uma determinada região. Essa constatação levou Holland a desenvolver um método computacional que simulasse o processo de evolução das

⁸ Os fenômenos gerados por sistemas adaptativos complexos são aqueles cujo resultado depende das interações não lineares entre vários agentes adaptativos.

espécies que posteriormente passou a se chamar de algoritmos genéticos. A Figura 2.8 ilustra os principais estados de um algoritmo genético.

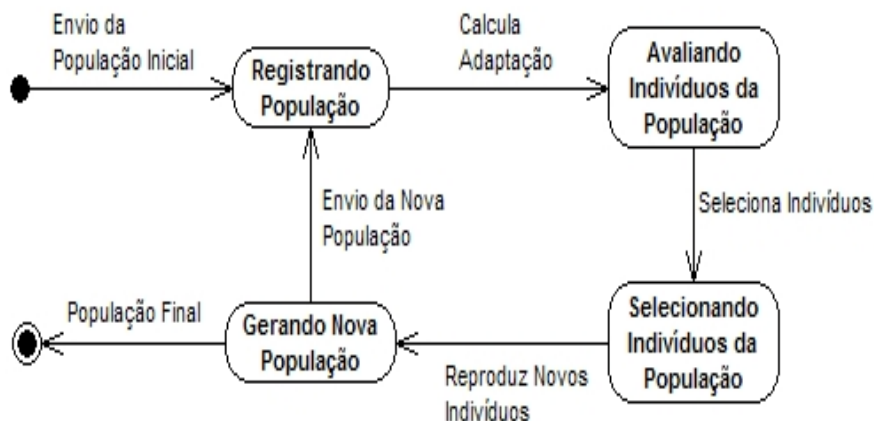


Figura 2.8 - Ilustração de um AG através de diagrama de estado (Qi, et al., 1994).

Para Holland, uma população P_j é um conjunto de cromossomos no qual cada cromossomo é codificado como um vetor de caracteres, ou cadeia de atributos, de tamanho fixo, formado pela concatenação dos caracteres “0” e “1” do alfabeto binário $\{0,1\}$ (representação binária). O algoritmo genético parte de uma população inicial P_0 e através de um mecanismo de seleção dará preferência aos indivíduos mais adaptados ao ambiente (que apresenta maior f) para aplicar os operadores genéticos de cruzamento (ou o termo em inglês *crossover*) e mutação. Caracteriza-se um processo evolutivo, quando o f do melhor indivíduo a cada geração apresente uma tendência de crescimento continuado, sendo este indivíduo, por conseguinte, capaz de representar uma boa solução do problema proposto, a qual pode não ser a solução ótima.

Holland propôs duas versões de algoritmos genéticos, chamados de R_1 e R_d . No primeiro, em cada iteração do algoritmo será gerado um novo indivíduo que substituirá um dos indivíduos da população atual. Ao passo que no segundo algoritmo, em cada iteração do algoritmo serão gerados n descendentes, onde n é o tamanho da população atual, os quais irão substituir todos os indivíduos da população atual. Em seguida Holland propôs o algoritmo R que simula o comportamento desses dois algoritmos dependendo dos valores de dois parâmetros: o número de cromossomos selecionados para gerar descendentes e o número de descendentes que devem ser gerados. A Figura 2.9 ilustra um pseudocódigo do algoritmo R de Holland.

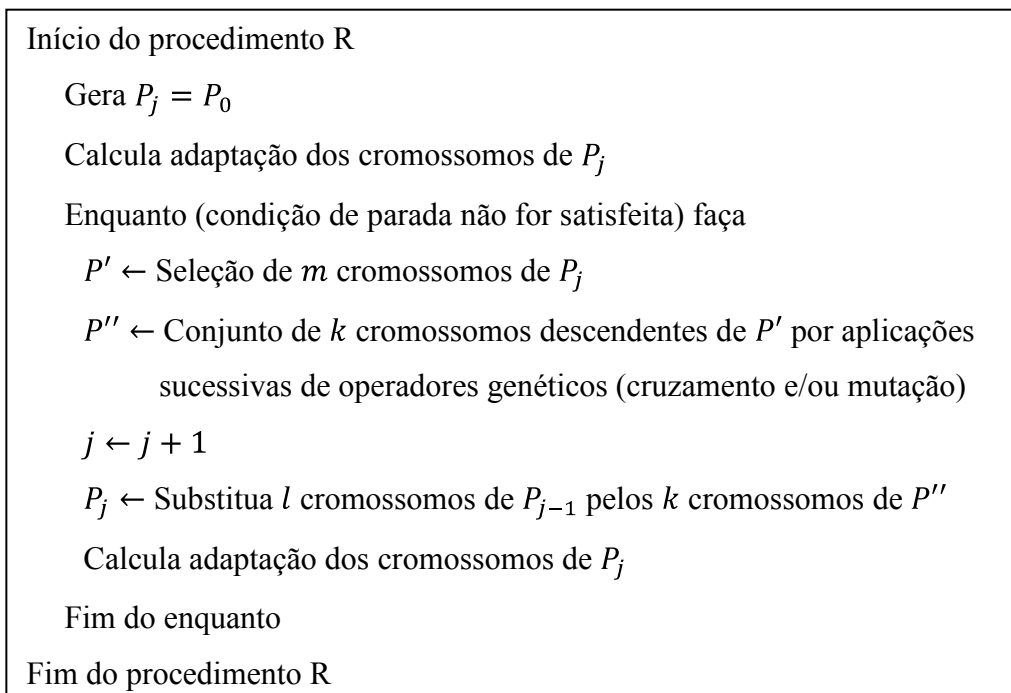


Figura 2.9 - Pseudocódigo do algoritmo R (Vieira, 2003).

Segundo o pseudocódigo da Figura 2.9, um algoritmo genético tem início com a geração da população inicial P_0 composta por n cromossomos, que podem ser fornecidos pelo usuário, gerados aleatoriamente ou construídos por um algoritmo baseado no conhecimento existente sobre o problema a ser solucionado. Em seguida, é calculada a adaptação dos cromossomos da população, para verificar a qualidade destes cromossomos para o problema investigado. Se o critério de parada definido para o problema não for satisfeito, então um novo ciclo (iteração, até a condição for satisfeita) dá início. Nesse ciclo, é selecionado m cromossomos para gerar descendentes, gera os descendentes, incrementa o contador de iterações j , nascimento de alguns dos cromossomos gerados e morte dos cromossomos existentes na população considerados de baixa qualidade para o problema analisado tem início. Os critérios de parada usados pelos algoritmos genéticos são encontrar um cromossomo com adaptação x ou atingir a j -ésima iteração.

As variáveis j, m, k, l, P_j, P' e P'' do pseudocódigo do algoritmo R representam respectivamente: o número de gerações (e conseqüentemente o número de iterações), o número de cromossomos selecionados para gerar descendentes, o número de descendentes que devem ser gerados, o número de cromossomos que serão substituídos pelos k descendentes, a população na geração j , o reservatório dos cromossomos que irão gerar novos

cromossomos e o reservatório dos novos cromossomos gerados a partir da aplicação do conjunto de operadores genéticos que foram aplicados aos cromossomos do reservatório P' , os quais farão parte da próxima população. O número de gerações é incrementado em cada execução do laço *Enquanto..faça* para simular a contagem do tempo genealógico.

Cruzamento de um ponto de corte, inversão e mutação foram os operadores genéticos trabalhados por Holland em seu algoritmo genético. O operador de cruzamento de um ponto de corte tem por objetivo combinar a cadeia de dois cromossomos da seguinte maneira:

- Ter dois cromossomos: $a = \{a_1, a_2, \dots, a_n\}$ e $b = \{b_1, b_2, \dots, b_n\}$ de P_j , com $n \in \mathbb{N} | n \geq 1$;
- Selecionar, de maneira aleatória, um número: $x \in \mathbb{N} | x \geq 1 \wedge x \leq n$;
- Construir dois novos cromossomos: $c' = \{a_1, \dots, a_{x-1}, a_x, b_{x+1}, \dots, b_n\}$ e $c'' = \{b_1, \dots, b_{x-1}, b_x, a_{x+1}, \dots, a_n\}$.

A Figura 2.10 ilustra um cruzamento, considerando que $x = 3$, $n = 7$, $a = \{0,1,0,0,1,1,0\}$ e $b = \{1,1,1,0,0,0,1\}$, resultando nos novos cromossomos $c' = \{0,1,0,0,0,0,1\}$ e $c'' = \{1,1,1,0,1,1,0\}$.

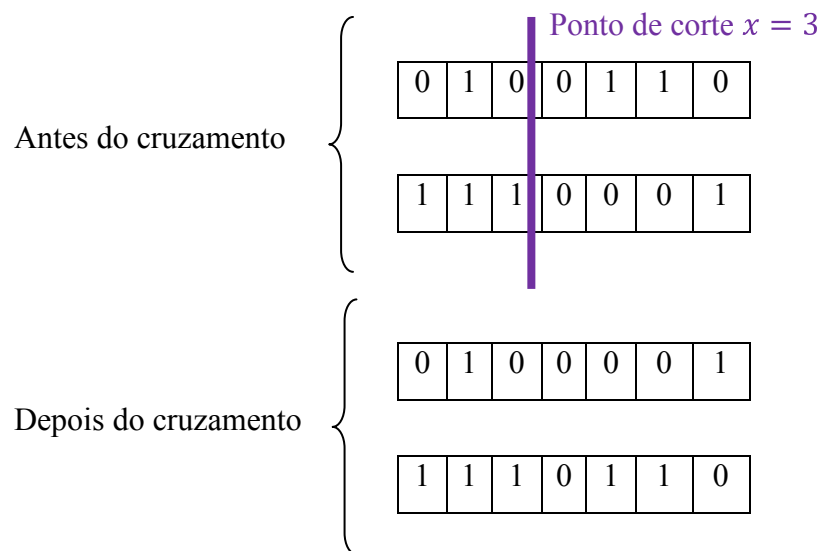


Figura 2.10 - Exemplo do operador de cruzamento de Holland (Vieira, 2003).

O operador de inversão tem por objetivo alterar a ordem dos caracteres de uma subcadeia de um cromossomo da seguinte maneira:

- Ter um cromossomo: $a = \{a_1, a_2, \dots, a_n\}$ de P_j , com $n \in \mathbb{N} | n \geq 1$;

- Selecionar, de maneira aleatória, um intervalo:

$$[x, y] \in \mathbb{N} \mid x \geq 1 \wedge x < n \wedge y > 1 \wedge y \leq n \wedge x < y$$

- Construir um novo cromossomo:

$$c = \{a_1, \dots, a_{x-1}, a_y, a_{y-1}, \dots, a_{x+1}, a_x, a_{y+1}, \dots, a_n\}$$

A Figura 2.11 ilustra uma inversão, considerando que $x = 2, y = 5, n = 7$ e $a = \{1,1,1,0,0,0,1\}$, resultando no novo cromossomo $c = \{1,0,0,1,1,0,1\}$.

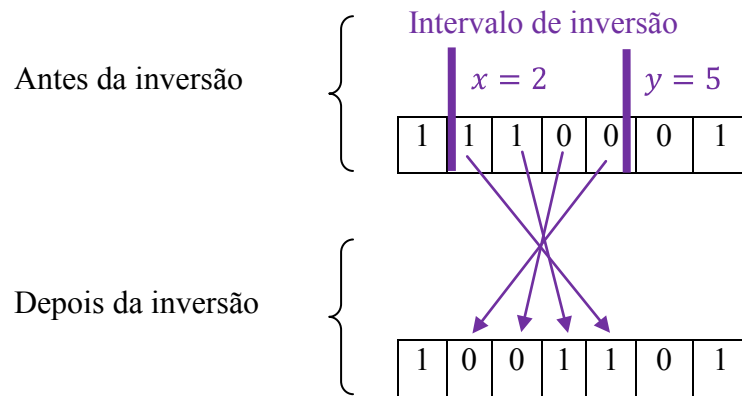


Figura 2.11 - Exemplo do operador de inversão de Holland (Vieira, 2003).

O operador de mutação tem por objetivo alterar a cadeia de um cromossomo da seguinte maneira:

- Ter um cromossomo: $a = \{a_1, a_2, \dots, a_n\}$ de P_j , com $n \in \mathbb{N} \mid n \geq 1$;
- Selecionar, de maneira aleatória, k atributos x_1, x_2, \dots, x_k com $k \in \mathbb{N} \mid k \geq 1 \wedge k \leq n$ e x_k com $x_k \in \mathbb{N} \mid x_k \geq 1 \wedge x_k \leq n$
- Construir um novo cromossomo:

$$c = \{a_1, \dots, a_{x_1-1}, a_{x_1}, a_{x_1+1}, \dots, a_{x_k-1}, a_{x_k}, a_{x_k+1}, \dots, a_n\}$$

Os atributos x_i da cadeia do novo cromossomo, após a operação de mutação, representam o complemento do atributo que ocupava essa mesma posição no cromossomo $a = \{a_1, a_2, \dots, a_n\}$ de P_j . A Figura 2.12 ilustra uma mutação, considerando que $k = 2, x_1 = 2, x_2 = 5, n = 7$ e $a = \{1,1,1,0,0,0,1\}$, resultando no novo cromossomo $c = \{1,0,1,0,1,0,1\}$.

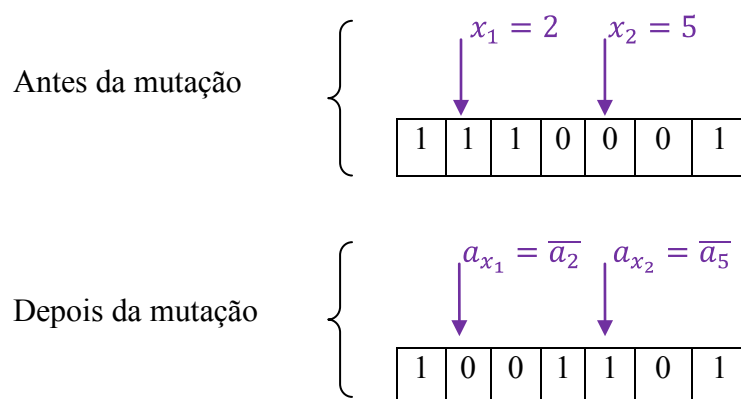


Figura 2.12 - Exemplo do operador de mutação de Holland (Vieira, 2003).

O uso do algoritmo genético de Holland na solução de diversos problemas ensejou algumas questões como (Vieira, et al., 1999):

- Quais problemas podem ser solucionados por algoritmo genético?
- O tempo de convergência⁹ de um algoritmo genético pode ser usado como medida de sua eficiência?
- A distância entre o melhor e o pior resultado encontrado pelo algoritmo genético pode ser usado como um critério para determinar a qualidade desse resultado?
- O algoritmo genético deve trabalhar com uma população de tamanho fixo ou não?
- O quanto à diversidade da população influencia na qualidade do resultado encontrado pelo algoritmo genético?

Vários experimentos foram desenvolvidos com o intuito de responder tais conjecturas, mas os resultados obtidos foram, na maioria das vezes, contraditórios. Isso porque cada experimento dependia: da forma como o algoritmo foi implementado; do problema resolvido por tal experimento; da população inicial adotada; da plataforma em que o algoritmo foi implementado, entre outros fatores. O insucesso da tentativa de responder as conjecturas listadas acima a partir de conhecimentos experimentais serviu para alertar os pesquisadores que a resposta a essas conjecturas só poderá ser obtida a partir de conhecimentos gerados por

⁹ Número de populações geradas pelo algoritmo genético até que um resultado satisfatório para o problema seja encontrado.

uma teoria que trate dos algoritmos genéticos de maneira geral e que seja descomprometida de qualquer apelo semântica e contextual.

Outros estudantes de pós-graduação e alguns pesquisadores, como em Rudolph (1994), Qi (1994), Fonseca (1998), Raymer (2000) e Maia (2001), passaram a desenvolver variações do algoritmo genético de Holland na solução de problemas de otimização de funções, otimização combinatorial, sistemas de classificação e máquinas de aprendizagem. O sucesso obtido por essas variações na solução dos seus respectivos problemas, fez aumentar o interesse da comunidade científica pelo algoritmo genético.

2.5 Manutenção da Base de Conhecimento

Segundo Raymer (2000), na literatura existem muitas aplicações relacionadas a sistemas de apoio à decisão na resolução de problemas, em muitas situações existe uma relação direta entre o número de ocorrências a ser armazenados na base de conhecimento e a eficiência de recuperação (Wangenheim, et al., 2003). Ou seja, quanto maior for a base de dados, maior será o espaço do problema a ser diagnosticado. Assim, a degradação do desempenho do sistema poderá está relacionada ao crescimento do número de ocorrências de forma inaceitável.

De acordo com Wangenheim (2003), uma das tarefas mais importantes na manutenção de base de conhecimento está relacionada a remoção das ocorrências redundantes ou daquelas menos úteis, para que se possa atingir um nível de erro aceitável. A principal atividade de manutenção de sistemas de apoio à decisão está relacionado ao desenvolvimento de algumas medidas de competência, a qual propoporciona a confiabilidade dos problemas solucionados pelo sistema. Muitas propriedades poderão ser úteis, tais como: tamanho, distribuição, densidade na base de dados, cobertura de ocorrências individuais, semelhança e conhecimento de adaptação de um determinado sistema (Pal, et al., 2004).

Outra forma de se trabalhar com a manutenção da base de dados, está relacionada à possível existência de ocorrências contraditórias, devido a mudanças em conhecimento de domínio ou ambientes específicos para uma determinada tarefa. Além disso, se duas ocorrências são consideradas equivalentes (com valores de característica idênticos) ou se uma ocorrência incluir outra tendo mais critérios de característica, um processo de manutenção poderá ser exigido para remover os dados redundantes.

A manutenção de base de conhecimento pode ser resumida da seguinte forma (Wangenheim, et al., 2003): dado um conjunto de conhecimento $C_i = \{a_1, a_2, \dots, a_k\}$, onde cada conhecimento C_i é composto por k atributos, armazenados em uma base de dados que representa a base de conhecimento $BC = \{C_1, C_2, \dots, C_j\}$, seja a função de medida similaridade $sim(C_x, C_y): BC \times BC \rightarrow [0,1] \in \mathbb{R}$ expressando o julgamento de semelhança entre dois conhecimento, satisfazendo o requisito de $\forall C_x \in BC$ teremos $sim(C_x, C_x) = 1$ e ainda que $\forall C_x, C_y \in BC$ teremos $sim(C_x, C_y) = sim(C_y, C_x)$, onde se procura um conhecimento que possui similaridade máxima a C_x , isto é, para uma solução de $C_x \in BC$ deve valer $\exists C_y \in BC (\forall C_z \in BC)$ onde teremos $sim(C_x, C_y) \geq sim(C_x, C_z)$.

Partindo-se de uma base de conhecimento $BC_0 \neq \emptyset$ e de uma medida de similaridade sim_0 , é gerada uma sequência de tuplas $(BC_1, sim_1), (BC_2, sim_2), \dots, (BC_h, sim_h)$ com $BC_h < BC_{h-1} | BC_h = BC_{h-1} - C_i, BC_h \subseteq \{C_1, C_2, \dots, C_j\}$. O objetivo do processo de manutenção da base de conhecimento é, em seu extremo, descrever uma sequência S que gere uma base CB_n exatamente por meio de uma tupla (BC_n, sim_n) , onde durante o processo a base a ser gerada é aproximada com a sequência $S_1 = (BC_1, sim_1), S_2 = (BC_2, sim_2), \dots, S_n = (BC_n, sim_n)$. A manutenção finaliza quando $\exists n \forall m S = (BC_n, sim_n) = (BC_m, sim_m)$, ou seja, durante a retirada de mais conhecimento $C_i, i \geq n$ a descrição do classificador (BC_n, sim_n) não se altere mais.

Capítulo 3

O Algoritmo Genético através de Tipos Abstratos de Dados

Neste capítulo apresentaremos a definição de tipos básicos, operadores e ambiente do algoritmo genético baseado em tipos abstratos de dados. Para ilustrar estas definições será apresentada uma instanciação deste algoritmo ao problema de alinhamento múltiplo de proteínas.

3.1 Algoritmo Genético via Tipo de Dados Abstrato

O algoritmo genético baseado em tipo de dados abstrato (GAADT) é um modelo computacional genético, onde indivíduos serão representados de acordo com o material genético (cromossomos), o qual tem nas bases nitrogenadas¹⁰ suas unidades elementares de formação (Vieira, 2003). Por exemplo, no caso da espécie humana, o material genético é constituído pelos elementos das bases nitrogenadas, sendo estas classificadas como: adenina (A), timina(T), guanina(G), citosina(C). Outro exemplo de material genético está relacionado ao problema do Caixeiro Viajante, onde os cromossomos (caminhos completos) constituídos pelos elementos da base, neste caso, referem-se aos nomes das cidades (Vieira, 2003).

Para indicar as características como as bases deverão ser agrupadas para se formar uma dada característica se faz necessário proporcionar uma lei de formação. A lei de formação de características será representada pelo conjunto de axiomas de formação de genes (AFG) (Vieira, 2003). As propriedades desta definição só podem ser completadas no momento da instanciação do algoritmo genético a um problema específico (Ahn, et al., 2006).

¹⁰ Juntamente com a desoxirribose e um ácido fosfórico compõem os ácidos nucleicos e são divididas em púricas que são as mesmas tanto no DNA quanto no RNA representadas por adenina e guanina. E as pirimídicas representadas pela citosina presente no DNA e RNA, a timina no DNA e a uracila no RNA.

A seguir serão apresentados os conceitos de gene, cromossomo, população, grau de adaptação e operador no algoritmo genético baseado em tipos abstratos de dados. Em seguida uma definição de AG no GAADT e um exemplo de instanciação do mesmo.

3.1.1 Gene

Um gene g é uma sequência formada pelos elementos da base que pertence ao conjunto AFG (Vieira, 2003). Os genes são representados por cadeias sobre o alfabeto $B = \{b_1, b_2, \dots, b_k\}$, existindo sempre um elemento em B , denotado por b_λ , que representa a base inócua, esta base ajudará a construção dos elementos neutros para os operadores do algoritmo genético baseado em tipos abstratos de dados. As regras sintáticas para a construção dos genes definidas pelo conjunto de axiomas de formação de genes AFG, definido de maneira axiomática. Para que as operações do GAADT possam manipular os objetos pertencentes ao tipo gene, é preciso primeiro sempre incluir um axioma em AFG que defina a construção do gene inócua denotado por g_λ , e de outros axiomas que construam as características do indivíduo de acordo com o problema em foco. O tipo do gene irá variar dependendo das restrições impostas pelo conjunto AFG aos seus elementos.

3.1.2 Cromossomo

Com o conjunto de genes $G = \{ \langle b_1, b_2, \dots, b_k \rangle \mid \forall x (1 \leq x \leq k) b_x \in B \wedge (b_1 \neq \emptyset \wedge b_2 \neq \emptyset \wedge \dots \wedge b_k \neq \emptyset) \}$ e $G \subset AFG$, os cromossomos são representados por sequências de genes que satisfazem as regras de sintáticas definidas pelo conjunto de axiomas de formação do cromossomo AFC (Vieira, 2003). Assim, os genes são agrupados em conjuntos para formar os cromossomos da população tal que $C = \{ \langle g_1, g_2, \dots, g_k \rangle \mid \forall x (1 \leq x \leq k) g_x \in G \}$. O conjunto de genes que compõe um dado cromossomo serve para identificar este cromossomo dentro da população. A identidade dos cromossomos será usada para impedir que várias cópias de um cromossomo possam coexistir ou renascer na população em qualquer tempo durante o processo de evolução da mesma na busca por um cromossomo mais adaptado.

3.1.3 População

Os cromossomos são agrupados em conjuntos para formar uma população $P = \{ \langle c_1, c_2, \dots, c_k \rangle \mid \forall x (1 \leq x \leq k) c_x \in C \}$ com $C \subset AFC$. Esta representação para a população irá garantir a imparcialidade na avaliação dos cromossomos que compõem a população, já que cada cromossomo só poderá ocorrer uma vez na população.

3.1.4 Grau de Adaptação

O grau de adaptação de uma população (adaptação média) é calculado em função do grau de adaptação do cromossomo, que por sua vez é calculado em função do grau de adaptação dos genes. O grau de adaptação de um gene é dado por $grau: G \rightarrow K$, onde K é um corpo ordenado, tal que a cada gene $g \in G$ lhe é associado um número K chamado de $grau(g)$ e que reflita, segundo o observador, uma medida estratificação entre a adaptação dos elementos de G , sendo a o grau de adaptação de g_λ o menor valor possível para esta função. Assim, o grau de adaptação de um cromossomo pode ser dado por $adapt: P \rightarrow K \mid adapt(c) = (\sum_{g \in gen(c)} grau(g)) / card(P)$, onde $card(P)$ é uma função que informa a cardinalidade da população P e K é um corpo ordenado¹¹, tal que a cada gene $g \in G$ lhe é associado um número K chamado de $grau(g)$ e que reflita, segundo o observador, uma estratificação comparada entre a adaptação dos genes de G .

3.1.5 Operadores

Uma indagação que pode ser observada é como criar cromossomos e que tipo de codificação escolher para geração de uma nova população. Os dois operadores básicos para criação de um novo indivíduo são cruzamento (*cross*) e mutação (*mut*) (Rudolph, 1994). No cruzamento, os cromossomos pais tem seu material genético combinado para produzir a descendência (Qi, et al., 1994). E na mutação os genes escolhidos são mudados (Fonseca, et al., 1998). No GAADT, o operador genético de cruzamento trabalha sobre um par de indivíduos selecionados em função dos requisitos do ambiente.

¹¹ Um corpo ordenado é uma estrutura algébrica, com duas operações, sem divisor próprio de zero e munido de uma ordem.

A seleção dos cromossomos que satisfazem um requisito $r \in Rq$ é feita através de um predicado sel_r dado por $sel_r(\{c_1, c_2, \dots, c_n\}) = \{c_i | \forall i(1 \leq i \leq n) r(c_i)\}$, onde $r(c_i)$ indica que o cromossomo c_i satisfaz o requisito $r \in Rq$.

Os indivíduos resultantes da aplicação do gerador genético de cruzamento são formados pela combinação dos genes fecundados. Os genes fecundados por sua vez são os genes pertencentes a um dos indivíduos fornecedores, e que satisfazem um dado requisito do ambiente e tem grau de adaptação maior ou igual ao grau de adaptação do gene do outro indivíduo e que também satisfaz ao mesmo requisito. A comparação entre dois genes é denominada de gene dominante é identificado pela função $domi$ que recebe um par de genes dos cromossomos pai e retorna o gene de maior grau de adaptação se os genes fornecidos satisfazem um mesmo requisito r do conjunto de requisito Rq de um ambiente e dada por $domi: G \times G \rightarrow G$, de tal forma que se $\forall r \in Rq \neg(r(g) \wedge r(g')) \vdash domi(g, g') = g_\lambda$. Caso $\exists r \in Rq r(g) \wedge r(g') \wedge (grau(g) > grau(g')) \vdash domi(g, g') = g$. E se $\exists r \in Rq r(g) \wedge r(g') \wedge (grau(g) < grau(g')) \vdash domi(g, g') = g'$. Onde $r(g)$ significa que o gene g satisfaz a regra r de Rq . A fecundação pode ser dada por $fec: \wp(G) \times \wp(G) \rightarrow \wp(G) | fec(g = \{g_1, g_2, \dots, g_n\}, g' = \{g'_1, g'_2, \dots, g'_m\}) = \{h | \forall i(1 \leq i \leq n) \forall j(1 \leq j \leq m) h = domi(g_i, g'_j)\}$ e a combinação dada por $comb: \wp(G) \rightarrow \wp(P)$, de tal forma que $comb(A = \{g_1, g_2, \dots, g_n\}) = \{c | \forall j(1 \leq j \leq n) \forall i(1 \leq i \leq j) \exists h_i \in G \wedge c = crom(\{h_1, h_2, \dots, h_i\} \cap A) \wedge c \in P \cap [AFC]\}$. Assim, o operador de cruzamento é dado por $cruz: sel_x(P) \times sel_y(P) \rightarrow P$, de tal forma que $cruz(c_1, c_2) = \{c | c \subseteq fec(c_1, c_2)\}$ (Vieira, 2003).

A ação genética de mutação é expressa por um predicado que forma uma população de indivíduos obtidos pela aplicação da função troca sobre todos os indivíduos da população atual. A função troca retorna os cromossomos gerados pela troca de um gene g por um gene g' no cromossomo em questão e dada por $troc: C \times G \times G \rightarrow C$, de tal forma que se $c \cup g_1 \in AFC \wedge (c \cup g_1) - g_2 \in AFC \vdash troc(c, g_1, g_2) = (c \cup g_1) - g_2$. Caso $c \cup g_1 \in AFC \wedge (c \cup g_1) - g_2 \notin AFC \vdash troc(c, g_1, g_2) = c \cup g_1$. Caso $c \cup g_1 \notin AFC \wedge c - g_2 \in AFC \vdash troc(c, g_1, g_2) = c - g_2$. E se $c \cup g_1 \notin AFC \wedge c - g_2 \notin AFC \vdash troc(c, g_1, g_2) = c$. Assim a mutação é dada por $mut \subseteq \wp(P) | mut(c) = \{c' \exists g, g' \in G c' = troc(c, g, g') \wedge c' \in pcorde(P)\}$ onde $pcorde$ é um critério de aceitação de cromossomos imposto pelo ambiente.

3.2 Algoritmo Genético pelo GAADT

Um algoritmo genético opera sobre populações de indivíduos que evoluem de acordo com as características de um ambiente A . Um ambiente A é uma 6-tupla $\langle P, \wp(P), Tx(P), \Sigma, Rq, AFG, AFC, P_0 \rangle$, onde P é uma população com no mínimo um indivíduos, $\wp(P)$ é o conjunto potência de P , Tx é a classificação taxonômica dos indivíduos da população P , Σ é um conjunto de operadores genealógicos que atuam sobre a população P (por exemplo: cruzamento, troca), Rq é um conjunto dos requisitos (características expressas através de fórmulas numa linguagem de primeira ordem) do ambiente A que influenciam a genealogia da população P , AFG é um conjunto de axiomas de formação dos genes dos indivíduos da população P , AFC é um conjunto de axiomas de formação dos cromossomos da população P e P_0 é uma sub-população de P , distinguida em $\wp(P)$, chamada população inicial.

O tipo população é o conjunto formado por todos os objetos do tipo cromossomo (ou seja, $P = P(P(C))$), que são possíveis resultados para o problema em foco segundo a interpretação adotada para os tipos C , G e B . Deve-se ressaltar que a geração da população vazia pelo algoritmo indica que a interpretação adotada para o problema está errada. Para o algoritmo genético baseado em tipo de dados abstrato, deve-se observar a necessidade de se estabelecer um critério de seleção sobre a população P que oriente o corte de indivíduos que não devem figurar numa população seguinte a ela. Por exemplo, o grau de adaptação média de P . Em um ambiente A , o algoritmo genético é dado por:

$$AG: A \rightarrow A | AG(P_n) = \begin{cases} P_{otm} & \text{se } P_{otm} = \{c | c \in P_{n+1} \wedge adap(c) \geq t\} \neq \emptyset \\ P_{n+1} & \text{se } n + 2 = k \\ AG(P_{n+1}) & \text{caso contrário} \end{cases}$$

Onde, $P_{n+1} = cruz(a, b) \cup mut(c) \cup Rq(P_n)$, P_0 é a população inicial considerada, t é um valor imposto pelo ambiente A , como critério de aceitação de indivíduos em P_n , $k \in K$ é um número dado como critério de satisfação do número de iterações e $a, b, c \in P_n$.

3.3 Exemplo

Para demonstrar o uso do GAADT, nesta seção iremos mostra um exemplo de instanciação do GAADT (Furtado, 2006), no qual mostra um modelo capaz de realizar o alinhamento múltiplo de proteínas. Onde cada cromossomo irá representar um possível alinhamento entre as

sequências de proteínas e a adaptação do cromossomo é calculada conforme as bases se dispõem no alinhamento.

Segundo Furtado (2006), o alinhamento de múltiplas proteínas pode ser resumido da seguinte forma: dado um conjunto de proteínas, que são cadeias de aminoácidos, $S = \{s_1, s_2, \dots, s_m\} | \forall i (1 \leq i \leq m) \wedge m \in \mathbb{N}$, sejam os pesos $\alpha, \beta, \delta \in \mathbb{Q}$, onde α expressa a relevância do alinhamento de x pares de aminoácidos iguais na mesma posição nas cadeias, β para o alinhamento de y pares de aminoácidos diferentes na mesma posição e δ para o alinhamento de z pares formados por uma lacuna representado pelo símbolo “-” e um aminoácido na mesma posição, com $x, y, z \leq m$. O alinhamento de S é um conjunto $R = \{r_1, r_2, \dots, r_m\} | \forall i (1 \leq i \leq m)$, construídas pela inserção de lacunas nas sequências dos elementos de um conjunto $s_i \in S | \forall i (1 \leq i \leq m)$ o valor da formula $\alpha x + \beta y + \delta z$ é máximo, e que para cada sequência $s_j \in S | \forall j (1 \leq j \leq m)$ exista uma única sequência $r_j \in R | \forall j (1 \leq j \leq m)$, cuja remoção das lacunas de r_j reproduza a sequência s_j dada.

A base B foi definida como o conjunto formado pela união do conjunto de aminoácidos $A_{prot} = \{D, E, A, R, N, C, F, G, Q, H, I, L, K, M, P, S, Y, T, W, V\}$ com o símbolo $\{-\}$ que serão a base inócua, e pelo produto com o conjunto \mathbb{N} . Ou seja, $B = A_{prot} \cup \{-\} \times \mathbb{N}$.

O conjunto de gene foi dado por $G = \{g_j = \langle b_{1,j}, b_{2,j}, \dots, b_{i,j} \rangle | \forall k (1 \leq k \leq i) g_j \in B\}$ que satisfaz o axioma $AFG = \{g_j = \langle b_{1,j}, b_{2,j}, \dots, b_{i,j} \rangle | i = card(S) \wedge b_{i,j} \in B \wedge \forall j (1 \leq j \leq i) g_j \subset membro(b_{i,j}, s_j) \vee (b_{i,j} = \{-\})\}$, onde $card(S)$ é a função que recebe um conjunto e retorna a sua cardinalidade e $membro(b_{i,j}, s_j)$ é um predicado que recebe um elemento e uma cadeia e retorna verdade se o elemento ocorre na cadeia e falso caso contrário. Um exemplo de genes no problema de alinhamento múltiplo de proteínas é ilustrado na Figura 3.1.

g_1	g_2	g_3	g_4	g_5	g_6	g_7	g_8	g_9	g_{10}
$\langle N, 1 \rangle$	$\langle L, 2 \rangle$	$\langle -, 0 \rangle$	$\langle V, 3 \rangle$	$\langle N, 4 \rangle$	$\langle S, 5 \rangle$	$\langle E, 6 \rangle$	$\langle H, 7 \rangle$	$\langle R, 8 \rangle$	$\langle M, 9 \rangle$
$\langle N, 1 \rangle$	$\langle L, 2 \rangle$	$\langle Y, 3 \rangle$	$\langle V, 4 \rangle$	$\langle P, 5 \rangle$	$\langle S, 6 \rangle$	$\langle E, 7 \rangle$	$\langle M, 8 \rangle$	$\langle I, 9 \rangle$	$\langle -, 0 \rangle$
$\langle -, 0 \rangle$	$\langle -, 0 \rangle$	$\langle Y, 1 \rangle$	$\langle V, 2 \rangle$	$\langle -, 0 \rangle$	$\langle N, 3 \rangle$	$\langle E, 4 \rangle$	$\langle H, 5 \rangle$	$\langle M, 6 \rangle$	$\langle -, 0 \rangle$

Figura 3.1- Exemplo de genes conforme problema em (Furtado, 2006).

O conjunto de cromossomo foi dado por $C = \{c_x = \langle g_1, g_2, \dots, g_x \rangle | \forall k (1 \leq k \leq x) g_k \in G\}$ que satisfaz o axioma $AFC = \{c_x = \langle g_1, g_2, \dots, g_x \rangle | (\forall i (1 \leq i \leq y) \exists s_i | (\forall j (1 \leq j \leq$

$x)b_{i,j} \in g_j \wedge \exists r_i = \text{concat}(\text{prim}(b_{i,j}) \wedge (s_i = \text{rem}(r_i)))$. Onde $\text{concat}(\text{prim}(b_{i,j}))$ é um predicado que concatena os símbolos dos primeiros elementos das bases $b_{i,j}$ e $\text{rem}(r_i)$ é uma função para remoção das lacunas em r_i . Um exemplo de cromossomo é ilustrado na Figura 3.2.

c ₁	<i>g</i> ₁	<i>g</i> ₂	<i>g</i> ₃	<i>g</i> ₄	<i>g</i> ₅
	< N, 1 >	< L, 2 >	< -, 0 >	< V, 3 >	< N, 4 >
	< N, 1 >	< L, 2 >	< Y, 3 >	< V, 4 >	< P, 5 >
	< -, 0 >	< -, 0 >	< Y, 1 >	< V, 2 >	< -, 0 >
	<i>g</i> ₆	<i>g</i> ₇	<i>g</i> ₈	<i>g</i> ₉	<i>g</i> ₁₀
	< S, 5 >	< E, 6 >	< H, 7 >	< R, 8 >	< M, 9 >
	< S, 6 >	< E, 7 >	< M, 8 >	< I, 9 >	< -, 0 >
	< N, 3 >	< E, 4 >	< H, 5 >	< M, 6 >	< -, 0 >

Figura 3.2 - Exemplo de cromossomos conforme problema em (Furtado, 2006).

A população foi dada por $P = \{p_i = \langle c_1, c_2, \dots, c_q \rangle \mid (\forall k(1 \leq k \leq q)c_k \in C)\}$ com $C \subset AFC$. Um exemplo de população é ilustrado na Figura 3.3, juntamente com cada cromossomo e seus respectivos genes.

P ₁	c ₁	<i>g</i> ₁	<i>g</i> ₂	<i>g</i> ₃	<i>g</i> ₄	<i>g</i> ₅	
		< N, 1 >	< L, 2 >	< -, 0 >	< V, 3 >	< N, 4 >	
		< N, 1 >	< L, 2 >	< Y, 3 >	< V, 4 >	< P, 5 >	
		< -, 0 >	< -, 0 >	< Y, 1 >	< V, 2 >	< -, 0 >	
		<i>g</i> ₆	<i>g</i> ₇	<i>g</i> ₈	<i>g</i> ₉	<i>g</i> ₁₀	
		< S, 5 >	< E, 6 >	< H, 7 >	< R, 8 >	< M, 9 >	
		< S, 6 >	< E, 7 >	< M, 8 >	< I, 9 >	< -, 0 >	
		< N, 3 >	< E, 4 >	< H, 5 >	< M, 6 >	< -, 0 >	
		c ₂	<i>g</i> ₁	<i>g</i> ₂	<i>g</i> ₃	<i>g</i> ₄	<i>g</i> ₅
			< N, 1 >	< L, 2 >	< -, 0 >	< V, 3 >	< N, 4 >
	< N, 1 >		< L, 2 >	< Y, 3 >	< V, 4 >	< P, 5 >	
	< -, 0 >		< Y, 1 >	< -, 0 >	< V, 2 >	< -, 0 >	
	<i>g</i> ₆		<i>g</i> ₇	<i>g</i> ₈	<i>g</i> ₉	<i>g</i> ₁₀	
	< S, 5 >	< E, 6 >	< H, 7 >	< R, 8 >	< M, 9 >		
	< S, 6 >	< E, 7 >	< M, 8 >	< I, 9 >	< -, 0 >		
< N, 3 >	< E, 4 >	< H, 5 >	< -, 0 >	< M, 6 >			
c ₃	<i>g</i> ₁	<i>g</i> ₂	<i>g</i> ₃	<i>g</i> ₄	<i>g</i> ₅		
	< N, 1 >	< L, 2 >	< V, 3 >	< N, 4 >	< -, 0 >		
	< N, 1 >	< L, 2 >	< Y, 3 >	< V, 4 >	< P, 5 >		
	< -, 0 >	< Y, 1 >	< V, 2 >	< -, 0 >	< N, 3 >		
	<i>g</i> ₆	<i>g</i> ₇	<i>g</i> ₈	<i>g</i> ₉	<i>g</i> ₁₀		
< S, 5 >	< E, 6 >	< H, 7 >	< R, 8 >	< M, 9 >			
< S, 6 >	< E, 7 >	< -, 0 >	< M, 8 >	< I, 9 >			
< -, 0 >	< E, 4 >	< -, 0 >	< H, 5 >	< M, 6 >			

Figura 3.3- Exemplo de população conforme problema em (Furtado, 2006).

O grau de adaptação de um gene foi dado por $grau: G \rightarrow \mathbb{R}$, de tal forma que:

$$grau(g_j = \langle b_{1,j}, b_{2,j}, \dots, b_{m,j} \rangle) = \sum_{x=1}^{x'} p(\text{getPairs}(g_j)_x) + \sum_{y=1}^{y'} G_{pen}(\text{getGaps}(g_j)_y).$$

Onde $\text{getPairs}(g_j)$ busca todas as possíveis combinações de pares de bases sem lacunas, $\text{getGaps}(g_j)$ busca todas as possíveis combinações de pares de bases com lacunas e $p(a_u, a_v)$ é a função que determina a pontuação entre os aminoácidos a_u e a_v .

O grau de adaptação de um cromossomo foi dado por $adapt: C \rightarrow \mathbb{R}$, tal que $adapt(c) = \sum_{j=1}^L grau(g_j)$, onde L é o tamanho do cromossomo (número de genes do cromossomo).

O cruzamento foi definido pela função $cruz: C \times C \rightarrow \wp(P)$, tal que $cruz(c_1, c_2) = \{c' \mid c' \subseteq fec(c_1, c_2)\}$, onde $fec(c_1, c_2) = \{g = \text{domi}(g_x, g_y) \mid \forall g_x \in c_1 \wedge \forall g_y \in c_2\}$.

A mutação foi definida pela função $mut: C \rightarrow \wp(P)$, tal que $mut(c) = \{c' = \text{troc}(c, G_1, G_2) \mid \exists G_1, G_2 \in \wp(G)\}$, onde $\text{troc}(c, G_1, G_2) = (c - G_1) \cup G_2$ se $(c - G_1) \cup G_2 \in C$ ou do contrário $\text{troc}(c, G_1, G_2) = c$.

Em (Furtado, 2006) o algoritmo genético para o problema do alinhamento múltiplo de proteínas ficou definido pela função $GAADT_A: A \rightarrow A$, tal que $GAADT_A(P_t) = \begin{cases} P_{t+1} & \text{se } t + 2 = T \\ GAADT_A(P_{t+1}) & \text{caso contrário} \end{cases}$ onde $P_{t+1} = \text{pcorte}(P_t) \cup cruz(a, b) \cup mut(c)$ com $a, b, c \in P_t$ e $T \in \mathbb{N}$ é um número dado como critério de satisfação do número de iterações.

Capítulo 4

Modelagem do Sistema de Apoio à Compra Eletrônica

Com o crescimento das cidades, fica cada vez mais difícil um órgão público ou privado, prestar serviços de atendimento ao público, pois a quantidade de pessoas e entidades, que ali se encontram, aumenta na mesma proporção da evolução da cidade, aumentando o consumo de produtos e serviços daquele órgão. Assim, se faz necessário uma reposição constante de estoque de produtos usados naquele órgão e uma expansão ou melhoria esporádica dos equipamentos usados para prestação de serviços.

É de grande importância a utilização de uma ferramenta que auxilie no processo de tomada de decisão para compras de produtos ou equipamentos de qualidade com bom preço, diminuindo o tempo para a escolha dos mesmos e melhorando a qualidade do atendimento.

A seguir, iremos mostrar a especificação e descrição do protótipo do sistema criado a partir das técnicas estudadas neste trabalho, bem como os detalhes de implementação¹². Foram utilizadas as seguintes tecnologias: UML, banco de dados e as técnicas de algoritmo genético através de tipo de dados abstrato.

4.1 Esquema Relacional

A base de conhecimento do sistema é a memória de todas as experiências previamente armazenadas. Na Figura 4.1 apresentamos o esquema relacional proposto para a criação e adaptação de sistemas automatizados.

¹² Nos apêndices A e B encontramos, respectivamente, as especificações de UML da fase de análise e projeto do sistema.

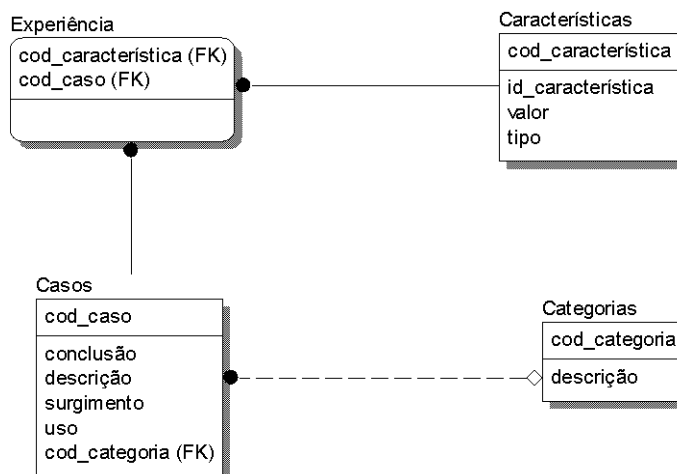


Figura 4.1 - Esquema relacional proposto.

Com a estrutura proposta na Figura 4.1 é possível armazenar as experiências de compra e venda com suas respectivas características. O mecanismo de gerenciamento e armazenamento das experiências poderá armazenar quantas características forem necessárias. Cada caso de experiência estará relacionado com uma categoria, significando que uma mesma base poderá trabalhar em mais de um contexto, ou seja, em um único sistema o usuário final poderá escolher em que contexto deverá solucionar seu problema. A Tabela 4.1 mostra a descrição de cada entidade envolvida no modelo relacional da Figura 4.1.

Tabela 4.1 - Descrição das entidades existentes no modelo relacional proposto.

Entidade	Descrição
Casos	Cadastro de cada ocorrência de compra ou venda passada, contendo sua conclusão, descrição, tempo (data e hora) do surgimento e código da categoria (ou grupo) que o caso faz parte.
Categorias	Cadastro do contexto que um grupo de ocorrência de compra ou venda passadas esta associado.
Características	Cadastro dos segmentos que compõem as experiências de compra ou venda.
Experiência	Associação das características (ou segmentos) para cada ocorrência de compra ou venda.

4.2 Instanciação pelo GAADT

As ocorrências de compra ou venda será a população do algoritmo genético desta dissertação. Logo, uma base B será considerada como um conjunto de todas as unidades genéticas elementares e poderá ser usadas na formação do material genético dos cromossomos de uma

população. Para ilustrar esse vínculo, tomemos como base os dados da Tabela 4.2, conforme o que segue:

Tabela 4.2 - Exemplo de ocorrências armazenadas na base de dados.

Referência	Período	Descrição	Preço	Modelo
1	24/07/2008	Aparelho de Pressão Digital	199,00	BP33AA1
2	05/08/2008	Aparelho de Pressão Digital	99,00	MS-918
3	05/08/2008	Aparelho de Pressão Digital	162,00	Microlife
4	05/08/2008	Aparelho de Pressão Digital	139,00	BP3AF13
5	05/08/2008	Oxímetro de Pulso	650,00	Com Fio
6	20/09/2007	Otoscópio	307,00	Pocket Jr
7	05/09/2007	Otoscópio	329,00	Pocket Jr
8	10/10/2007	Oxímetro de Pulso	1.845,00	Sem Fio
9	15/10/2007	Otoscópio	259,00	Control
10	20/10/2007	Oxímetro de Pulso	1.295,00	Sem Fio

A Tabela 4.2 é uma amostra dentro de um domínio de compras on-line que originalmente contém mais de mil registros de vendas cadastrados. Fizemos vários testes com dados extraídos do site Buscapé¹³ através do mecanismo existente no próprio software de protótipo criado para ilustrar o funcionamento dos modelos apresentados nesta dissertação, a Figura 4.2 ilustra o mecanismo citado.

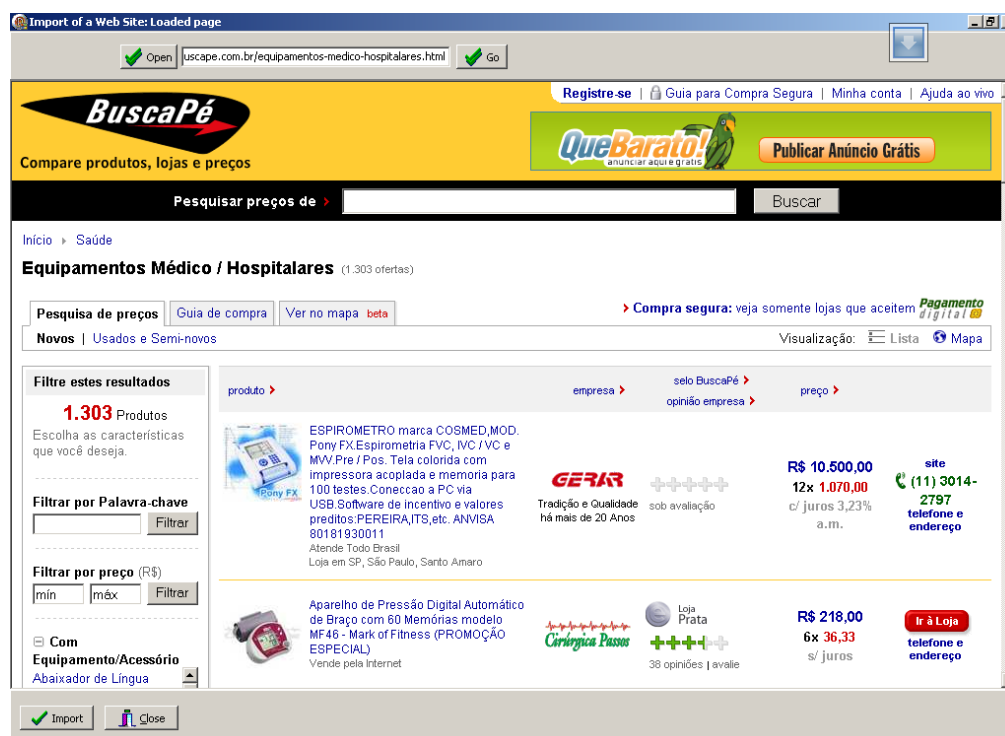


Figura 4.2 - Tela de extração através do site Buscapé.

¹³ <http://www.buscape.com.br> (Site de comparação de preços).

Para exemplificar o funcionamento do modelo, usaremos o exemplo do problema representado na Tabela 4.2 no qual envolve 10 registros de casos de compra e venda de produtos eletroeletrônicos hospitalares, onde cada registro é formado por n características (período de compra ou consulta, descrição, preço e modelo do produto). Cada característica terá uma identificação, valor, tipo e um código para distinguir das outras características. Por exemplo, a ocorrência 1 tem a característica identificada como *Período*, um valor *24/07/2008*, um tipo *Data*¹⁴ e o código *1.1*. Outra característica, ainda na ocorrência 1, seria a característica *Preço*, com valor *R\$199⁰⁰*, com tipo *Numérico* e o código *1.3*.

Dentro do que é estudado em (Vieira, 2003) e (Raymer, et al., 2000), poderemos dar as seguintes definições para o problema de apoio à compra eletrônica: Por A entenda-se o conjunto de n características, onde tais características são formadas uma base B citada no modelo na Tabela 4.3.

Tabela 4.3 - Modelo da base B do ambiente.

Título	Definição
Base B do ambiente	Representação da base B do gene para o ambiente A .
$B = B_{\text{códigos}} \times B_{\text{Identificadores}} \times B_{\text{Tipos}} \times B_{\text{Valores}}$	
Onde:	
$B_{\text{códigos}} = \{cod \in \mathbb{N} cod \neq \emptyset\}$	
$B_{\text{Identificadores}} = \{i \in \text{Alfabético}\}$	
$B_{\text{Tipos}} = \{\text{"Numérico"}, \text{"Data"}, \text{"Alfabético"}\}$	
$B_{\text{Valores}} = \{v \in \mathbb{R} \vee v = ddmmaa dd, mm, aa \in \mathbb{N} 1 \leq dd \leq 31, 1 \leq mm \leq 12 \vee v \neq \emptyset\}$	

Na Tabela 4.3, temos as unidades elementares de formação do gene para o ambiente A : um código, um identificador (ou descrição), um tipo e um valor correspondente ao tipo. O código cod é o campo que irá distinguir uma característica das outras e o conjunto possível de códigos estará dentro dos números naturais ($cod \in \mathbb{N}$). A função do identificador é associar um rótulo id a característica e o conjunto possível de identificadores estarão dentro dos símbolos alfabéticos ($id \neq \emptyset$). O tipo trata-se do domínio que a característica da ocorrência de compra ou venda representa, onde o conjunto possível de tipos será *Numérico*, *Data* e *Alfabético*. E os conjuntos possíveis de valores de acordo com o tipo escolhido são definidos no modelo da Tabela 4.4.

¹⁴ É um ponto flutuante (número real) cuja parte inteira armazena o número de dias desde um ponto de referência (exemplo, 30/12/1899) e cuja parte fracionária armazena o quanto se passou do dia.

Tabela 4.4 - Modelo da função $valor(v)$.

Título	Definição
Função $valor(v)$	Conjunto de possíveis valores de acordo com o tipo escolhido.
$valor(x) = \begin{cases} v \in \mathbb{R} & \text{se } x['tipo'] = "Numérico" \\ v \in ddmmaa & \text{se } x['tipo'] = "Data", \{dd, mm, aa \in \mathbb{N} 1 \leq dd \leq 31, 1 \leq mm \leq 12\} \\ v \in \mathbb{N} & \text{se } x['tipo'] = "Alfabético" \end{cases}$	

A representação do conjunto dos genes do ambiente A é dada pelo modelo $G = \{g = \langle cod, id, t, v \rangle | cod, id, t, v \in B\}$, obedecendo as regras axiomáticas de formação de gene da Tabela 4.5.

Tabela 4.5 - Modelo do AFG do ambiente.

Título	Definição
AFG do ambiente	Representação do conjunto de axiomas de formação de genes do ambiente A .
$AFG = \{g = \langle cod, id, t, v \rangle \forall (cod, id, t, v) \exists (cod \neq 0 \wedge id \neq \emptyset \wedge t \neq \emptyset \wedge v \neq \emptyset)\}$	

Na Tabela 4.6, temos C como o cromossomo e g_n como o gene pelo modelo GAADT que, em A , equivale a um determinada ocorrência de compra ou venda. Ou seja, C representa o caso de compra e venda e g_n as características deste caso.

Tabela 4.6 - Modelo do cromossomo C do ambiente.

Título	Definição
Cromossomo C do ambiente	Representação do conjunto dos cromossomos do ambiente A .
$C = \{\langle g_1, g_2, \dots, g_n \rangle \forall x (1 \leq x \leq n) g_x \in G\}$	

O conjunto de axiomas de formação do cromossomo para o ambiente A é definido na Tabela 4.7 a seguir.

Tabela 4.7 - Modelo do AFC do ambiente.

Título	Definição
AFC do ambiente	Representação do conjunto de axiomas de formação do cromossomo do ambiente A .
$AFC = \{C_i = \langle g_1, g_2, \dots, g_x \rangle, C_j = \langle g_1, g_2, \dots, g_y \rangle \forall C_i, C_j \in C \wedge \forall m, n \in (1 \leq m \leq x, 1 \leq n \leq y) \wedge \forall g_m, g_n \in G (cod_m \neq cod_n \wedge id_m = id_n)\}$	

Para as ocorrências da Tabela 4.2, como resultado, teríamos o seguinte conjunto de genes para o cromossomo da ocorrência 1:

Tabela 4.8 - Resultado da implementação do cromossomo para a ocorrência 1.

<i>cod</i> (código)	<i>id</i> (identificador)	<i>t</i> (tipo)	<i>v</i> (valor)
1.1	Período	Data	24/07/2008
1.2	Descrição	Alfabético	Aparelho de Pressão Digital
1.3	Preço	Numérico	199, ⁰⁰
1.4	Modelo	Alfabético	BP33AA1

Para cada acontecimento ou experiência de compra ou venda contida na Tabela 4.2 teremos um resultado de implementação do cromossomo correspondente. Para a realização das operações se faz necessário a definição da função $grau(g)$ que está determinada na Tabela 4.9 a seguir.

Tabela 4.9 - Modelo da função $grau(g)$.

Título	Definição
Função $grau(g)$	Função de adaptação do gene do ambiente A .
$grau(g) = \begin{cases} \frac{1}{valor(g)}, & \text{se } valor(g) \in Numérico \\ \frac{1}{data\ atual - valor(g)}, & \text{se } valor(g) \in Data \\ \ln r r \equiv representação \in \mathbb{N}, & \text{se } valor(g) \in Alfabético \end{cases}$	

Na Tabela 4.10 descreve a função dominante entre dois genes para o ambiente A , retornando o gene dominante.

Tabela 4.10 - Modelo da função $domi(g_1, g_2)$.

Título	Definição
Função $domi(g_1, g_2)$	Função dominante entre dois genes.
$domi(g_1, g_2) = \begin{cases} g_\lambda & \text{se } tipo(g_1) \neq tipo(g_2) \\ g_1 & \text{se } tipo(g_1) = tipo(g_2) \wedge grau(g_1) > grau(g_2) \\ g_2 & \text{se } tipo(g_1) = tipo(g_2) \wedge grau(g_2) > grau(g_1) \end{cases}$	

Para a função $tipo(g)$ usada na Tabela 4.10, a mesma pode ser expressa na Tabela 4.11 a seguir.

Tabela 4.11 - Modelo da função $tipo(g)$.

Título	Definição
Função $tipo(g)$	Função tipo do gene.
$tipo(g) = \begin{cases} \text{Numérico} & \text{se } (g \in B_{\text{códigos}}) \vee (g \in \mathbb{R}) \vee (g \in \mathbb{N}) \\ \text{Alfanumérico} & \text{se } g \in B_{\text{identificadores}} \vee g \in B_{\text{tipos}} \\ \text{Data} & \text{se } g = ddmmaa dd, mm, aa \in \mathbb{N} 1 \leq dd \leq 31, 1 \leq mm \leq 12 \end{cases}$	

Diante do exposto, sugere-se utilizar os operadores *cross* e *mut*, citados na seção 3.1.5. Assim, a modelagem de criação do algoritmo genético será capaz de gerar novas ocorrências com cardinalidade menor que as ocorrências atuais e diminuir o espaço da base de dados para aumentar a eficiência de recuperação dentro de um domínio e-commerce para aquisição de materiais eletroeletrônicos. A descrição exposta na Tabela 4.12 reforça a teoria do GAADT definidas no Capítulo 3.

Tabela 4.12 - Modelo da função $AG(P_n, t, k)$.

Título	Definição
Função $AG(P_n, t, k)$	Função algoritmo genético do ambiente A .
$AG(P_n, t, k) = \begin{cases} P_n = P_{otm}, & \text{se } adapt_m(P_n) > t \\ P_{n+1}, & \text{se } n + 2 = k \\ AG(P_{n+1}, t, k), & \text{do contrário} \end{cases}$	

Dentro do que foi mencionado na seção 3, na Tabela 4.12 temos que:

- P_n é o conjunto de experiências dentro de momento n . P_o será o conjunto de experiência inicial oriundo da base de dados;
- P_{otm} é o conjunto de experiências otimizados ou ideal;
- $adapt_m(P_n) > t$ é uma condição de parada para garantir que um determinado conjunto experiências, representado pelos cromossomos C_n , tenha uma adaptação maior do que um valor imposto t ;
- $n + 2 = k$ é uma condição de parada para garantir que mesmo o algoritmo não convergindo para um P_{otm} ele parará em um número de k iterações;

A Figura 4.3 ilustra o pseudocódigo do algoritmo genético da função $AG(P_n, t, k)$.

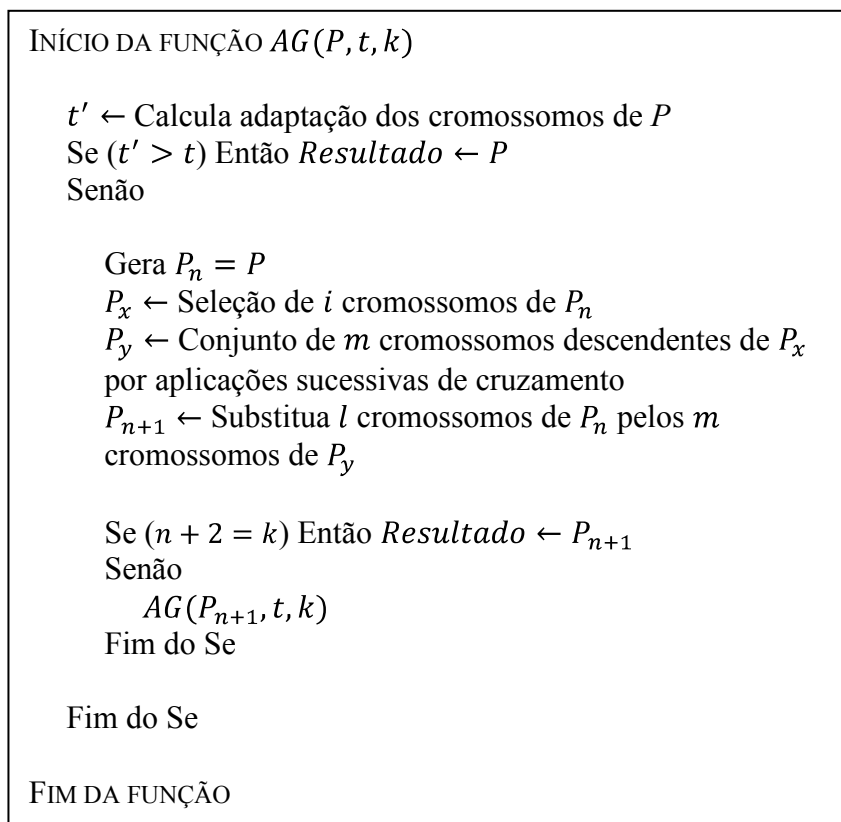


Figura 4.3 - Pseudocódigo da função $AG(P_n, t, k)$.

Assumindo os registros contidos na Tabela 4.2 como população, ao aplicar-se o algoritmo exposto na Figura 4.3, os resultados poderão ser visualizados a seguir:

Tabela 4.13 - Nova base de dados, com cardinalidade menor que a base de dados inicial.

Referência	Período	Descrição	Preço	Modelo
11	05/08/2008	Aparelho de Pressão Digital	119, ⁰⁰	BP3AF13
12	15/10/2007	Otoscópio	288, ⁵⁰	Pocket Jr
13	05/08/2008	Oxímetro de Pulso	1.110, ⁵⁰	Sem Fio

A Tabela 4.13 é o auxílio dado pelo sistema na seleção de material eletroeletrônico. Por exemplo, referente ao item descrito por *Otoscópio* da Tabela 4.2, recomenda-se a compra do modelo *Pocket Jr* por um preço de R\$288,⁵⁰.

A Figura 4.4 ilustra a tela do *software*, criado para ilustrar funcionamento dos modelos apresentados nesta dissertação, referente aos casos de compra ou venda existentes na base de dados com 10.000 ocorrências cadastradas, aproximadamente.

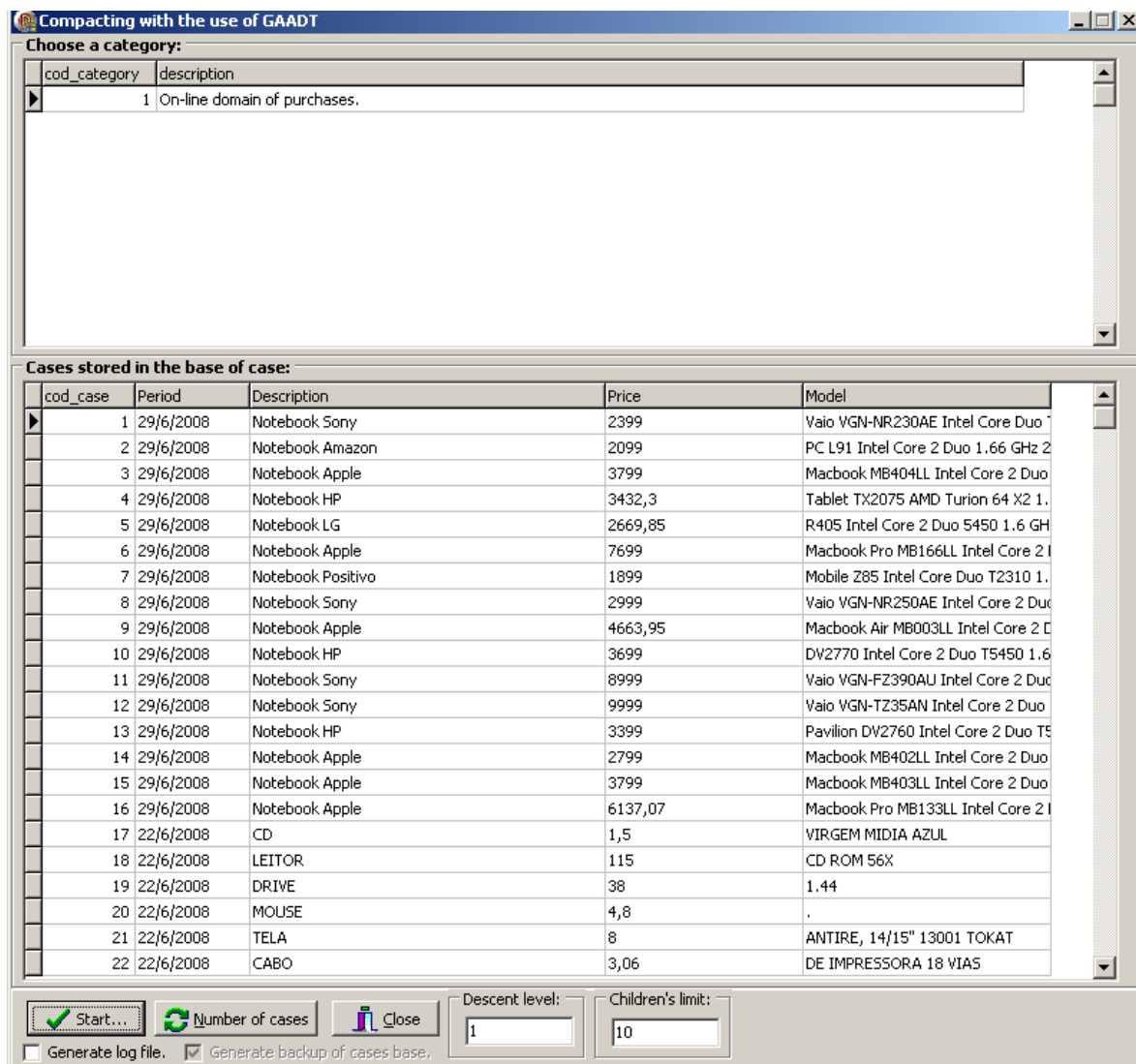


Figura 4.4 - Tela de compactação do software protótipo.

4.3 Resultados

A análise e o desenvolvimento da presente instanciação foram realizados utilizando dados públicos de produtos eletroeletrônicos oriundos da *Internet* no período de março de 2008 a abril de 2009. O propósito do levantamento destes dados foi servir como base e criação de banco de dados para testes e comparações entre alguns métodos de recomendação de produtos existentes, o que permitiu comparar os resultados da instanciação do GAADT apresentada nos capítulos anteriores com alguns sistemas analisadas.

A vantagem de coletar dados da *Internet* é de poder contar com dados reais, em grande quantidade e acessíveis a qualquer momento. Os dados podem ser encontrados em sites de

vendas (por exemplo, lojas virtuais como Extra¹⁵, Americanas¹⁶, CompraFacil¹⁷), em sites de pesquisas de produtos (por exemplo, mecanismo de buscas como BuscaPé¹⁸, Bondfaro¹⁹) e, muitas vezes, estes dados são localizados de uma fonte mais próxima possível (por exemplo, um representante dos produtos Xerox²⁰) do fabricante ou até mesmo um site de vendas do próprio fabricante (por exemplo, Brastemp²¹).

4.3.1 Desempenho

O ambiente de execução dos testes foi composto sucintamente por: um processador Intel Pentium 4 de 3.0 Ghz, 1024 MB de memória RAM e sistema operacional Windows XP SP3. O GAADT foi implementado²² na linguagem de programação Object Pascal, usando MySQL como banco de dados. A tabela a seguir informa o tempo de resposta do protótipo e a quantidade de sugestões geradas com relação ao volume de dados (número de tuplas) e a situação em que os dados aparecem.

Tabela 4.14 - Tempo de resposta.

Volume (número de tuplas)	Situação dos dados	Tempo de resposta	Convergência (número de sugestões)
100	Cem produtos idênticos ²³ .	< 1s	1
1000	Mil produtos idênticos.	3m, 16s	1
10000	Dez mil produtos idênticos.	3h, 43m, 14s	1
100	Cem produtos homogêneos ²⁴ .	< 1s	1
1000	Mil produtos homogêneos.	2m, 57s	1
10000	Dez mil produtos homogêneos.	3h, 17m, 14s	1
1000	Mil produtos heterogêneos ²⁵ .	~ 3s	78
5000	Cinco mil produtos heterogêneos.	2m, 33s	92
10000	Dez mil produtos heterogêneos.	11m, 46s	114

¹⁵ Em <http://www.extra.com.br/>.

¹⁶ Em <http://www.americanas.com.br/>.

¹⁷ Em <http://www.comprafacil.com.br/>.

¹⁸ Em <http://www.buscapede.com.br/>.

¹⁹ Em <http://www.bondfaro.com.br/>.

²⁰ Em <http://www.lealcopiadoras.com.br/>.

²¹ Em <http://www.brastemp.com.br/ch/iprodutos.aspx>.

²² Realizações também foram feitas nas linguagens Java e PHP, e em várias versões de base de dados (Access, Interbase, Firebird e Postgres). Por ser uma solução *Desktop* e com conexão a *Internet*, as tecnologias usadas para a versão atual do protótipo foram as que obtiveram maior performance com a quantidade de dados coletados e a configuração do computador usado para testes.

²³ Representando uma situação crítica dos dados com uma redundância de 100%.

²⁴ Representando uma situação com foco em um único tipo de produto, com características diferentes.

²⁵ Representando uma situação com foco em vários tipos de produtos, com características diferentes cada.

Na Tabela 4.14 é possível perceber que quanto mais heterogêneo for a base de dados, mais rápido é a convergência da instância do modelo GAADT para base de dados com grande volume de dados. Ou seja, para base de dados de grande volume leva-se mais tempo e recurso computacional para as situações onde na base de dados há dados redundantes. Porém, quanto maior a redundância maior será a convergência da sugestão de compra, tendendo para uma única sugestão.

4.3.2 Trabalhos Relacionados

Os resultados apresentados comparam o protótipo desenvolvido com os sistemas de aconselhamentos abaixo e cujos desempenhos estão resumidos na Tabela 4.15:

- Sistema de Raciocínio Baseado em Casos para Recomendação de Programa Alimentar (DietSysNet): É aplicado ao domínio da especialidade médica de nutrição. O benefício principal do sistema é apoiar a tomada de decisão do profissional de nutrição (Telles, et al., 2006);
- *Intelligent agent-based systems for personalized recommendations in Internet commerce* (MovieGuide e NB Computers): O primeiro é aplicado a recomendação de filmes em DVD na Internet. E o segundo é aplicado na sugestão de computadores portáteis a partir de uma lista de *rank* de computadores portáteis. O benefício principal, dos dois sistemas, é apoiar a tomada de decisão de um usuário que utiliza a Internet como meio de entretenimento (Wei, et al., 2002).
- Sistemas de recomendação de lojas virtuais como Americanas.com, Extra.com, Shoptime.com, CompraFacil.com e Submarino.com: São aplicados nas compras de produtos de vários segmentos como informática, eletrônicos, eletrodomésticos, livros, lazer, esporte, brinquedos, entre outros. O benefício principal do sistema é deixar o usuário informado de quais produtos estão em ofertas naquele determinado estabelecimento de acordo as preferências armazenadas no sistema de cada loja virtual;
- Sistemas de alerta de preços de sites de pesquisa de preços como Buscape.com e Bondefaro.com: São aplicados na divulgação do preço igual ou inferior ao mínimo cadastrado no sistema do site de busca de produtos de vários segmentos como informática, eletrônicos, eletrodomésticos, livros, lazer,

esporte, brinquedos, entre outros, que se pretende obter. O benefício principal do sistema é deixar o usuário informado do preço que o mesmo quer pagar para adquirir o produto.

Tabela 4.15 - Resumo do desempenho de outros sistemas de aconselhamento.

Sistema	Situação da entrada e ambiente de execução	Descrição do tempo de resposta	Descrição da saída
DietSysNet	40 registros foram usados para testes. Faz-se necessário de um teste preliminar dos dados armazenados, onde um nutricionista irá realizar quantos ajustes forem necessários, nos coeficientes de cada entrada, até garantir que a simulação está próxima da realidade.	Indeterminado. Pois, durante os testes houve problemas graves de desempenho de sistema (tempo de processamento e resposta).	11 novos registros foram gerados e 9 registros foram reavaliados.
MovieGuide	100 registros foram usados para testes. Onde, 30 filmes foram cadastrados como exemplos positivos, 30 exemplos negativos e 40 filmes não foram classificados.	Não informa o tempo e sim número de iterações que variou de 20 ^a a 90 ^a geração.	10 novos registros foram gerados.
NB Computers	Quantidade de registros não revelada. O usuário do sistema entra com os dados do computador que pretende comprar. Logo em seguida, o sistema começa a realizar dez perguntas referentes ao grau de preocupação com os dez atributos da informação dada. O menor grau de preocupação é 1, significando que a característica do produto não é importante, e 5, significando que a característica do produto é muito importante.	Não informado.	Mostra o <i>rank</i> calculado de cada característica repassada na informação de entrada e dá à oportunidade ao usuário de intervir nos coeficientes calculados de cada característica, caso o mesmo tenha conhecimento prévio de qual característica deveria ter maior ou menor <i>rank</i> .
Lojas virtuais	Indeterminado. O usuário faz um cadastro no sistema do estabelecimento virtual. Durante o cadastro, ou após, é feito um preenchimento de preferências de categorias de produtos.	Indeterminado.	Baseado nas informações do cadastro será enviado periodicamente uma lista de sugestão de compra de produto.
Alerta de preço	Indeterminado. O usuário faz um cadastro no sistema do estabelecimento virtual. Durante o cadastro, ou após, é feito um preenchimento de preferências de preços.	Indeterminado.	Baseado nas informações do cadastro será enviado um aviso, ao usuário, de que o produto se encontra com preço igual ou menor ao mínimo estabelecido.

Na tabela a seguir citamos a pergunta chave para a problemática de cada sistema de aconselhamento apresentado.

Tabela 4.16 - Pergunta chave de cada sistema de aconselhamento.

Sistema	Pergunta
Protótipo GAADT	Qual a melhor sugestão de compra ou venda de um produto eletroeletrônico?
DietSysNet	Quais os casos de tratamento de pacientes bem sucedidos que são mais similares com este novo paciente?
MovieGuide	Qual é a recomendação de filmes em DVD baseado em uma lista de filmes prediletos e outra como detestáveis?
NB Computers	Qual é o <i>rank</i> deste computador portátil?
Lojas virtuais	Qual é a recomendação do dia?
Alerta de preço	Quero um produto por x reais. Tem? Caso não, me avise quando tiver.

Apesar dos sistemas, estarem dentro do domínio de *e-commerce*, eles seguem propósitos parecidos com estratégias diferentes. De fato, encontrar um sistema de recomendação que tenha um propósito idêntico, contemplando as mesmas estratégias, dados de entrada e informando o desempenho de processamento dos dados, é uma tarefa árdua. Contudo, dos apresentados nesta seção podemos citar que:

- MovieGuide: Foi o que mais se aproximou do protótipo GAADT. Pois, apesar de não ter propósito idêntico, usou AG como mecanismo de decisão e um histórico de situações reais fornecidos pelo usuário para recomendação de filmes em DVD. Contudo, não é informado o tempo de processamento dos dados. Faz uso de um algoritmo supervisionado, pois precisa de um modelo prévio de dados para calcular a saída;
- NB Computers: Tem o mesmo propósito, voltado para um único tipo de produto. Não informa o tempo de processamento dos dados. É supervisionado, pois precisa de um modelo prévio de dados para calcular a saída. O usuário intervém diretamente na saída gerada;
- DietSysNet: Apesar de não ter propósito idêntico, faz recomendação no domínio da especialidade médica de nutrição, baseado em conjunto de modelos de pacientes bem sucedidos para comparar com o novo paciente. Tem uma dependência do usuário para corrigir a saída, adaptando para futuras consultas;
- Lojas virtuais: Não têm um volume de dados definido e nem um tempo de resposta definido. É imposto pela loja virtual e está associado com a campanha

da empresa e não, necessariamente, com uma estimaco para se calcular qual   o melhor produto;

- Alerta de preos: Semelhante   loja virtual, com a diferena que a imposio   feita pelo usu rio. O usu rio   quem decide o que   necess rio para que o produto seja o melhor.

Na Tabela 4.17 temos o resumo da comparao do prot tipo GAADT com os sistemas de recomendao mencionados nesta seo.

Tabela 4.17 - Comparao do prot tipo GAADT com outros sistemas.

Sistema	Volume de dados testado	Tempo de resposta	Supervisionado	Necessidade de ajustes externos
Prot�tipo GAADT	Mais de 50000 produtos.	M�ximo de ~6h. M�nimo ~1s.	N�o	N�o
DietSysNet	Aproximadamente, 60 pacientes.	Elevado.	Sim	Sim
MovieGuide	100 filmes.	N�o informado.	Sim	N�o
NB Computers	N�o informado.	N�o informado.	Sim	Sim
Lojas virtuais	Indeterminado.	Indeterminado.	Misto.	N�o
Alerta de preo	Indeterminado.	Indeterminado.	Sim	Sim

O prot tipo GAADT apresentou um desempenho satisfat rio, perante os demais sistemas, uma vez que com o mesmo volume de dados usados nos demais sistema, o prot tipo consegue executar em menos de 2s com uma configurao b sica de *hardware*. Al m disso, alguns autores chegam a citar que   necess rio ter um grande n mero de interao para obter uma resposta (Wei, et al., 2002) ou que houve problemas graves de desempenho do sistema, referentes ao tempo de processamento e resposta (Telles, et al., 2006).

Capítulo 5

Conclusões

A manipulação de dados usando sistemas de apoio à decisão que se baseiam em acontecimentos ou experiências passadas pode ser considerada uma estratégia intuitivamente atraente porque é semelhante ao comportamento de solução de problemas humanos. As pessoas utilizam experiências passadas para solucionar problemas novos e esta aproximação se torna conveniente, efetiva e frequentemente, suaviza a carga de análise de um domínio específico. Isto conduz à vantagem de que um sistema inteligente pode estar baseado em conhecimento superficial e não requer esforço significativo em conhecimento criado.

5.1 Contribuições e Relevâncias

Neste trabalho, foi apresentado um algoritmo genético capaz de melhorar o tempo de processamento de uma base de dados dentro de um sistema de apoio à decisão. No qual foi testado em um exemplo domínio de compras on-line que originalmente contém mais de 10 mil vendas cadastradas e o modelo apresentado convergiu para 850 registros, aproximadamente. Com os exemplos citados nas seções anteriores, foi possível ilustrar a redução da base de dados sem perda de informação e descrever um modelo relacional que atenda a maioria das representações e mecanismos de um sistema evolutivo.

Decorrente deste estudo desenvolveu-se um algoritmo capaz de gerar novas tuplas com cardinalidade menor que as tuplas atuais para diminuir o espaço da base de dados e aumentar a eficiência da tomada de decisão sem perda de informação dentro de um domínio de compras on-line.

Mesmo o protótipo ainda não tendo alcançado todo o potencial de espaço proposto pelo GAADT, é importante ressaltar que é possível concluir que o GAADT consegue resolver a tarefa de aconselhamento de compra de produtos eletroeletrônicos com certa regularidade quando a escolha dos dados obedece ao perfil de compra de um usuário. E ainda, acreditamos que em um computador com maiores recursos computacional, os resultados seriam superiores aos que foram apresentados na seção 4.3.

A modelagem, implementação e testes gerados por essa dissertação foram submetidos para análise de pesquisadores, principalmente, nas áreas de inteligência computacional para modelagem, controle e automação, aprendizado de máquina e análise de dados, computação evolutiva, sistemas inteligentes e em informática em saúde, através das seguintes publicações:

- Representação de Casos: Um modelo entidade relacionamento para negociação automatizada (Santos, et al., 2007);
- Uso de algoritmos genéticos através de tipos abstratos de dados para manutenção da base de casos dentro do estudo do raciocínio baseado em casos (Santos, et al., 2007);
- Sistema Evolutivo para Aconselhamento de Aquisição de Material Hospitalar (Santos, et al., 2008);
- Maintenance of the base of cases of an E-commerce system through Genetic Algorithm (Santos, et al., 2008);
- Evolutionary system for advice of acquisition of equipment pertaining to a hospital (Santos, et al., 2008);
- An Evolutionary Computing Model to Implement Decision Support Systems for Buying Electronics (Santos, et al., 2009).

5.2 Dificuldades Encontradas

Por se tratar de uma técnica nova e sucessora de várias técnicas existentes, a principal dificuldade encontrada foi de encontrar artigos escritos por pessoas com experiência prática para citar exemplos reais e, com isso, ter dados para justificar ou argumentar a necessidade do uso dessas técnicas nas empresas.

Outra dificuldade foi a de conciliar o tempo de pesquisa do mestrado com emprego e família. Uma vez que se leva tempo para busca, análise e atualização das referências teórica

através de artigos de congresso e seminários, livros acadêmicos e de soluções práticas. Além disso, também a coleta de dados para testes no protótipo e em outros sistemas.

O tempo e a qualidade dos testes nos demais sistemas de aconselhamento também poderiam ser melhores e mais preciso. Contudo, alguns exemplos não estavam acessíveis. Por exemplo, em (Wei, et al., 2002) chega a citar a URL *www.allmovieguide.com* para testes do sistema, contudo a página principal informa acesso negado.

Ainda, durante a realização do trabalho, foram notadas as diferentes padronizações de busca, recepção e formato dos dados através dos sites de buscas de produtos e serviços, dificultando o desenvolvimento do protótipo.

5.3 Trabalhos Futuros

Primeiramente, planejamos criar um sistema evolutivo multiagentes contemplando as principais teorias que envolvam a linha de pesquisa de descoberta de conhecimento para automatizar processos de negociação. Agregar o modelo aqui apresentado com outros modelos, como redes SOM e RBC, verificando o desempenho dos processos e fazendo uma análise estatística mais detalhada dos modelos aqui apresentados. Paralelo a essas tarefas, escrever artigos ilustrando o funcionamento do software e o comportamento do mesmo com essas abordagens.

Uma avaliação do protótipo sendo executado em *cluster* e comparando com outros sistemas nas mesmas condições, avaliando uso do processador, memória, tempo de resposta e ponto crítico.

A criação de um sistema *Web*, modelado em UML, oferecendo serviços de compactação de base de dados e apoio à decisão a partir das informações de sites de buscas de produtos e serviços, seria também um bom estudo para um trabalho futuro.

Apêndice A

Análise do Sistema de Apoio à Compra Eletrônica

Nesta fase de análise, utilizamos as técnicas de UML. Neste tópico, veremos os casos de uso essenciais, diagramas de caso de uso, o modelo conceitual, diagrama de sequência e diagrama de estado.

A.1 Casos de Uso

Abaixo estão relacionados os casos de uso essenciais utilizados na modelagem da fase de análise do sistema:

I- Caso de uso: Solicitar extração de dados;

Atores: Solicitante, Administrador;

Descrição: O solicitante seleciona o tipo de importação de dados: através da *Web* ou arquivo XML. Se a opção for através da *Web*, o sistema fornecerá um ambiente interativo de busca para escolha dos produtos. Se a opção for através de arquivo XML, o sistema fornecerá um ambiente para escolha do arquivo XML. Em ambas as opções, o sistema avalia os dados (se os dados a extrair não forem consistentes, o sistema irá acusar e informar ao administrador). Quando concluído, o sistema comunica ao solicitante a quantidade de dados extraídos.

II- Caso de uso: Solicitar recomendação;

Atores: Solicitante, Administrador;

Descrição: O solicitante comunica de qual categoria de produtos pretende solicitar a sugestão de compra. O sistema avalia os dados da categoria (se os dados forem suficientes para tomada de decisão, o sistema irá dar início a processo de geração de sugestões de compras). Quando concluído, o sistema comunica ao solicitante as sugestões geradas, efetua *backup* e registra as sugestões.

III- Caso de uso: Retirar relatório;

Atores: Administrador;

Descrição: O administrador solicita a impressão de um relatório. O sistema pede o tipo do relatório. Se o tipo informado for válido a impressão será aceita e o administrador terá acesso para visualizar e/ou imprimir um relatório, caso contrário o sistema comunicará a falha de acesso.

IV- Caso de uso: Cadastrar experiência;

Atores: Administrador;

Descrição: O administrador solicita o cadastro (ou modificação) de uma experiência de compra. O sistema pede as novas informações. Se as informações tratarem de uma nova experiência, ela será inserida na base, caso contrário será editada.

V- Caso de uso: Cadastrar categoria;

Atores: Administrador;

Descrição: O administrador solicita o cadastro (ou modificação) de uma categoria de compra. O sistema pede as novas informações. Se as informações tratarem de uma nova categoria ela será inserida na base, caso contrário será editada.

VI- Caso de uso: Solicitar manutenção;

Atores: Administrador;

Descrição: O administrador solicita manutenção na base. O sistema avalia os dados (se os dados apresentar redundância ou falta de consistência, o sistema irá fazer a manutenção). Quando concluído, o sistema comunica ao administrador se houve ou não a manutenção e registra a solicitação.

Abaixo, na Figura A.1, é apresentado diagrama de caso de uso utilizado na modelagem na fase de análise do sistema.

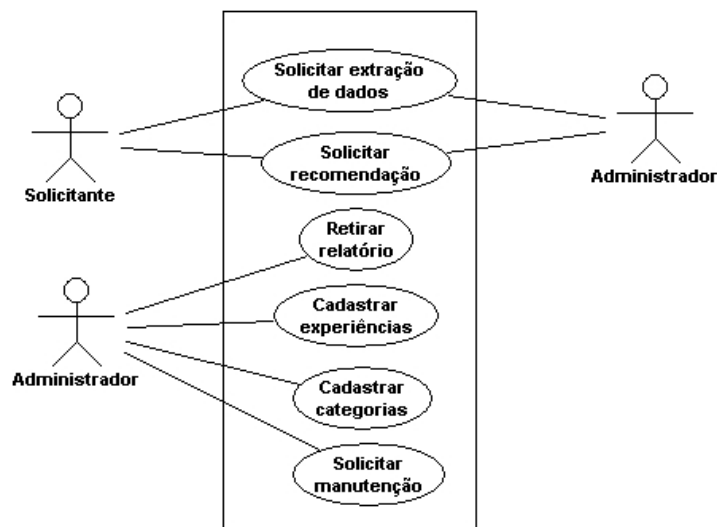


Figura A.1 - Diagrama de caso de uso do sistema.

A.2 Esquema Conceitual

A Figura A.2, abaixo, apresenta o esquema conceitual utilizado na modelagem na fase de análise do sistema.

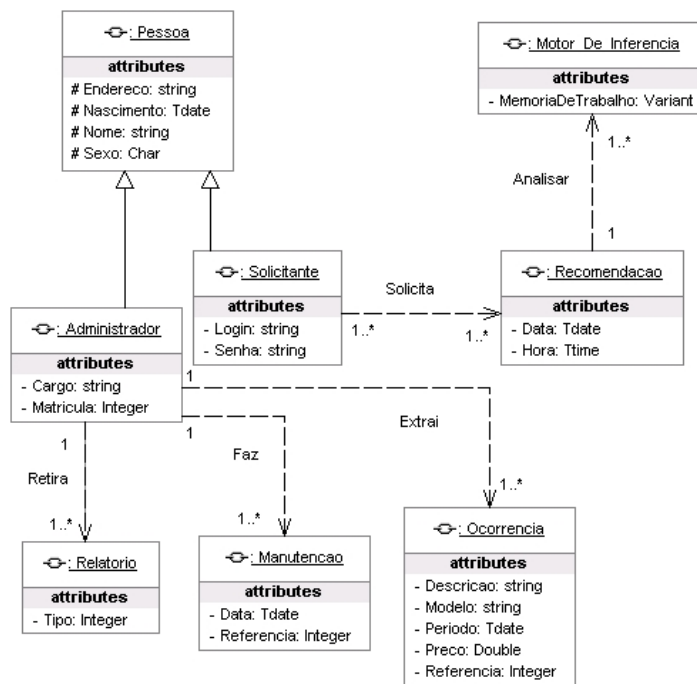


Figura A.2 - Esquema conceitual do sistema.

A Figura A.2 apresenta uma visão de alto nível do banco, descrevendo como os dados estão organizados do ponto de vista lógico, além de um conjunto de restrições de integridade. Os principais conjuntos de estrutura de dados que consiste o esquema conceitual da Figura A.2 são referentes ao cadastro de pessoas no sistema que poderá ser um solicitante do atendimento ou um administrador para supervisionar e gerenciar o sistema. Também consta a análise das experiências de compra ou venda realizadas fornecidas e o gerenciamento das decisões tomadas.

A.3 Diagrama de Sequência

Abaixo estão relacionados os principais diagramas de sequência utilizados na modelagem na fase de análise do sistema.

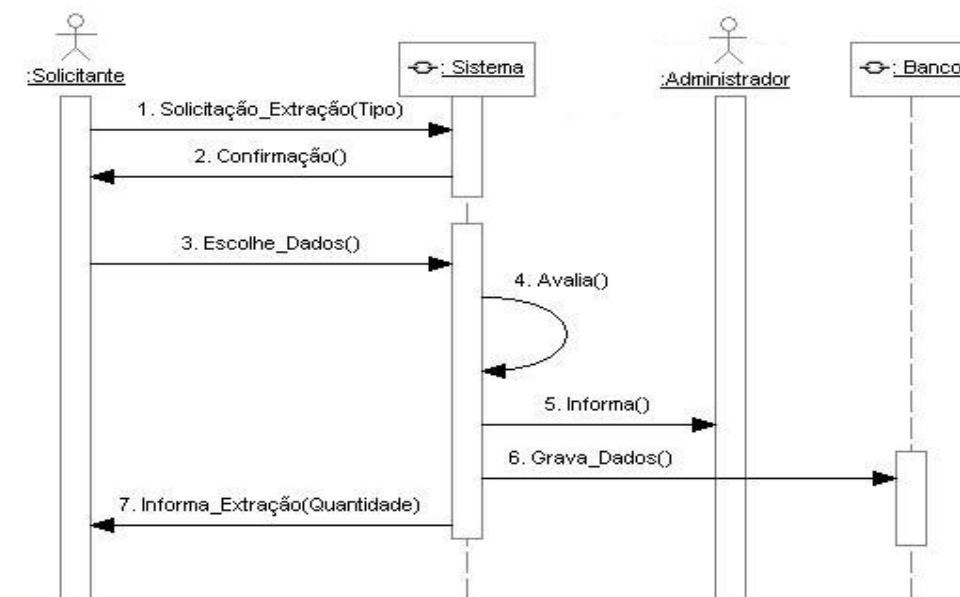


Figura A.3- Diagrama de sequência solicita extração.

Para o diagrama de sequência da Figura A.3, o solicitante seleciona o tipo de importação de dados que poderá ser via *Web* ou arquivo XML. Após a confirmação da opção escolhida o sistema solicita a escolha dos dados. Os dados fornecidos serão avaliados, onde se os dados não forem consistentes o sistema irá acusar e informar ao administrador. Havendo a extração, os dados serão armazenados e sem seguida a quantidade de dados extraídos será informada ao solicitante.

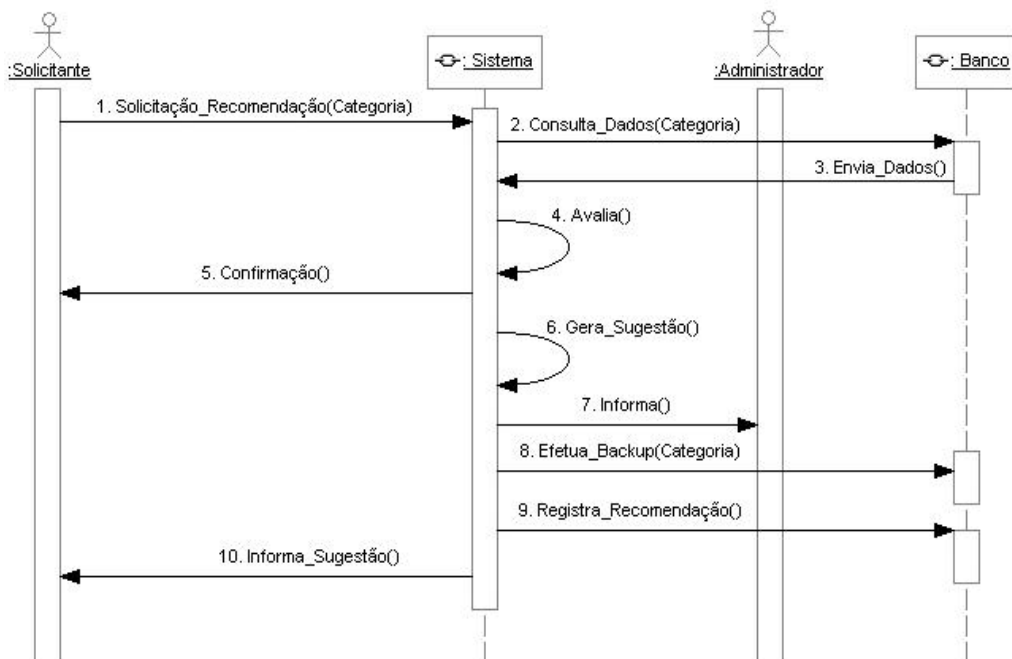


Figura A.4 - Diagrama de sequência solicita recomendação.

Para o diagrama de sequência da Figura A.4, o solicitante escolhe a categoria de produtos que pretende solicitar a sugestão de compra. O sistema faz uma consulta dos dados armazenados no banco e verifica se os dados são suficientes para tomada de decisão. O sistema comunica ao solicitante as sugestões geradas, gera um *backup* e registra as sugestões.

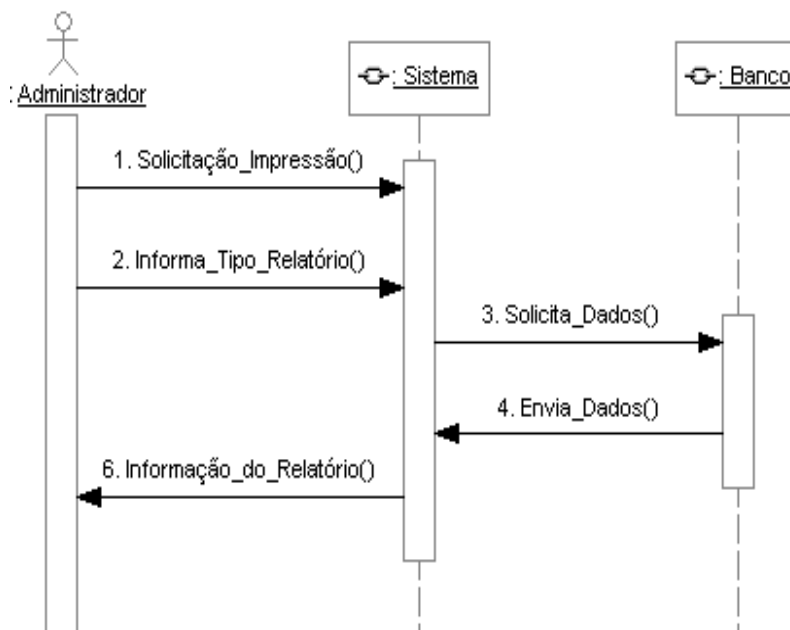


Figura A.5 - Diagrama de sequência retira relatório.

Para o diagrama de sequência da Figura A.5, o administrador solicita a impressão de um relatório ao sistema. Logo após deverá ser informado o tipo do relatório. O sistema irá efetuar a busca dos dados para o relatório solicitado no banco. Em seguida, o administrador terá acesso para visualizar e/ou imprimir um relatório.

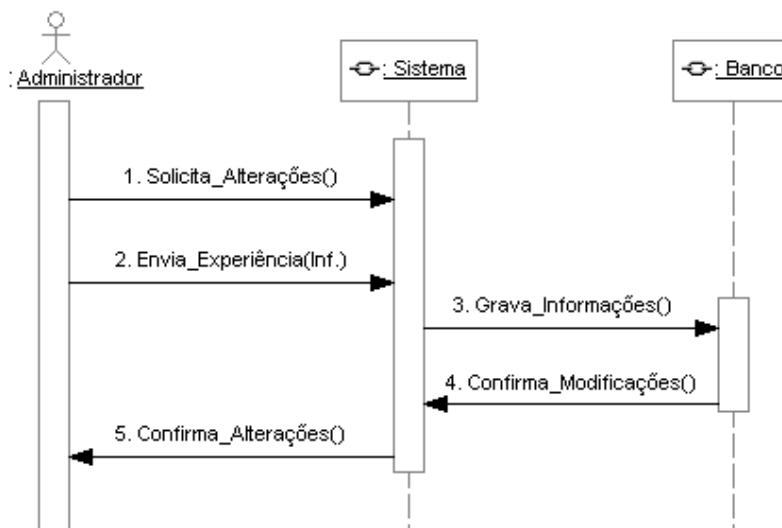


Figura A.6 - Diagrama de sequência cadastra experiência.

Para o diagrama de sequência da Figura A.6, o administrador solicita uma modificação de dados. Fornece as informações a ser enviado para o sistema. As informações serão gravadas no banco e uma confirmação da transação será apresentada.

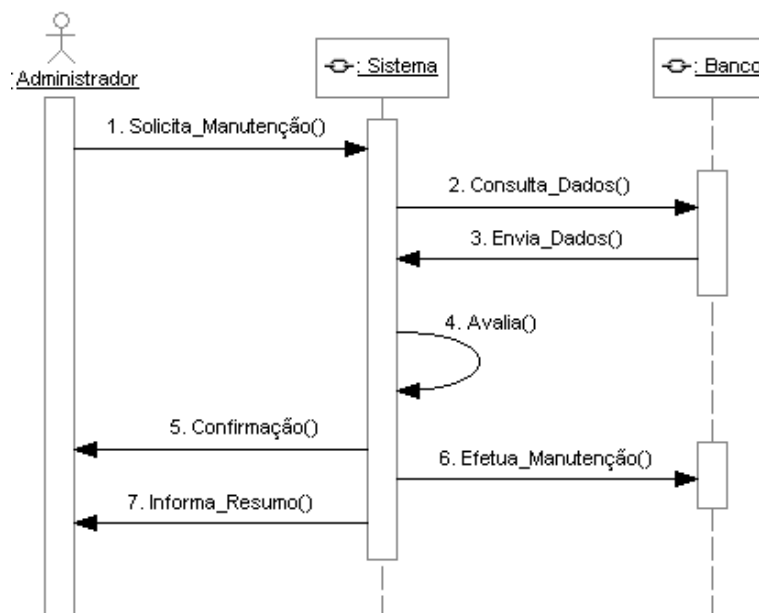


Figura A.7 - Diagrama de sequência solicita manutenção.

Para o diagrama de sequência da Figura A.7, o administrador solicita manutenção na base de conhecimento do sistema. O sistema realiza uma consulta dos dados no banco, e avalia os dados, retornando um resumo para o administrador para que possa confirmar a inicialização da manutenção.

A.4 Diagrama de Estados

Abaixo estão relacionados os principais diagramas de estados utilizados na modelagem na fase de análise do sistema.

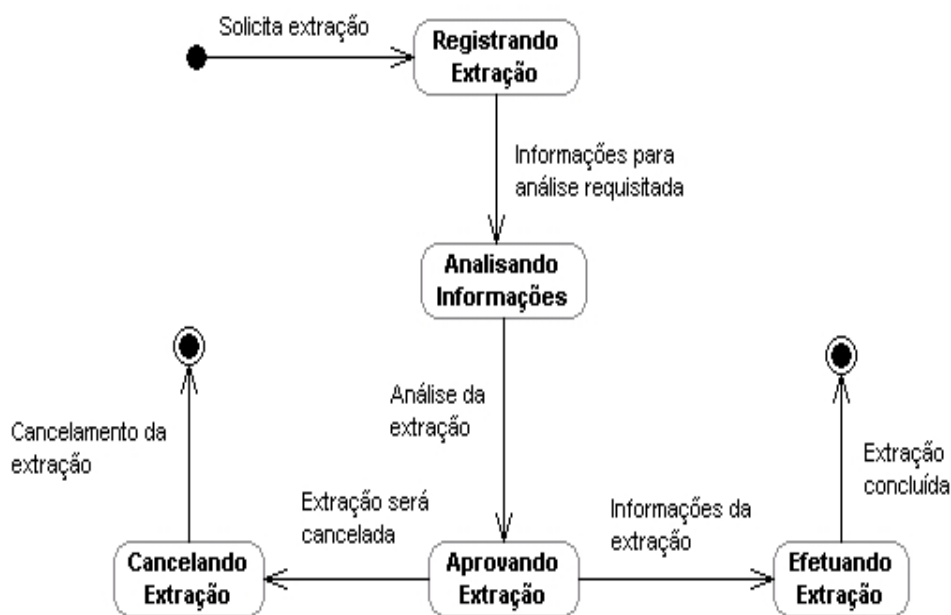


Figura A.8- Diagrama de estados para a classe extração.

Para o diagrama de estados da Figura A.8, o primeiro estado é referente a inicialização do solicitante no sistema, onde será informado o tipo de importação dos dados. O estado seguinte faz a análise das informações oriundas do arquivo ou *Web*. O próximo estado é referente a aprovação da execução da extração. Caso haja algum problema na aprovação da extração, o estado a ser executado é do cancelamento da extração e em seguida o estado final. Do contrário, será executado o estado para efetuar a extração seguida do estado final.

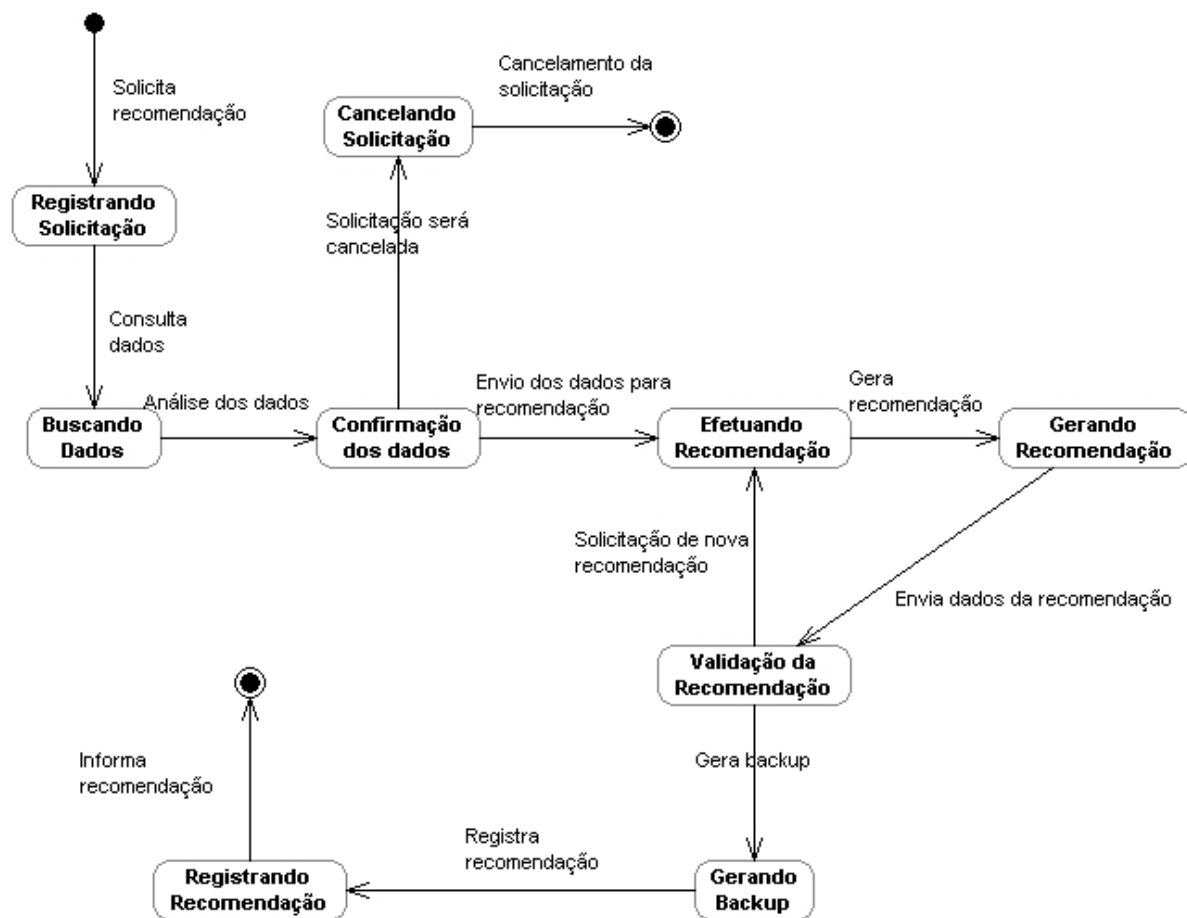


Figura A.9 - Diagrama de estados para a classe recomendação.

Para o diagrama de estados da Figura A.9, o primeiro estado é referente a inicialização do solicitante no uso do sistema, onde será informado a categoria de produtos que pretende solicitar a sugestão de compra. O estado seguinte faz a busca dos dados da categoria escolhida. O próximo estado fornece um resumo dos dados e solicita uma confirmação. Caso informação passada seja a desistência da operação, o estado a ser executado é do cancelamento da operação de recomendação no sistema e em seguida o estado final. Do contrário, o ciclo formado pelos estados de efetuação, geração e validação de recomendação será executado várias vezes até verificar se os dados são suficientes para tomada de decisão. Ao término deste ciclo, próximo estado gera um *backup*. O próximo estado registra as sugestões, chegando ao estado final.

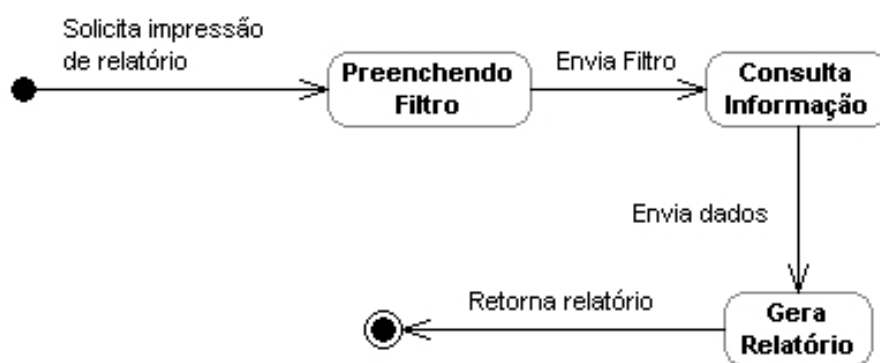


Figura A.10 - Diagrama de estados para a classe relatório.

Para o diagrama de estados da Figura A.10, o primeiro estado é referente a inicialização do administrador no uso do sistema quanto a retirada do relatório. O estado seguinte efetua uma consulta no banco de acordo com os dados da solicitação. O próximo estado gera o relatório proporcionando a visualização ou impressão e em seguida o estado final.

Apêndice B

Fase de projeto do Sistema

Nesta seção, iremos apresentar as principais funções utilizadas no sistema, casos de uso reais, arquitetura do sistema, diagrama de classe e o diagrama de implementação.

B.1 Funções Básicas do Sistema

A Tabela B.1, abaixo, lista as principais funções do sistema utilizadas na modelagem na fase de projeto do sistema.

Tabela B.1 - Funções básicas.

Referência	Função	Categoria
1.1	Solicita extração.	Evidente
1.2	Escolhe dados.	Evidente
1.3	Valida dados.	Oculto
1.4	Grava dados.	Oculto
1.5	Informa quantidade de dados extraídos.	Evidente
2.1	Solicita recomendação.	Evidente
2.2	Avalia dados da categoria.	Oculto
2.3	Gera sugestão.	Oculto
2.4	Informa recomendação.	Evidente
3.1	Solicita relatório.	Evidente
3.2	Recebe relatório.	Evidente
4.1	Solicita manutenção.	Evidente
4.2	Avalia dados da base.	Oculto
4.3	Faz manutenção.	Oculto
5.1	Efetua login.	Evidente
6.1	Efetua <i>backup</i> .	Oculto

Onde:

- *Evidente* são as funções que o ator ou usuário podem observar. Funções que estão visíveis ao usuário.
- *Oculto* são as funções que o ator ou usuário não podem observar. Funções que são de processo interno, exclusivo ao sistema.

B.2 Casos de Uso Reais

Abaixo estão relacionados os principais casos de uso reais utilizados na modelagem na fase de projeto do sistema.

I- Caso de uso: Solicitar extração de dados;

Atores: Solicitante (iniciador), Administrador;

Propósito: Capturar uma importação, oriundo de um solicitante, e avaliar as informações por ele dado;

Referência: 1.1, 1.2, 1.3, 1.4, 1.5, 5.1, 6.1;

Sequência de eventos:

Ação do ator

Resposta-sistema

1. O solicitante efetua *login* e solicita importação;

2. O sistema pede fonte de origem dos dados;

3. O solicitante informa fonte;

4. O sistema valida as informações;

5. Se os dados foram aceitos, então o sistema efetua o *backup* e logo após ativa importação dos dados;

6. O administrador é informado da importação.

Podendo o administrador restaurar o banco para seu estado inicial;

7. O solicitante visualiza uma confirmação do sistema.

II- Caso de uso: Solicitar recomendação;

Atores: Solicitante(iniciador), Administrador;

Propósito: Efetuar um auxílio para escolha de um produto, a partir das informações armazenadas na base de dados;

Referência: 2.1, 2.2, 2.3, 2.4, 6.1;

Sequência de eventos:

Ação do ator

Resposta-sistema

1. O solicitante requer sugestão;

2. O sistema solicita escolha da categoria;

3. O solicitante informa categoria;

4. O sistema valida as informações;

5. Se os dados foram aceitos, então o sistema efetua a solicitação e logo após repassa um lista de sugestão;

6. O administrador é informado da solicitação;

7. O solicitante visualiza uma confirmação do sistema;

8. O sistema substitui os dados antigos pelos novos dados da lista gerada, realizando antes um *backup*.

III- Caso de uso: Retirar relatório;

Atores: Administrador(iniciador);

Referência: 3.1, 3.2, 5.1;

Propósito: Imprimir relatórios de documentação (declaração, lista de produtos, categorias, experiências etc.) e pesquisa (estatística);

Sequência de eventos:

Ação do ator

Resposta-sistema

1. O administrador chega ao terminal que tenha acesso ao recurso de Impressão de relatório;

2. O administrador aciona o recurso de retirar relatório;

3. O sistema valida as informações e se elas forem válidas ele terá acesso a visualização e impressão de relatórios;

4. O administrador visualiza uma mensagem de confirmação positiva ou negativa da operação. Este último o administrador deverá ler atentamente a mensagem e corrigir as informações, repetindo novamente a opção de relatório.

B.3 Arquitetura do Sistema

Um esboço da arquitetura do sistema é ilustrado na Figura B.1 e está representada por três camadas. Nas camadas, é possível visualizar a principal organização do sistema, a seleção dos elementos estruturais e suas interfaces que compõem o sistema e seus comportamentos.

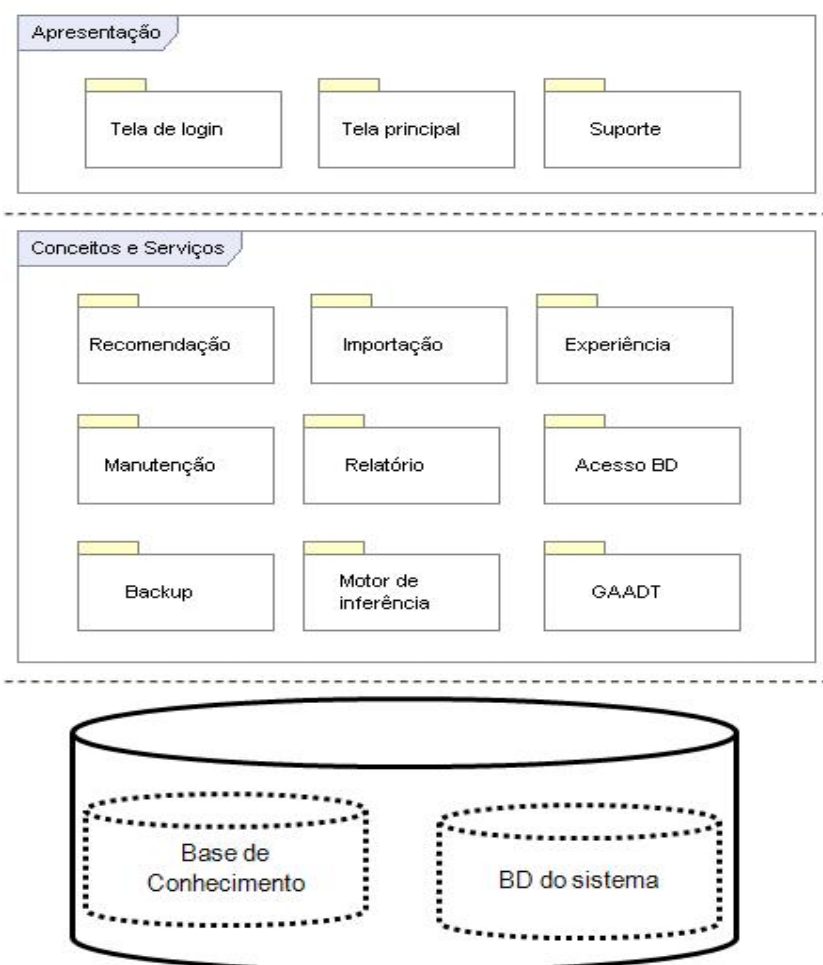


Figura B.1 - Arquitetura do sistema.

A primeira camada, camada de interface, é responsável pela interação com o usuário, nela encontramos as principais interações. Na segunda camada, camada de comportamento, temos os principais comportamentos do sistema. E por fim, a terceira camada, camada de armazenamento, é responsável por guardar as informações durante todo funcionamento do sistema, nela estará armazenado o histórico das consultas e experiências de compra e venda ocorrida dentro do sistema.

B.4 Diagrama de Colaboração

Abaixo estão relacionados os principais diagramas de colaboração utilizados na modelagem na fase de projeto do sistema.

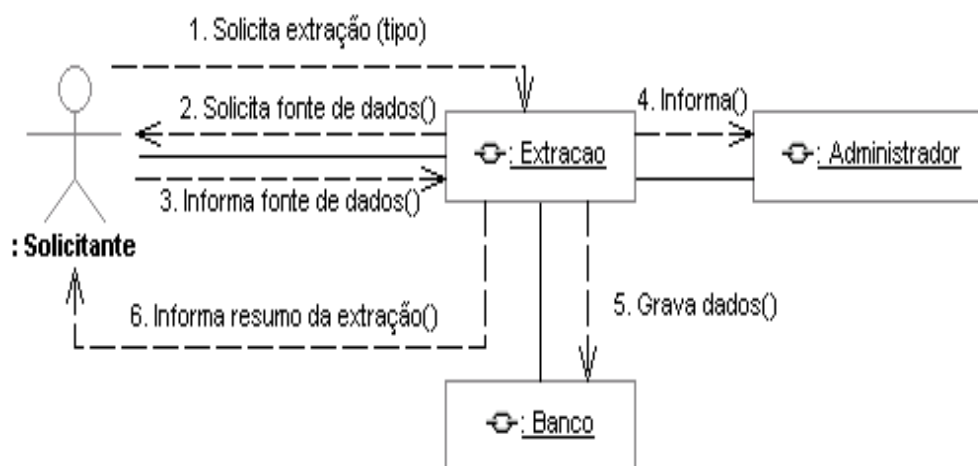


Figura B.2 - Diagrama de colaboração de uma solicitação de extração.

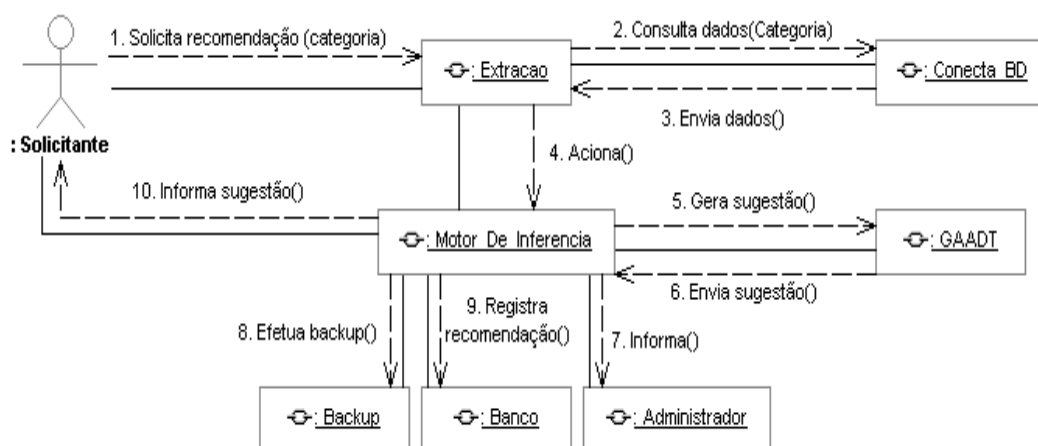


Figura B.3 - Diagrama de colaboração de uma solicitação de recomendação.

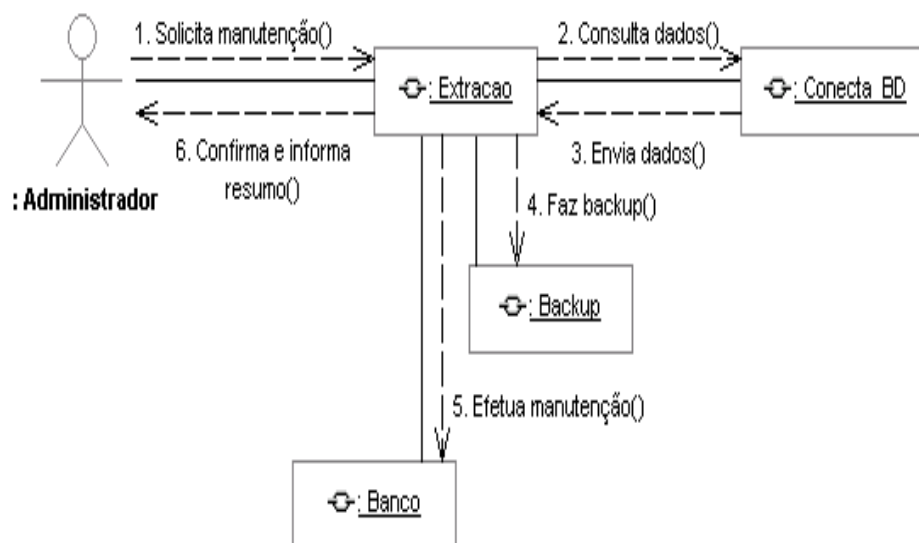


Figura B.4 - Diagrama de colaboração de uma solicitação de manutenção.

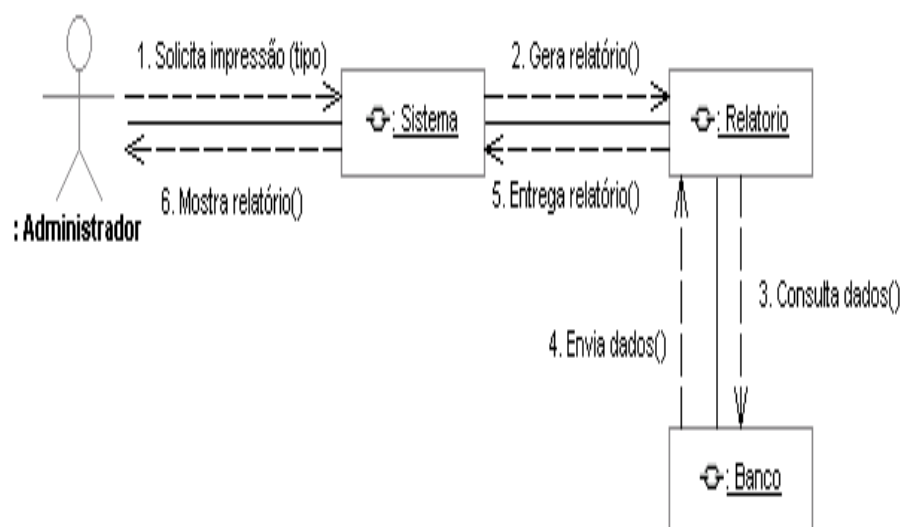


Figura B.5 - Diagrama de colaboração de uma impressão de relatório.

B.5 Diagrama de Classe

A Figura B.6, a seguir, apresenta o diagrama de classe utilizado na modelagem na fase de projeto do sistema.

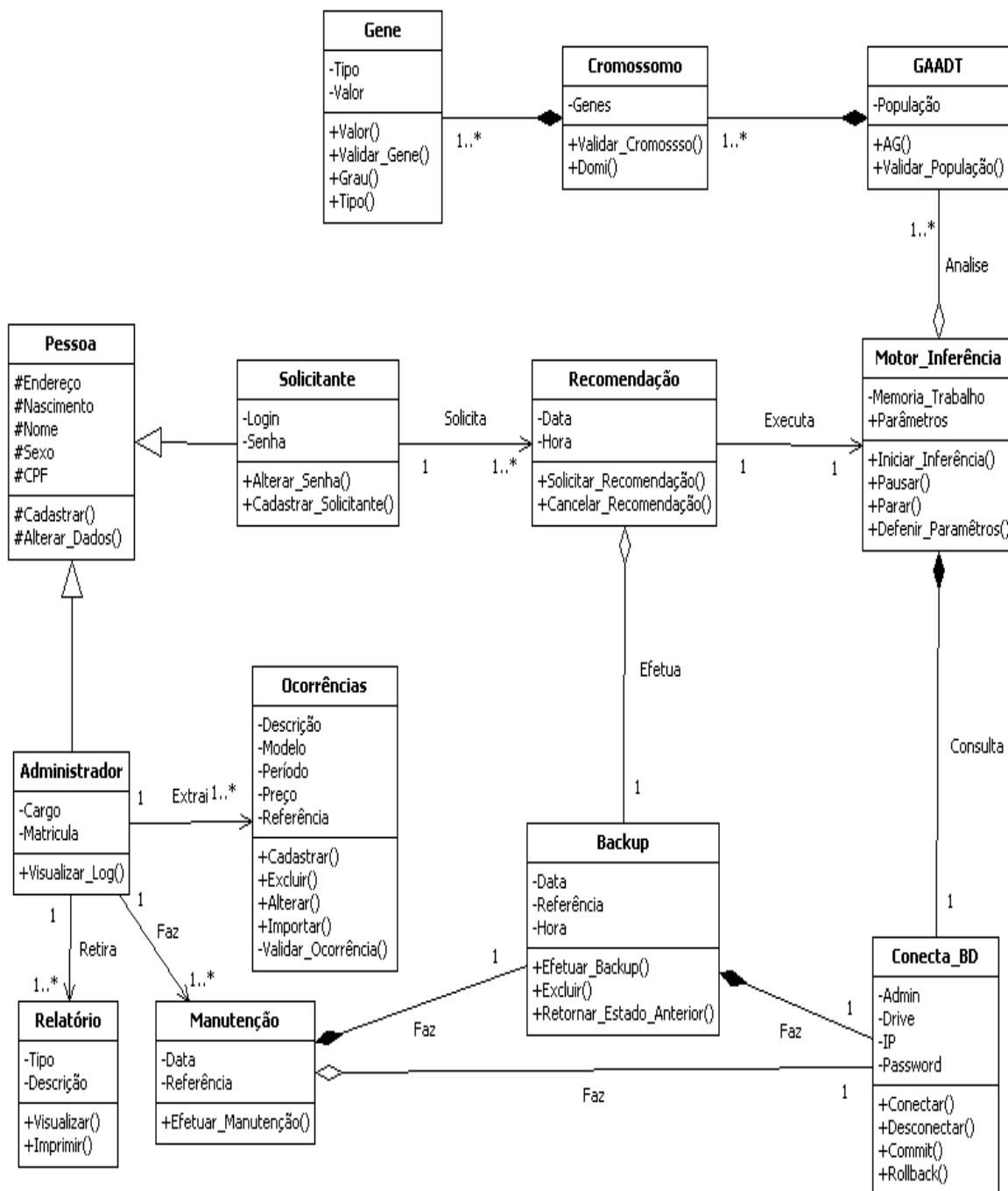


Figura B.6 - Diagrama de classe do sistema.

B.6 Diagrama de Implantação

A Figura B.7, a seguir, apresenta o diagrama de implantação utilizado na modelagem na fase de projeto do sistema.

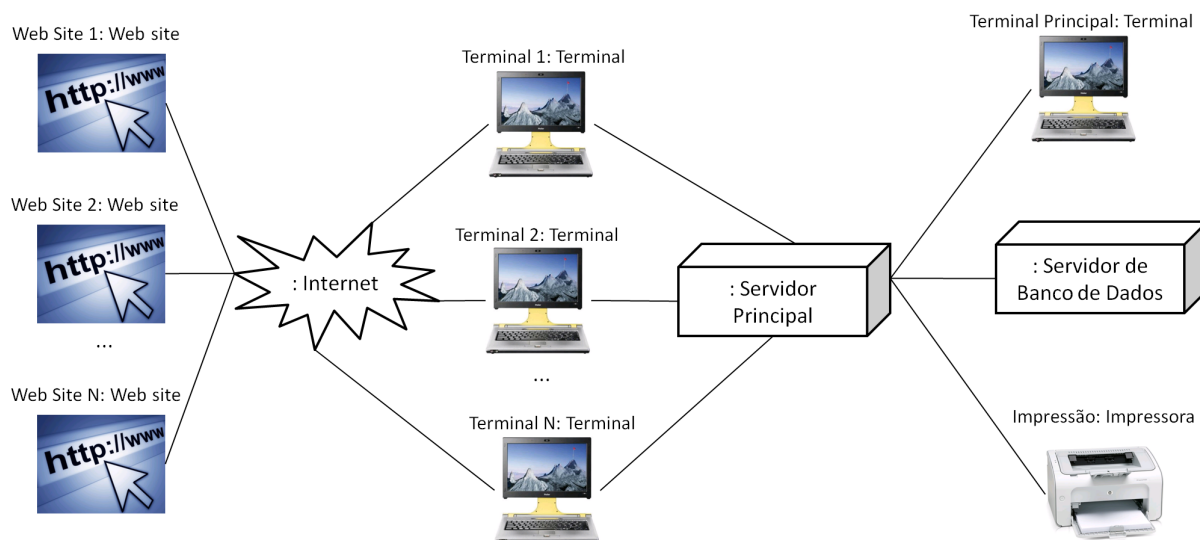


Figura B.7 - Diagrama de implantação do sistema.

Referências Bibliográficas

- Ahn Hyunchul, Kim Kyoung-jae e Han Ingoo. Hybrid genetic algorithms and case-based reasoning systems for customer classification [Periódico]. - [s.l.] : Expert Systems, 2006. - 3 : Vol. 23.
- Arruda E. F. e Kagan N. Estimação de Estados Harmônicos usando Estratégias Evolutivas [Periódico]. - [s.l.] : Revista Controle & Automação, Abril de 2009. - 2 : Vol. 20. - pp. 177-191.
- Barcellos Carla D. [et al.] Sistema de Recomendação Acadêmico para Apoio a Aprendizagem [Artigo]. - 2007.
- Bertino Elisa, Catania Barbara e Zarri Gian P. Intelligent Database Systems [Livro]. - Great Britain : Addison-Wesley, 2001. - 0201877368.
- Bichindaritz Isabelle, Montani Stefania e Portinale Luigi. Special issue on case-based reasoning in the health sciences [Periódico]. - USA : Springer, 2007. - Vols. pg. 207-209.
- Booch Grady, Rumbaugh James Rumbaugh e Jacobson Ivar. UML - Guia do Usuário [Livro] = The Unified Modeling Language User Guide / trad. Silva Fábio F.. - Rio de Janeiro : Campus, 2000. - 8535217843.
- Braga Antônio P, Ludermir Teresa e Carvalho André. Redes Neurais Artificiais: Teorias e aplicações [Livro]. - Rio de Janeiro : LTC, 2000. - 1ª.
- Cortez Renato Planejando no E-Commerce [http://www.pensandomarketing.com/home/id99.html] = Artigos sobre Marketing e Negócios / prod. Dutra Marcos. - Belém : Pensando Marketing, Junho de 2004.
- Costin Bădică, Maria Ganzha e Marcin Paprzycki. Developing a Model Agent-based E-Commerce System [Livro]. - Berlin : Springer, 2007. - Vol. 37.
- Elmasri Ramez e Navathe Shamkant. Fundamentals Of Database Systems [Livro]. - Vancouver : Addison Wesley, 2000.

- Fonseca Carlos M e Fleming Peter J. Multiobjective Optimization and Multiple Constraint Handling with Evolutionary Algorithms - Part I: A Unified Formulation [Artigo] // IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS - PART A: SYSTEMS AND HUMANS. - 1998. - Vol. 28. - 1083–4427/98\$10.00.
- Furlan José D. Modelagem de Objetos Através da UML [Livro]. - São Paulo : Makron books, 1998. - 8534609241.
- Furtado Danielle. A Genetic Algorithm Based On Abstract Data Types to the Multiple Protein Alignment Problem [Artigo] // Encontro de Modelagem Computacional. - Encontro de Modelagem Computacional : [s.n.], 2006. - IX.
- Haykin Simon. Redes Neurais: Princípios e prática [Livro]. - Porto Alegre : Bookman, 2001.
- Kovacs Zsolt L. Redes Neurais Artificiais: Fundamentos e aplicações [Livro]. - São Paulo : Livraria da Física, 2002. - 3ª. - 8588325144.
- Kraus Sarit. Strategic negotiations in multiagent environments [Livro]. - Massachusetts : Bradford, 2001.
- Lucas Henry C. Information Systems Concepts for Management [Livro]. - New York : McGraw Hill Higher Education, 1990.
- Maia L. C. e Bianchi R. A. Usando Programação Genética para Evoluir Agentes Jogadores de Futebol de Robôs [Periódico] // Pesquisa & Tecnologia. - São Bernardo do Campo : [s.n.], Outubro de 2001. - 21. - pp. 3-11.
- Minsky Marvin L. Semantic Information Processing [Livro]. - Cambridge : MIT Press, 1969.
- Montani Stefania. Exploring new roles for case-based reasoning in heterogeneous AI systems for medical decision support [Periódico]. - Italy : Springer Science, 2006. - Vols. 275-285.
- Nascimento Cairo L. Inteligência Artificial: Em controle e automação [Livro]. - São Paulo : Fapesp, 2000. - 8521202490.
- Object Data Management Group The Resource Portal Education and Research [Online] // Object Database Management Systems. - ODBMS.ORG, 1991. - 07 de 05 de 2009. - <http://www.odbms.org/ODMG/>.
- Pal Sankar e Shiu Simon. Foundations of Soft Case-Based Reasoning [Livro]. - New Jersey : Wiley, 2004.
- Pressman Roger S. Engenharia de software [Livro]. - São Paulo : Makkron Books, 1995. - 85-86804-57-6.

- Qi Xiaofeng e Palmieri Francesco. Theoretical Analysis of Evolutionary Algorithms With an Infinite Population Size in Continuous Space Part 11: Analysis of the Diversification Role of Crossover [Artigo] // TRANSACTIONS ON NEURAL NETWORKS. - [s.l.] : IEEE, 1994. - n° 1 : Vols. Vol 5, pg 120-128.
- Qi Xiaofeng e Palmieri Francesco. Theoretical Analysis of Evolutionary Algorithms With an Infinite Population Size in Continuous Space Part I: Basic Properties of Selection and Mutation [Artigo] // IEEE TRANSACTIONS ON NEURAL NETWORKS. - 1994. - Vol. 5. - 1045-9227/94\$04.00.
- Raymer Michael L. [et al.] Dimensionality Reduction Using Genetic Algorithms [Artigo] // IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION. - 2000. - Vol. 4.
- Rockart John F e Bullen Cristine V The Rise of Managerial Computing [Livro]. - [s.l.] : Dow Jones-Irwin, 1986.
- Rockett Rajeev Kumar and Peter. Multiobjective Genetic Algorithm Partitioning for Hierarchical Learning of High-Dimensional Pattern Spaces: A Learning-Follows-Decomposition Strategy [Artigo] // TRANSACTIONS ON NEURAL NETWORKS. - [s.l.] : IEEE, 1998. - ed. 5 : Vol. Vo. 9.
- Rudolph Günter. Convergence Analysis of Canonical Genetic Algorithms [Artigo] // IEEE TRANSACTIONS ON NEURAL NETWORKS. - 1994. - Vol. 5. - 1045-9227/94\$04.00.
- Russell Stuart J. e Norvig Peter. Inteligência Artificial [Livro] / trad. Souza Vandenberg D.. - Rio de Janeiro : Campus, 2004. - 853521177-2.
- Santos Genildo, Maciel Andrilene, Vieira Roberta, Costa Evandro e Coradine Luis. Maintenance of the base of cases of an E-commerce system through Genetic Algorithm [Artigo] // The Huntsville Simulation Conference 2008. - Alabama : Society for Modeling and Simulation, 2008.
- Santos Genildo, Vieira Roberta e Costa Evandro. Sistema Evolutivo para Aconselhamento de Aquisição de Material Hospitalar [Artigo] // Anais do XI Congresso Brasileiro de Informática em Saúde, CBIS'2008. - Campos do Jordão : Sociedade Brasileiro de Informática em Saúde - SBIS, 2008.
- Santos Genildo, Nascimento Hélio e Costa Evandro. Representação de Casos: Um modelo entidade relacionamento para negociação automatizada [Periódico] // Revista Digital. - Maceió : Publicar, 2007. - Vol. 1. - pp. 39-52. - 1983-2419.

- Santos Genildo, Vieira Roberta e Costa Evandro. An Evolutionary Computing Model to Implement Decision Support Systems for Buying Electronics [Artigo] // International Conference on Machine Learning and Data Analysis 2009 - ICMLDA'09. - San Francisco : International Association of Engineers - IAENG, 2009. - 978-988-17012-6-8.
- Santos Genildo, Maciel Andrilene, Vieira Roberta, Costa Evandro e Coradine Luis. Evolutionary system for advice of acquisition of equipment pertaining to a hospital [Artigo] // International Conference on Computational Intelligence for Modelling, Control and Automation - CIMCA08. - Vienna : School of Information Science & Engineering University of Canberra, 2008.
- Santos Genildo, Vieira Roberta e Costa Evandro. Uso de algoritmos genéticos através de tipos abstratos de dados para manutenção da base de casos dentro do estudo do raciocínio baseado em casos [Periódico] // Educação e Trabalho. - Maceió : [s.n.], 2007. - pp. 83-96. - 1806-5376.
- Sardinha José Alberto Rodrigues Pereira. MAS-School e ASYNC: Um Método e um Framework para Construção de Agentes Inteligentes // Tese de doutorado. - Rio de Janeiro : Pontifícia Universidade Católica do Rio de Janeiro, 2005. - 0115614/CA.
- Savory S E. Some Views on the State of Art in Artificial Intelligence [Artigo] // Artificial Intelligence and Expert Systems. - New York : John Wiley & Sons, 1988. - pp. 21-34.
- Schafer J. Ben, Konstan Joseph e Riedl John. Recommender systems in e-commerce [Artigo] // Proceedings of the 1st ACM conference on Electronic commerce. - New York : ACM Special Interest Group on Electronic Commerce , 1999. - pp. 158-166. - 1-58113-176-3.
- Schutzer Daniel. Artificial Intelligence - An Applications-Oriented Approach [Livro]. - New York : Van Nostrand Reinhold Company, 1987.
- Silver Gerald A. e Silver Myrna L. Systems Analysis and Design [Livro]. - Massachusetts : Addison Wesley Publishing Company, 1999. - 9780201066159.
- Souza Ricardo A e Lima André C. UML Aplicada: Da teoria à implementação. [Livro]. - Rio de Janeiro : Ciência Moderna Ltda, 2007. - 978-85-7393-594-3.
- Sprague Ralph H e Watson Hugh J. Sistema de apoio à decisão: Colocando a teoria em prática [Livro]. - Rio de Janeiro : Campus, 1991.

- Telles Dagoberto [et al.] Sistema de Raciocínio Baseado em Casos para Recomendação de Programa Alimentar [Periódico]. - [s.l.] : RESI - Revista Eletrônica de Sistemas de Informação, 2006. - 9 : Vol. 3.
- Tsai Chieh-Yuan e Chiu Chuang-Cheng. A case-based reasoning system for PCB principal process parameter identification [Artigo] // Expert Systems with Applications. - Taiwan : Elsevier, 2007. - 32. - Vols. pg 1183–1193.
- Turban Efraim e King David. Comércio Eletrônico - Estratégia e Gestão [Livro]. - São Paulo : Pearson Education do Brasil, 2004. - 85-87918-09-5.
- Vidal Antônio G R. Projeto e Desenvolvimento de Sistemas de Informação [Relatório] / FEA/USP. - São Paulo : [s.n.], 1996.
- Vieira Roberta Vilhena e Lopes Manoel Agamemnon. Uma Abordagem Teórica Inicial para os Algoritmos Genético através de tipos Abstratos de Dados [Relatório] : Relatório Técnico / Departamento de Informática ; Universidade Federal de Pernambuco. - Recife : [s.n.], 1999. - RT-D1/UFPE, N° 002/99.
- Vieira Roberta Vilhena. Um Algoritmo Genético Baseado em Tipos Abstratos de Dados e sua Especificação em Z // Tese de Doutorado. - Pernambuco : Departamento de Ciência de Computação da Universidade Federal de Pernambuco, 2003.
- Wanderley José Augusto. Negociação Total: Encontrando Soluções, Vencendo Resistências, Obtendo Resultados [Livro]. - São Paulo : Gente, 1998. - 857312168.
- Wangenheim Christiane Gresse e Wangenheim Aldo. Raciocínio Baseado em Casos [Livro]. - São Paulo : Manole, 2003.
- Wei Po Lee, Chih Hung Liu e Cheng Che Lu. Intelligent agent-based systems for personalized recommendations in Internet commerce [Periódico] // Expert System with Applications. - [s.l.] : Elsevier Science Ltd., 2002. - 22. - pp. 275-284. - PII: S0957-4174(02)00015-5.
- Zuben Fernando J. Von. Computação Evolutiva: Uma Abordagem Pragmática [Artigo] // Jornada de Estudos em Computação. - Piracicaba : [s.n.], 2000. - I : Vol. I. - pp. 25-45.