



**UNIVERSIDADE FEDERAL DA PARAÍBA – UFPB
CENTRO DE CIÊNCIAS EXATAS E DA NATUREZA
DEPARTAMENTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA**

**UMA ABORDAGEM SEMÂNTICA PARA SELEÇÃO DE ATRIBUTOS
NO PROCESSO DE KDD**

LAMARK DOS SANTOS RIBEIRO

**JOÃO PESSOA – PB
2010**

**UNIVERSIDADE FEDERAL DA PARAÍBA – UFPB
CENTRO DE CIÊNCIAS EXATAS E DA NATUREZA
DEPARTAMENTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA**

**UMA ABORDAGEM SEMÂNTICA PARA SELEÇÃO DE ATRIBUTOS
NO PROCESSO DE KDD**

LAMARK DOS SANTOS RIBEIRO

**JOÃO PESSOA – PB
2010**

LAMARK DOS SANTOS RIBEIRO

**UMA ABORDAGEM SEMÂNTICA PARA SELEÇÃO DE ATRIBUTOS
NO PROCESSO DE KDD**

Dissertação apresentada ao Centro de ciências exatas e da natureza da Universidade Federal da Paraíba, como requisito parcial para obtenção do título de Mestre em Informática (Sistemas de Computação).

Orientação: Profa. Dra. Valéria Gonçalves Soares
Coorientação: Profa. Dra. Natasha Queiroz

**JOÃO PESSOA – PB
2010**

Ata da Sessão Pública de Defesa de Dissertação de
Mestrado do LAMARK DOS SANTOS RIBEIRO,
candidato ao Título de Mestre em Informática na
Área de Sistemas de Computação, realizada em 27
de agosto de 2010.

6 Aos vinte e sete dias do mês de agosto do ano dois mil e dez, às treze horas, na Sala de
7 Reunião do Centro de Ciências Exatas e da Natureza da Universidade Federal da Paraíba,
8 reuniram-se os membros da Banca Examinadora constituída para examinar o candidato ao
9 grau de Mestre em Informática, na área de "*Sistemas de Computação*", na linha de pesquisa
10 "*Computação Distribuída*", o Sr. Lamark dos Santos Ribeiro. A comissão examinadora
11 composta pelos professores doutores: Valéria Gonçalves Soares (DI - UFPB), Primeiro
12 Orientador e Presidente da Banca Examinadora, Natasha Correia Queiroz Lino (DI -
13 UFPB), Segundo Orientador - Lucídio dos Anjos Formiga Cabral, (DI-UFPB), como
14 examinador interno e Rosalie Barreto Belian (UFPE), como examinador externo. Dando
15 início aos trabalhos, a Prof^ª. Valéria Gonçalves Soares, cumprimentou os presentes,
16 comunicou aos mesmos a finalidade da reunião e passou a palavra ao candidato para que o
17 mesmo fizesse, oralmente, a exposição do trabalho de dissertação intitulado "*Uma*
18 *Abordagem Semântica para Seleção de Atributos no Processo de KDD*". Concluída a
19 exposição, o candidato foi argüido pela Banca Examinadora que emitiu o seguinte parecer:
20 "aprovado" Assim sendo, deve a Universidade Federal da Paraíba expedir o respectivo
21 diploma de Mestre em Informática na forma da lei e, para constar, a professora Tatiana
22 Aires Tavares, Sra. Coordenadora do PPGI, lavrou a presente ata, que vai assinada por
23 ela, e pelos membros da Banca Examinadora. João Pessoa, 27 de agosto de 2010.

Tatiana Aires Tavares

Prof^ª. Dra. Valéria Gonçalves Soares
Primeiro Orientador (DI-UFPB)

Prof^ª. Dr^a. Natasha Correia Queiroz Lino
Segundo Orientador (DI-UFPB)

Prof. Dr. Lucídio dos Anjos Formiga Cabral
Examinador Interno (DI-UFPB)

Rosalie Barreto Belian
Examinador Externo (UFPE)

AGRADECIMENTOS

A Deus, pelas bênçãos e oportunidades concedidas, além das forças para que pudesse conquistar mais uma vitória em minha vida;

Aos meus pais, Virgílio e Eliane, por quem tenho profunda admiração, por terem dado toda a força necessária para a conclusão desta etapa de minha vida;

A minha filha Mayra, a razão das minhas alegrias e maior motivação na minha vida;

A Francys, pelos bons momentos;

Aos meus irmãos Samara e Darwin, que sempre colaboraram em todos os momentos;

A minha tia Fátima, sem você não teria conseguido. Essa conquista também é sua;

Ao meu amigo Klebber, obrigado pelos conselhos e apoio sempre que precisei;

As minhas orientadoras, professora Valéria e Natasha, pela paciência, ensinamentos, orientações. Aprendi muito com vocês. Valéria, recebi todas as suas sugestões e críticas como conselhos de uma segunda mãe. Natasha, sua calma e perseverança me fizeram acreditar que eu podia concluir esse mestrado.

A todos aqueles que contribuíram de alguma forma para a realização deste trabalho de mestrado.

RESUMO

Atualmente, dois temas de grande importância para a computação, estão sendo utilizados conjuntamente de uma forma cada vez mais aparente: a Descoberta de Conhecimento em Bancos de Dados (*Knowledge Discovery in Databases* – **KDD**) e as Ontologias. Com o aperfeiçoamento das formas com que os dados são armazenados, a quantidade de informação disponível para análise aumentou exponencialmente, tornando necessário técnicas para analisar esses dados e obter conhecimento para os mais diversos propósitos. Nesse contexto, o processo de KDD introduz etapas que possibilitam a descoberta de conhecimentos úteis, novos e com características que geralmente não podiam ser vistas apenas visualizando os dados de forma bruta.

Em um campo complementar, a Descoberta de Conhecimento em Banco de Dados pode ser beneficiada com **Ontologias**. Essas, de certa forma, apresentam a capacidade para armazenar o “conhecimento”, segundo um modelo de alta expressividade semântica, sobre determinados domínios. As ontologias permitem que o conhecimento seja recuperado através de inferências nas classes, descrições, propriedades e restrições.

Nas fases existentes no processo de descoberta do conhecimento, a **Seleção de Atributos** permite que o espaço de análise para os algoritmos de Mineração de Dados possa ser melhorado com atributos mais relevantes para o problema analisado. Porém, algumas vezes esses métodos de seleção não eliminam de forma satisfatória os atributos irrelevantes, pois não permitem uma análise prévia sobre o domínio tratado. Para tratar esse problema, esse trabalho propõe um sistema que utiliza ontologias para armazenar o conhecimento prévio sobre um domínio específico, possibilitando uma análise semântica antes não viável pelas metodologias convencionais. Foi elaborada uma ontologia, com reuso de diversos repositórios de ontologias disponíveis na Web, específica para o domínio médico e com possíveis especificações comuns nas principais áreas da medicina. Para introduzir semântica no processo de seleção de atributos primeiro é realizado o mapeamento entre os atributos do banco de dados e as classes da ontologia. Feito esse mapeamento, o usuário agora pode selecionar atributos através de categorias semânticas, reduzir a dimensionalidade dos dados e ainda visualizar redundâncias existentes entre atributos correlacionados semanticamente.

Palavras-Chave: Descoberta de Conhecimento em Banco de Dados, Ontologias, Seleção de Atributos.

ABSTRACT

Currently, two issues of great importance for the computation are being used together in an increasingly apparent: a Knowledge Discovery in Databases (**KDD**) and Ontologies. By developing the ways in which data is stored, the amount of information available for analysis has increased exponentially, making it necessary techniques to analyze data and gain knowledge for different purposes. In this sense, the KDD process introduces stages that enable the discovery of useful knowledge, and new features that usually cannot be seen only by viewing the data in raw form.

In a complementary field, the Knowledge Discovery can be benefited with **Ontologies**. These, in a sense, have the capacity to store the "knowledge" about certain areas. The knowledge that can be retrieved through inference classes, descriptions, properties and constraints.

Phases existing in the process of knowledge discovery, the selection of attributes allows the area of analysis for data mining algorithms can be improved with attributes more relevant to the problem analyzed. But sometimes these screening methods do not eliminate the attributes satisfactorily, do allow a preliminary analysis on the area treated. To address this problem this paper proposes a system that uses ontologies to store the prior knowledge about a specific domain, enabling a semantic analysis previously not possible using conventional methodologies. Was elaborated an ontology, with reuse of various repositories of ontologies available on the Web, specific to the medical field with a possible common specifications in key areas of medicine. To introduce semantics in the selection of attributes is first performed the mapping between data base attributes and classes of the ontology. Done this mapping, the user can now select attributes by semantic categories, reducing the dimensionality of the data and view redundancies between semantically related attributes.

Keywords: Knowledge Discovery in Databases, Ontologies, Attributes Selection.

LISTA DE ILUSTRAÇÕES

Figura 1 - Hierarquia entre Dado, Informação e Conhecimento.	22
Figura 2 - Etapas do Processo de KDD.	23
Figura 3 - Etapas Operacionais do Processo de KDD.	24
Figura 4 - Seleção de Atributos no Weka.	28
Figura 5 - Demonstrativo da linguagem OWL.	36
Figura 6 - Estrutura das Linguagens de Representação de Ontologias	37
Figura 7 - Classificação das Ontologias.	37
Figura 8 - Framework Genérico de Mineração de Dados com Ontologias.	41
Figura 9 - Ontologias nas Etapas de KDD.	42
Figura 10 – Distancia de Levenshtein na prática.	45
Figura 11 - Uso de Filtros para Seleção de Atributos.	46
Figura 12 - Uso de Wrapper para Seleção de Atributos.	47
Figura 13 - Esquema utilizado como exemplo no D2OMapper.	50
Figura 14 - Mapeamento entre Esquema do Banco de dados e Ontologia pelo D2OMapper.	51
Figura 15 - Comunicação de Ontologia e Base de Dados.	52
Figura 16 - Mapeamento final do KAON.	52
Figura 17 - Diagrama de Caso de Uso do SAAO.	56
Figura 18 - Arquitetura Básica do Sistema.	57
Figura 19 - Interface Gráfica SAAO.	60
Figura 20 - Método de construção da ontologia.	61
Figura 21 – Construção/Adaptação da Ontologia de Categorização.	63
Figura 22 - Nomenclatura sistematizada dos termos médicos.	65
Figura 23 - Hierarquia Principal.	66
Figura 24 - Classes Equivalentes.	67
Figura 25 - Sub-hierarquia da Ontologia SAAO.	68
Figura 26 - Indivíduos pertencentes a classes.	68
Figura 27 - Indivíduos Pertencentes a classe RDFS.	69
Figura 28 - Último nível de inferência.	70
Figura 29 – Parte da ontologia remodelada para efetuar o mapeamento dos dados.	73
Figura 30 – Parte da ontologia com Atributos Categorizados.	74
Figura 31 - Mapeamento Atributo – Classe.	75
Figura 32 - Extenso Esquema de um Banco de Dados.	76
Figura 33 - Similaridade Sintática.	77
Figura 34 - Atributos categorizados.	78
Figura 35 - Hierarquia usando árvores com atributos redundantes.	79
Figura 36 - Atributos candidatos a redundância.	80
Figura 37 - Visualização de Possíveis Redundâncias.	81
Figura 38 – Exemplo Redução de dimensionalidade.	83
Figura 39 - Reduzindo Mapeamento.	83
Figura 40 - Módulo para auxílio na redução de dimensionalidade.	84
Figura 41 - Salvando o Mapeamento.	86

Figura 42 - Atributos selecionados.....	86
Figura 43 - Captura de atributos na ferramenta WEKA.....	87
Figura 44 - Demonstrativo do Modelo de Pesquisa com Históricos dos Pacientes	91
Figura 45 - Tabela de Atributos.....	92
Figura 46 - Similaridade Sintática.	93
Figura 47 - Mapeando Atributos.	94
Figura 48 - Atributos mapeados (hierarquia expandida).	95
Figura 49 - Saída da seleção usando Cfs.	97
Figura 50 - Seleção de Categorias.	98
Figura 51 - Atributos resultantes da Consulta.	99
Figura 52 - SQL Viewer no Weka.....	100
Figura 53 - Seleção de atributos pós SAAO.....	101
Figura 54 - Regras de associação.	102
Figura 55 - Árvore de Classificação.	103

LISTA DE QUADROS

Quadro 1 - Ferramentas para criação de Ontologias.	39
---	----

LISTA DE TABELAS

Tabela 1- Trabalhos com ontologias e KDD	49
Tabela 2 - Características dos algoritmos para seleção de atributos.	59
Tabela 3 - Quantidade de atributos por algoritmo.	98
Tabela 4 - Classificação do especialista de domínio (médico).	100

LISTA DE ABREVIATURAS E SIGLAS

KDD – *Knowledge Discovery in Databases.*

NASA – *National Aeronautics and Space Administration.*

RNA – *Redes Neurais Artificiais.*

GSP – *Generalized Sequential Pattern algorithm.*

DHP – *Direct Hashing and Pruning.*

WEKA – *Waikato Environment for Knowledge Analysis.*

XML – *eXtensible Markup Language.*

RDF – *Resource Description Framework.*

SHOE – *Simple HTML Ontology Extensions.*

OIL – *Ontology Inference Language.*

OWL – *Web Ontology Language.*

API – *Application Programming Interface.*

HP – *Hewlett Packard.*

CYC – *enCYClopedia.*

CFS – *Correlation-based Feature Selection.*

KNN – *k-nearest neighbor*

SUMÁRIO

1 INTRODUÇÃO	15
1.1 Motivação	15
1.2 Objetivos	17
1.3 Organização da Dissertação	17
2 FUNDAMENTAÇÃO TEÓRICA.....	20
2.1 Introdução	20
2.2 Descoberta de Conhecimento em Bases de Dados	21
2.2.1 Introdução.....	21
2.2.2 Caracterização do Processo de KDD	23
2.2.3 Tarefa de Seleção de Atributos em KDD.....	26
2.2.3.1 Redução de Dados Horizontal	29
2.2.3.2 Redução de Dados Vertical	30
2.2.4 Técnicas de KDD	31
2.2.5 Ferramentas de KDD.....	32
2.3 Ontologias	33
2.3.1 Introdução.....	33
2.3.2 Linguagens de Representação de Ontologias.....	34
2.3.2.1 OWL (<i>Web Ontology Language</i>)	34
2.3.3 Classificação das Ontologias.....	37
2.3.4 Ferramentas para Criação de Ontologias	39
2.4 KDD e Ontologias	40
2.5 Estratégias de Apoio a Seleção de Atributos	41
2.5.2 Critérios Clássicos para Avaliação de Atributos.....	43
2.5.2.1 Distância de Levenshtein (<i>Edit Distance</i>)	44
2.6 Considerações Finais	47
3 TRABALHOS RELACIONADOS	48
3.1 Introdução	48
3.2 D2OMapper	50
3.3 KAON	51
3.4 Considerações Finais	53
4 UMA ABORDAGEM SEMÂNTICA PARA SELEÇÃO DE ATRIBUTOS NO PROCESSO DE KDD	54
4.1 Introdução	54
4.2 Especificação do SAAO	55
4.2.1 Considerações Iniciais.....	55
4.2.2 Arquitetura	56
4.3 Protótipo para Seleção de Atributos com Auxílio de Ontologias (SAAO)	58
4.3.1 Contextualização do Problema.....	58
4.3.2 Ferramenta Desenvolvida.....	59
4.4 Efetivação Semântica.....	60
4.4.1 Adaptação de Ontologia de Domínio	60
4.4.1.1 Determinar o Domínio de Escopo da Ontologia	61
4.4.1.2 Considerar o Reuso de outras Ontologias	62

4.4.1.3 Enumerar os Termos Importantes da Ontologia.....	63
4.4.1.4 Definir Classes e a Hierarquia de Classes	64
4.4.1.5 Outros Passos Incorporados na Construção da Ontologia.....	66
4.4.2 Incorporando Ontologias na Seleção de Atributos.....	70
4.4.3 Categorizando Atributos	74
4.4.4 Visualizando Redundâncias	78
4.4.5 Reduzindo Dimensionalidade dos Dados.....	81
4.4.6 Método de Salvamento e Utilização dos Resultados do SAAO.....	85
4.5 Considerações Finais	87
5 CENÁRIOS DE APLICAÇÃO DO PROTÓTIPO SAAO	89
5.1 Considerações Iniciais	89
5.2 Coleta de Dados	90
5.2 Elaborando o Mapeamento entre a Ontologia e os Atributos Capturados.....	93
5.3 Cenários de Aplicação	95
5.3.1 Primeira Análise: Seleção Comparativa.....	96
5.3.2 Segunda Análise Comparativa: Usando o SAAO em Conjunto aos Algoritmos para Seleção de Atributos.....	101
6 CONCLUSÃO E TRABALHOS FUTUROS.....	104
6.1 Conclusão.....	104
6.2 Contribuições	104
6.3 Trabalhos Futuros	105
REFERÊNCIAS	106
APÊNDICE – Ontologia criada para utilização no protótipo SAAO.....	111

1 INTRODUÇÃO

1.1 Motivação

O grande desenvolvimento de tecnologias para aquisição e armazenamento de dados, em praticamente todas as áreas de conhecimento, possibilitou o acúmulo desses dados em uma velocidade bastante elevada. A utilização dos Bancos de Dados foi um dos fatores que tornou possível essa facilidade para armazenar os dados e, atualmente, qualquer empresa ou segmento que necessite armazenar, recuperar e visualizar os dados de forma rápida, segura e eficiente os utiliza. Com essa grande quantidade de informação e levando em conta toda a dificuldade para analisar e extrair conhecimento desses Bancos de Dados, diversos métodos têm sido propostos para realizar a tarefa de descobrir conhecimento e fazer análises que antes eram impossíveis de serem descobertas apenas de forma visual. A descoberta de Conhecimento em Banco de Dados (Knowledge Discovery in Databases - KDD) (FAYYAD et al., 1996a) surge com esse objetivo, permitindo que sejam feitas descobertas de associações e padrões entre os dados como, agrupamentos, classificações e outras avaliações e análises possibilitadas com o uso de poderosos algoritmos executados no decorrer do processo de KDD.

O processo de KDD pode ser dividido, em um nível mais alto, em três fases: Pré-processamento, Mineração de Dados e Pós-processamento. A etapa de pré-processamento compreende as funções relacionadas à captação, organização e tratamento dos dados. A próxima etapa é chamada de Mineração de dados (*Data Mining*), que é onde realmente efetua-se uma busca por conhecimentos úteis ao contexto da aplicação. Nessa etapa são definidos os algoritmos e as técnicas a serem utilizadas no problema (HAN, KAMBER, 2005). A última etapa no processo é o Pós-processamento, que abrange todo o conhecimento obtido na Mineração de Dados e a interpretação deste conhecimento.

Dentro do Processo de KDD, na etapa de Pré-processamento, é realizada uma primeira tarefa de suma importância para todas as outras etapas posteriores: a Seleção de Atributos (HAN, KAMBER, 2005). A tarefa de Seleção de Atributos permite, por exemplo, a ordenação dos atributos segundo algum critério de importância, a redução da dimensionalidade do espaço de busca de atributos, remoção de dados contendo ruídos (valores divergentes, fora do padrão, também conhecidos como *outliers*) entre outros. Intuitivamente, quanto maior o número de atributos em uma base de dados, maior o poder discriminatório do classificador e a facilidade de extrair modelos de conhecimento da base. Porém, levando esse

conceito em termos práticos, isso nem sempre é verdade. Em primeiro lugar, existe o problema da dimensionalidade, fazendo com que o tempo computacional aumente agressivamente de acordo com o número de atributos presentes, pois muitos dos métodos são gulosos (fazendo apenas escolhas ótimas locais) e não trabalham com otimização do conjunto de atributos selecionados. Além disso, atributos irrelevantes podem confundir o algoritmo de aprendizagem e esconder as distribuições de pequenos conjuntos de atributos realmente relevantes. Existem vários métodos estatísticos que são utilizados para selecionar atributos, seja através de “ranqueamento” ou até mesmo filtração por critérios de distância sintática. Mas, nem sempre os atributos escolhidos são os melhores, apresentando problemas de redundância e grande necessidade do especialista de analisar iterativamente por várias etapas o conjunto atributos até encontrar o melhor subconjunto para o problema em questão. Essas análises consomem muito tempo no processo de seleção de atributos e consecutivamente da descoberta de conhecimento. Tendo em vista esse problema, esse trabalho propõe o uso de ontologias que auxiliarão o analista na tarefa de seleção de atributos em determinado domínio.

As ontologias trazem diversos benefícios para as mais variadas áreas de conhecimento, e da mesma forma, encontram seu uso bem evidenciado dentro das fases da Descoberta de Conhecimento em Bancos de Dados. Por possuírem grande capacidade para armazenar conhecimento, ou seja, apresentam informações através de classes com suas devidas descrições, propriedades, restrições, podem ser utilizadas para auxiliar o especialista a inferir e tomar decisões com uma menor dificuldade. Este trabalho propõe um sistema que utiliza ontologias para armazenar o conhecimento prévio sobre um domínio específico, possibilitando uma análise semântica antes não viável pelas metodologias convencionais. Para esse objetivo, foi elaborada uma ontologia com reuso de diversos repositórios de ontologias disponíveis na Web, específica para o domínio médico e com possíveis especificações comuns nas principais áreas da medicina. Para introduzir semântica no processo de seleção de atributos, primeiro é realizado o mapeamento entre os atributos e as classes da ontologia. Feito esse mapeamento, o usuário pode selecionar atributos através de categorias semânticas, reduzir a dimensionalidade dos dados e ainda, visualizar redundâncias existentes entre atributos correlacionados semanticamente.

1.2 Objetivos

Tendo em vista os problemas anteriormente descritos, os objetivos desse trabalho são:

- Pesquisar sobre o uso de ontologias no processo de descoberta de conhecimento em banco de dados;
- Utilizar categorias semânticas com o uso de mapeamentos entre atributos e classes da ontologia;
- Possibilitar ao especialista em KDD a melhor visualização de redundâncias presente nos dados, principalmente após a junção de bases de dados;
- Facilitar a visualização de generalizações dos atributos com o uso dos recursos de redução de dimensionalidade;
- Auxiliar a tarefa de seleção de atributos na etapa de Pré-processamento em KDD, permitindo assim uma melhoria nos atributos selecionados e um considerável ganho computacional para os algoritmos de mineração que serão utilizados em problemas específicos.

Esses objetivos serão alcançados com o mapeamento entre os atributos presentes na base de dados e as classes de uma ontologia adaptada para esse propósito. A categorização dos atributos é o primeiro resultado positivo do mapeamento e permite selecionar atributos relacionados semanticamente visualizando as categorias em que estão enquadrados. Outra otimização é a melhor visualização de redundâncias e possíveis reduções de dimensionalidade nos dados. Com a conclusão desses objetivos, será possível melhorar o conjunto de atributos que serão analisados pelos algoritmos de Mineração de Dados, e assim conseguir informações mais eficientes na descoberta de conhecimento.

1.3 Organização da Dissertação

Este trabalho está organizado do seguinte modo:

Capítulo 2: Fundamentação Teórica

Este capítulo tratará de alguns conceitos relativos à Descoberta de Conhecimento (KDD) e as Ontologias. O processo de KDD é abordado evidenciando suas diversas fases características principais. A etapa de Seleção de Atributos é evidenciada, mostrando seus

principais conceitos e técnicas para se reduzir o conjunto de atributos inicial do problema. As ferramentas mais utilizadas para efetuar a Descoberta de Conhecimento em Banco de Dados serão discutidas e avaliadas sob suas características mais evidentes. E, finalmente, as ontologias também são discutidas e tratadas em seu aspecto formal, mostrando classificações e algumas linguagens usadas para sua construção.

Capítulo 3: Trabalhos Relacionados

Esse capítulo abordará alguns trabalhos relacionados com o uso de KDD em suas diversas etapas com as ontologias. A análise introdutória especifica cada etapa e alguns trabalhos relacionados, por fim são evidenciados especificamente outros trabalhos que relacionam bem a fase de seleção de atributos em KDD e ontologias.

Capítulo 4: Uma abordagem Semântica para Seleção de Atributos em KDD.

Neste capítulo será desenvolvido um protótipo para auxiliar o especialista em KDD a melhorar a seleção de atributos na fase de pré-processamento.

Também irá evidenciar de forma mais efetiva os problemas existentes na seleção de atributos e as principais características importantes nessa etapa da descoberta de conhecimento em banco de dados. Outro ponto abordado será uma característica elaborada para avaliar e efetuar comparações sintáticas entre os atributos e as classes da ontologia. O ponto principal desse capítulo é a explicação das possibilidades de incorporar ontologias na seleção de atributos em KDD.

O foco principal está em usar ontologias para prover semântica que possa ser utilizada durante o processo de seleção para verificar atributos que poderão ser eliminados sem perda de informação útil. Esse processo é feito identificando atributos relacionados semanticamente após a realização de mapeamentos com as classes da ontologia. Será também abordada a redução de dimensionalidade e a visualização de redundâncias nos atributos mapeados. Todos os tópicos serão explicados e visualizados em ação na ferramenta protótipo desenvolvida para este trabalho.

Capítulo 5: Cenários de Aplicação do Protótipo SAAO.

Neste capítulo serão criados alguns cenários para demonstrar a utilização da ferramenta protótipo desenvolvida. Será evidenciado o processo de seleção de atributos no domínio médico.

Capítulo 6: Conclusão e Trabalhos Futuros.

Esse capítulo aborda o trabalho de forma geral e mostra os pontos positivos da abordagem utilizada, como também as dificuldades encontradas. Também são evidenciados alguns trabalhos futuros para melhoria ou até mesmo modificações no trabalho aqui especificado.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Introdução

Os computadores surgiram para diminuir nossas tarefas rotineiras: fazer contas, imprimir relatórios, armazenar dados. Essas tarefas passaram a ser cada vez mais complexas e exigir maior tecnologia e uma ampla sofisticação nas arquiteturas e programações. Dados científicos em pesquisas, incluindo missões espaciais da *National Aeronautics and Space Administration* (NASA) e o Projeto do Genoma Humano agrupam grande quantidade de dados. Grandes empresas como Americanas, Wal-Mart e Microsoft, armazenam quantidade gigantescas de informações diariamente. Pensando nesse cenário, naturalmente surgem algumas questões como: “O que fazer com todos os dados armazenados?”, “Como analisar e utilizar de maneira útil todo o volume de dados disponível?”, entre outras. Para suprir essa lacuna e auxiliar o homem na tarefa de analisar esse grande volume de dados surge a Descoberta de Conhecimento em Banco de Dados (*Knowledge Discovery in Databases – KDD*).

As ontologias são outro grande tema de fundamental importância na computação, pois servem como um repositório de conhecimento. Um recurso formal que permite a inserção de conceitos que são interligados e que de certa forma podem ser inferidos por ferramentas adequadas. O uso das ontologias não se restringe apenas à Computação e sim a várias áreas do conhecimento, fornecendo suporte para armazenar informações em forma de classes, propriedades, restrições, entre outros. Quando todos esses dados estão juntos nas ontologias, eles possibilitam a computadores entender, analisar e tomar decisões que antes só eram possíveis com a intervenção humana. Portanto, as ontologias codificam conhecimento passível de entendimento pelos computadores.

Levando em conta essas duas áreas que recebem grande atenção na atualidade, esse capítulo busca explicar de forma breve os conceitos, algumas características e ferramentas utilizadas em KDD e Ontologias. Na seção referente ao processo de KDD será feita uma caracterização do processo, com suas diversas fases. Será enfatizada a etapa do processo de KDD conhecida como Seleção de Atributos, onde os dados (atributos) que serão utilizados nas próximas etapas do processo serão escolhidos da melhor forma possível, a fim de otimizar e tornar o processo mais eficiente. Na última seção é feito um breve resumo sobre as técnicas

utilizadas e as ferramentas que são usadas para analisar as informações em Bancos de Dados. Para as Ontologias são mostradas algumas definições mais conhecidas; as linguagens utilizadas na sua construção; classificações existentes e ferramentas utilizadas no processo de criação e edição.

2.2 Descoberta de Conhecimento em Bases de Dados

2.2.1 Introdução

O objetivo do processo de KDD pode ser definido, resumidamente, como uma descoberta de conhecimento não trivial e potencialmente útil. O termo KDD teve seu uso formalizado em 1989 se referindo ao conceito de procurar conhecimento a partir de bases de dados. A definição mais conhecida foi proposta por um grupo de pesquisadores da seguinte forma: “KDD é um processo, de várias etapas, não trivial, interativo e iterativo, para definição de padrões compreensíveis, válidos, novos e potencialmente úteis a partir de grandes conjuntos de dados” (WEISS e INDURKHYA, 1998). Nessa definição, o termo *interativo*, indica a necessidade de atuação do homem como responsável pelo controle do processo, utilizando os recursos computacionais disponíveis em função da análise e da interpretação dos fatos observados e resultados obtidos ao longo do processo. Outro termo que aparece na definição é *iterativo*, e nesse contexto, sugere a possibilidade de repetições totais ou parciais do processo de KDD na busca de resultados satisfatórios por meio de refinamentos sucessivos.

A expressão *não trivial* alerta para a complexidade normalmente presente na execução de processos de KDD. A expressão *padrão válido* indica que o conhecimento deve ser verdadeiro e adequado ao contexto da aplicação. Um *padrão novo* deve acrescentar novos conhecimentos aos conhecimentos existentes no contexto da aplicação. Finalmente, um *conhecimento útil* é aquele que pode ser aplicado de forma a proporcionar benefícios ao usuário do processo de KDD (FAYYAD et al., 1996a). Também existem outras definições, e que de certo modo tratam o processo de KDD como Mineração de Dados, mas neste trabalho é importante ressaltar que Mineração de Dados será considerada uma etapa do processo de Descoberta de Conhecimento – KDD, algumas delas são:

- “Mineração de Dados é a análise dos dados observáveis com o objetivo de encontrar relacionamentos não usuais, resumindo-os de forma a serem compreensíveis e úteis ao analista” (HAND et al., 2005).
- “Descoberta de Conhecimento em Banco de Dados é um campo multidisciplinar que traz técnicas de áreas como: aprendizado de máquina, reconhecimento de padrões, estatística, banco de dados e visualiza uma forma de extrair conhecimento e informações de grandes bases de dados” (SIMOUDIS, 1998).

Para um melhor entendimento da descoberta de conhecimento em Banco de Dados é importante destacar as diferenças e a hierarquia entre dado, informação e conhecimento. Essa hierarquia enfatiza o processo desde a existência apenas de dados até a obtenção de conhecimento útil, conforme mostra a Figura 1 (BELLINGER et al., 2005).

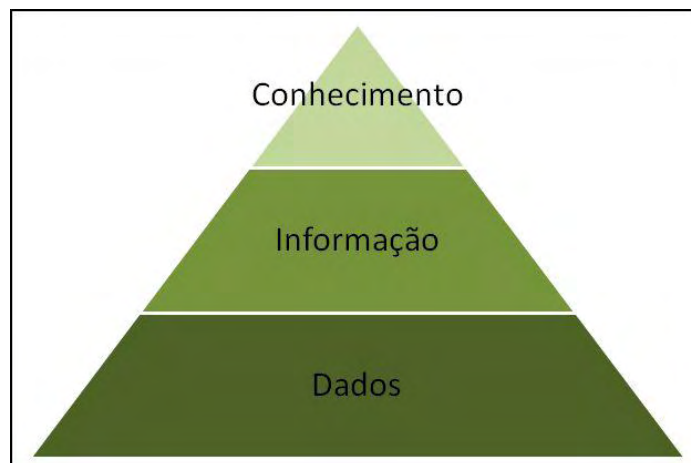


Figura 1 - Hierarquia entre Dado, Informação e Conhecimento.
Fonte: Adaptado de Bellinger et al. (2005).

Os dados podem ser interpretados como os itens elementares que serão captados e armazenados por recursos computacionais tradicionais de hardware e software. Por exemplo, pode-se pensar em uma base de dados de uma empresa financeira. As informações irão representar os dados processados, que apresentarão significados e contextos bem definidos. Nesse exemplo da financeira, poderia ser necessário saber a capacidade mensal de endividamento, podendo ser calculada a partir dos dados de renda e das despesas mensais de cada cliente, onde esse tipo de informação pode ser processada facilmente.

Por último, no topo da pirâmide, está o conhecimento. Padrão ou conjunto de padrões cuja formulação pode envolver e relacionar dados e informações. Pode ser

representado na forma de regras de produção ou outras forma de representação que são evidenciadas no estudo da descoberta de conhecimento e especificamente na fase de mineração dos dados (NALIATO, 2001).

2.2.2 Caracterização do Processo de KDD

O processo de descoberta de conhecimento é composto por várias etapas operacionais, que de acordo com Fayyad (1996a), são divididas em cinco fases: Seleção (*Selection*), Pré-Processamento (*Pré-Processing*), Transformação (*Transformation*), mineração de dados (*Data Mining*) e interpretação (*Interpretation*). Essa é uma sequência adotada pela maioria dos livros da área e que apresenta dados como entrada do processo e conhecimento como sua saída. Na figura 2 está representada graficamente essa sequência de fases do processo de KDD.

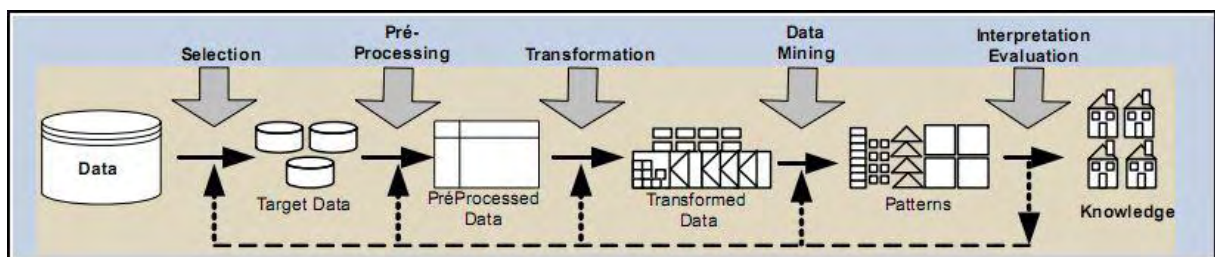


Figura 2 - Etapas do Processo de KDD.

Fonte: Adaptado de (FAYYAD et al., 1996a).

Para um melhor entendimento do processo, essas etapas podem ser agrupadas em três grandes etapas: Pré-Processamento, Mineração de Dados e Pós-Processamento. A etapa de pré-processamento compreende as funções relacionadas à captação, organização e tratamento dos dados. A próxima etapa é chamada de Mineração de dados (*Data Mining*), que é onde realmente efetua-se uma busca por conhecimentos úteis ao contexto da aplicação. Nessa etapa são definidos os algoritmos e as técnicas a serem utilizadas no problema (HAN, KAMBER, 2000). Por final, a última etapa no processo é o Pós-Processamento, que abrange todo o conhecimento obtido na Mineração de Dados. Essas três grandes etapas são representadas graficamente logo abaixo.

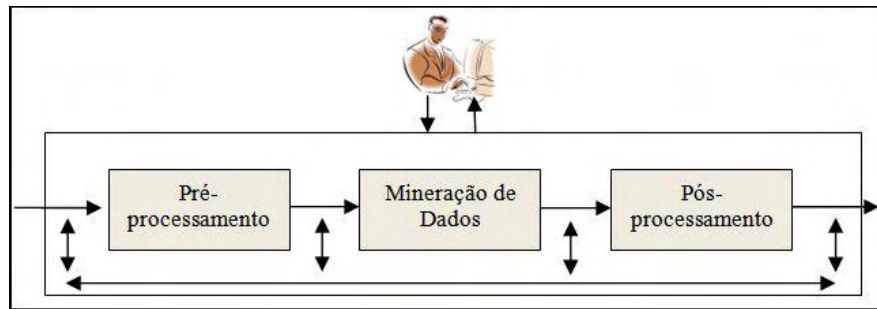


Figura 3 - Etapas Operacionais do Processo de KDD.
 Fonte: Adaptado de (FAYYAD et al., 1996a).

Na figura 3 existe a representação do analista que terá a tarefa de orientar a execução do processo. Essa exigência pode ser visualizada na definição formal de KDD vista anteriormente, quando é citado o termo *iterativo* e *iterativo* (na figura, as setas permitem visualizar a possibilidade de repetições), buscando assim resultados mais satisfatórios.

Cada etapa contém funções e tarefas que são realizadas durante o seu processo de execução. Inicialmente, na etapa de Pré-Processamento, podem-se realizar as seguintes funções:

- *Seleção de Dados*: compreende a identificação de quais informações, dentre as bases de dados existentes deve ser considerada durante o processo de KDD, essa etapa será mais bem explicada na seção 2.2.3, pois é um tópico de grande importância nesse trabalho.
- *Limpeza dos Dados*: qualquer tratamento realizado nos dados selecionados se enquadra nessa fase, de modo que seja assegurada a qualidade no processo (completude, veracidade e integridade). Informações errôneas, ausentes ou até mesmo inconsistentes devem ser corrigidas de forma a não comprometer as qualidades do modelo de conhecimento.
- *Codificação dos Dados*: nesse ponto os dados devem ser codificados de forma que possam ser usados como entrada para os algoritmos de mineração.
- *Enriquecimento dos Dados*: essa fase pode ser utilizada para que mais informação possa ser agregada a que já existe. Isso pode ser feito através de pesquisas de complementação, consultas a bases de dados externas, entre outras.

Na próxima etapa, chamada de mineração de dados, existem diversas técnicas: Redes Neurais (HAYKIN, 1999), Algoritmos Genéticos (DAVIS, 1990), Modelos Estatísticos e

Probabilísticos (MICHIE et al., 1994). A escolha da técnica depende, na maioria das vezes, do tipo de tarefa que vai ser realizada, ou seja, o tipo de conhecimento a ser minerado. Da mesma forma que a etapa anterior, que possuía funções que seriam realizadas durante sua execução, a Mineração de Dados possui tarefas. As tarefas que podem ser utilizadas são:

- *Descoberta de Associação*: Deverá abranger uma busca por itens que ocorram frequentemente de forma simultânea no Banco de Dados. Os algoritmos mais utilizados nesta etapa são Apriori, GSP (*Generalized Sequential Pattern algorithm*), DHP (*Direct Hashing and Pruning*), entre outros (Zaki, 2000).
- *Classificação*: Os itens serão mapeados em rótulos categóricos predefinidos, que serão chamados de classes. Depois que a classe é descoberta, a função prevê em que classes novos registros serão enquadrados. Um exemplo dessa função seria um sistema de apoio à decisão, que auxilia na filtragem e concessão de empréstimos, onde os clientes classificados na classe de bons pagadores serão escolhidos no processo. Os algoritmos mais utilizados nesse processo são as Redes Neurais, Algoritmos Genéticos, Lógica Indutiva (MITCHIE et al., 1994).
- *Regressão*: Nesta tarefa os registros do Banco de Dados serão mapeados para valores reais sendo muito parecida com a classificação, porém abrange apenas atributos contínuos. Como exemplo, poderia ser citado a estimativa da probabilidade de um paciente sobreviver, tomando o resultado de um conjunto de exames. Os métodos mais utilizados para essa tarefa podem ser estatística convencional, redes neurais, entre outros (SRIKANT et al., 1997).
- *Agrupamento (Clusterização)*: Será utilizada para separar os registros da Base de Dados em subconjuntos, que são chamados de grupos (*clusters*). No final, os elementos pertencentes ao mesmo cluster compartilharão propriedades comuns que os distingam dos demais. No desenvolvimento dessa tarefa podem ser utilizados algoritmos, tais como: K-Means, K-Modes, K-Prototipes, K-Medoids, Kohonen, dentre outros (MITCHIE et al., 1994).
- *Sumarização*: Irá procurar e identificar características comuns em conjuntos de dados (WEISS, INDURKHYA, 1998). Por exemplo, clientes que são assinantes de determinada revista. Pode-se usar Lógica Indutiva e algoritmos Genéticos para efetuar essa tarefa.

- *Detecção de Desvios*: Irá identificar registros no Banco de Dados que apresentam características que não se enquadram aos padrões considerados normais ao contexto (WEISS, INDURKHYA, 1998). Esses registros são chamados de *outliers*.
- *Descoberta de Sequências*: Seria como uma extensão da tarefa de descoberta de associações (HAN, 1996). Porém, são buscados itens frequentes levando em consideração várias transações ao longo de um período. Um exemplo disso seria clientes que comprem produtos em um supermercado e podem se beneficiar das regras de associação para identificar itens que ocorram de forma simultânea, porém com a descoberta de sequências é possível também identificar a ordem na qual os produtos são comprados ao longo do tempo.

A etapa de Pós-Processamento, de acordo com Fayyad et al. (1996a), tem o objetivo de facilitar a interpretação e a avaliação conhecimento obtido para benefício do homem. Nessa etapa são criadas simplificações com gráficos, diagramas e relatórios que mostrem de uma forma mais amigável o conhecimento obtido.

2.2.3 Tarefa de Seleção de Atributos em KDD

Os atributos são as variáveis do problema em questão e podem ser classificados sob dois aspectos distintos: quanto à representação de seus valores (tipo de dado), indicando a forma como eles são armazenados; e quanto à natureza da informação (tipo de variável), expressando a natureza com que a informação deve ser interpretada. Os tipos de dados indicam a forma como eles estão armazenados. Já os tipos de variáveis expressam a natureza com que a informação deve ser interpretada. Dessa forma, as variáveis podem ser classificadas em (PYLE, 1999):

- *Nominais ou Categóricas* – São variáveis utilizadas para nomear ou atribuir rótulos a objetos. Podem assumir valores pertencentes a um conjunto finito e pequeno de estados possíveis. Por exemplo: o estado civil de uma pessoa – solteiro, casado, viúvo, divorciado etc. Nesse tipo de variável não existe um tipo de ordenamento, pois não se pode dizer que “solteiro” é menor que “viúvo”.
- *Discretas* – São muito parecidas com as variáveis nominais, mas os valores que elas podem assumir possuem um ordenamento, e este possui algum significado. O dia da semana é um bom exemplo desse tipo de variável, pois

“segunda-feira” vem após o “domingo” e antes da “terça”, e assim sucessivamente.

- *Contínuas* – São variáveis quantitativas cujos valores possuem uma relação de ordem entre eles. Esses valores podem ser finitos ou infinitos. *Renda* e *Idade* são exemplos de variáveis contínuas, pois ambas podem ser ordenadas e normalmente podem ser medidas, por exemplo: peso (balança), altura (régua), tempo (relógio), etc.

A etapa de Seleção de Atributos compreende, em essência, a identificação de quais informações dentre as bases de dados existentes, devem ser efetivamente consideradas para os próximos passos do processo de KDD. Nos casos em que já exista uma estrutura de *Data Warehouse* (HALL et. al, 2009) , deve-se verificar a possibilidade de que esta seja utilizada no processo de KDD. Nos demais casos, é comum a congregação dos dados em uma única tarefa. Tal fato justifica-se, pois a maioria dos métodos de Mineração de Dados pressupõe que os dados estejam organizados em uma única estrutura tabular bidimensional. Essa junção dos dados em uma única tabela pode ocorrer de duas formas:

- *Junção Direta* – Todos os atributos e registros da base de dados são incluídos na nova tabela, sem uma análise crítica quanto a que variáveis e que casos podem realmente contribuir para o processo de KDD.
- *Junção Orientada* – O especialista no Domínio da aplicação, em parceria com o especialista de KDD, escolhe os atributos e os registros com algum potencial para influir no processo de KDD. É importante mencionar que se recomenda que sejam desconsiderados atributos e registros sobre os quais se tenha uma visão clara quanto à inexistência de potencial de contribuição para o processo de KDD.

A maioria dos algoritmos usados no processo de Mineração de Dados necessita de um conjunto de atributos não muito grande, mas com grande poder indutivo. A visualização da importância de alguns atributos e suas contribuições no processo de KDD às vezes pode ser bastante trabalhosa para o especialista, principalmente quando o número de atributos é muito grande. O auxílio para efetuar essa seleção se torna necessário e útil na grande maioria dos casos. A diminuição no espaço de hipótese (quantidade de atributos existentes) permite que os algoritmos trabalhem de forma mais rápida e eficiente, provendo um melhor critério de classificação com resultados compactos e mais fáceis de serem interpretados.

Dentre as ferramentas que serão citadas na Seção 2.2.4, por exemplo, o Weka (HALL et. al, 2009) apresenta vários algoritmos de seleção de atributos. Essa seleção é elaborada efetuando uma busca no espaço do subconjunto dos atributos e em seguida avaliando cada um. Onde, dependendo do tipo de atributo, pode apenas ranquear (posicionar do mais importante ao menos importante) ou evidenciar os atributos que obtiveram melhor análise estatística.

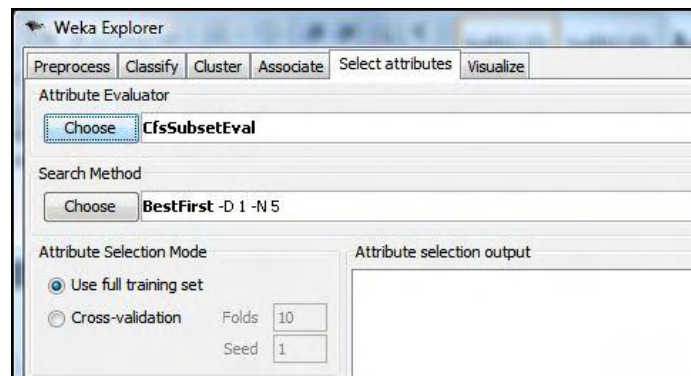


Figura 4 - Seleção de Atributos no Weka.
Fonte: Pesquisa Direta/2009.

Na figura 4, o Weka está na categoria de Seleção de Atributos (*Select Attributes*), sendo possível selecionar o método de avaliação de cada atributo e o modo como serão buscados. Os métodos para seleção de atributos avaliam de forma individual cada atributo ou um subconjunto de atributos, por exemplo:

- *CFS (Correlation-based Feature Selection)* – Avalia um conjunto de atributos, considerando “bons” se esses atributos forem altamente correlacionados com a classe e não correlacionados entre si. Calcula uma matriz de correlação atributo-classe e atributo-atributo.
- *Information Gain Attribute Ranking* – É um dos métodos mais simples e rápidos para seleção de atributos, avaliando os atributos com base em medidas dos seus ganhos de informação com relação a classe. É um dos tipos de métodos que fazem “*ranking*” de atributos um a um.
- *Principal Components* – É uma técnica estatística que reduz a dimensionalidade dos dados transformando o espaço de atributos original.

A Seleção de Atributos conta com dois tipos de redução da quantidade de dados existentes, a Redução de Dados Horizontal e Redução de Dados Vertical que serão explicados nas subseções abaixo.

2.2.3.1 Redução de Dados Horizontal

A Redução de Dados Horizontal é caracterizada pela escolha de casos e as operações mais utilizadas nesse tipo de redução são: Amostragem Aleatória, Eliminação Direta de Casos, Segmentação do Banco de Dados e Agregação de Informações.

A Segmentação do Banco de Dados é feita escolhendo um ou mais atributos para nortear o processo de segmentação. Um exemplo interessante seria uma empresa financeira que quer analisar os clientes que moram em residência própria, implementando uma instrução de seleção no seu Banco de Dados que mostre apenas esses clientes. Nesse caso, o conjunto de dados resultante desta consulta se tornaria o conjunto efetivamente considerado deste ponto em diante no processo de KDD.

A operação de Eliminação Direta de Casos pode ser interpretada como uma variação da Segmentação (operação de complemento), e nela são especificados os casos a serem eliminados e não os casos que devem permanecer em análise. A operação feita seria exatamente o contrário da anterior.

Diferentemente das operações anteriores, a Amostragem Aleatória consiste em sortear da base de dados um número preestabelecido de registros, de tal forma que o conjunto resultante possua uma quantidade menor que a original. Ela pode ser feita de forma simples, onde todas as tuplas possuem a mesma possibilidade de seleção. Também pode ser feita de forma estratificada, onde os dados são divididos em grupos disjuntos e selecionando aleatoriamente um subconjunto de amostras para cada grupo.

Por último, a Agregação de Informações consiste em reunir os dados de forma a reduzir o conjunto de dados original. Na agregação, dados com maior nível de detalhamento são consolidados em novas informações com menor detalhe. Por exemplo: somar os valores de todas as compras de cada cliente, obtendo o total de despesas por ele realizadas durante um determinado período.

2.2.3.2 Redução de Dados Vertical

A Redução de Dados Vertical é uma operação de pré-processamento muito importante no processo de KDD. O problema consiste em identificar qual das 2^n combinações entre tais atributos deve ser considerada no processo de descoberta de conhecimento. Em outras palavras, a redução de dados vertical, também denominada redução de dimensão, é implementada pela eliminação ou pela substituição dos atributos de um conjunto de dados. Tem como objetivo encontrar um conjunto mínimo de atributos de tal forma que a informação original seja preservada. As principais motivações para a aplicação da redução de dados vertical são:

- Um conjunto de atributos bem selecionados pode conduzir a modelos de conhecimento mais concisos e com maior precisão;
- Se o método de seleção de atributos for rápido, o tempo de processamento necessário para utilizá-lo e, em seguida, aplicar o algoritmo de Mineração de Dados será bem menor;
- A eliminação de um atributo é muito mais significativa em termos de redução do tamanho de um conjunto de dados do que a exclusão de um registro.

De acordo com Freitas (2002), as abordagens mais utilizadas na Redução de Dados Vertical são:

- Abordagem Independente de Modelo (*Filter*) – Nessa abordagem, a seleção de atributos é realizada sem considerar o algoritmo de Mineração de Dados que será aplicado nos atributos selecionados;
- Abordagem Dependente de Modelo (*Wrapper*) – Nessa abordagem o algoritmo de Mineração de Dados é experimentado para cada conjunto de atributos e os resultados são depois avaliados.

Existem três estratégias clássicas para escolha do conjunto de atributos que podem ser utilizados em quaisquer das abordagens acima, são elas (HAN; KAMBER, 1999) :

- Seleção Sequencial para Frente (*Forward Selection*) – Começa com o conjunto de atributos vazio, onde iterativamente cada atributo é adicionado ao subconjunto de candidatos de forma que é avaliado segundo alguma medida de qualidade.

- Seleção Sequencial para Trás (*Backward Selection*) – É feito de forma contrária ao anterior, o conjunto começa completo e então cada atributo é retirado segundo alguma medida de qualidade.
- Combinação das Estratégias Anteriores – A seleção para Frente e para Trás podem ser combinadas de tal forma que, a cada passo do algoritmo, o algoritmo seleciona o melhor atributo e remove o pior da mesma forma.

Para finalizar é ainda interessante ressaltar que algoritmos Genéticos, Algoritmos para Indução de Árvores de Decisão (ID3 e C4.5) podem ser utilizados no processo de otimização do conjunto de atributos.

2.2.4 Técnicas de KDD

Existem várias técnicas e algoritmos para Mineração de Dados, podendo ser divididos em: *Técnicas Tradicionais*, *Técnicas Específicas* e *Técnicas Híbridas*. As *Técnicas Tradicionais* são tecnologias já existentes e que já têm sido aplicadas a Mineração de Dados, algumas delas são:

- *Redes Neurais*: Simula uma Rede Neural Artificial (RNA), tendo capacidade de aprendizado a partir de repetidas apresentações dos dados, generalização do conhecimento através de exemplos anteriores, associação e relação entre padrões de natureza distinta e também identificam a essência de um conjunto de dados de entrada (HAYKIN, 2007).
- *Lógica Nebulosa*: Permite lidar com informações imprecisas ou subjetivas, imitando a habilidade humana de tomar decisões em ambientes de incertezas e imprecisões (PEDRYCZ; GOMIDE, 1998).
- *Algoritmos Genéticos*: São inspirados na evolução natural e na genética, sendo aplicados em complexos problemas de otimização. Irão existir variáveis para descrever os aspectos do problema, restrições indicando os valores que essas variáveis podem ter e funções que são utilizadas para avaliar a solução (PACHECO, 2000).

As *Técnicas Específicas* são desenvolvidas especificamente para aplicações em tarefas de KDD, por exemplo, o algoritmo *Apriori* foi desenvolvido para a tarefa de Descoberta de Associação. As *Técnicas Híbridas* irão gerar sistemas híbridos que utilizam mais de uma técnica para a solução de um problema de modelagem (SOUZA, 1999).

2.2.5 Ferramentas de KDD

O processo de KDD apresenta uma complexidade que pode ser motivada por diversos fatores, e podem ser divididos em dois conjuntos: fatores operacionais e fatores de controle (HELLERSTEIN et al., 1999) (FAYYAD et al., 1999b). Existem ferramentas para ambos os propósitos, ou seja, ferramentas que facilitam o controle do processo de KDD e ferramentas que tratam o processo operacional da descoberta de conhecimento.

Comentando um pouco sobre as ferramentas operacionais, atualmente encontram-se disponíveis academicamente diversas ferramentas que implementam ambientes integrados para facilitar a execução das etapas operacionais de KDD. Algumas delas são:

- WEKA (*Waikato Environment for Knowledge Analysis*) – é uma ferramenta de código aberto, aplicável em KDD, bastante flexível e desenvolvido na linguagem JAVA. Possui quatro diferentes implementações: *Simple Client* (o usuário interage por meio de linhas de comando), *Explorer* (é a mais comum, enquadrando separadamente as etapas de Pré-Processamento, Mineração de Dados e Pós-processamento), *Experimenter* (ambiente para realizar testes estatísticos) e o *KnowledgeFlow* (ferramenta gráfica para planejamento de ações). Disponível no endereço eletrônico da Universidade de Waikato, na Nova Zelândia (www.cs.waikato.ac.nz/ml/weka).
- Intelligent Miner: pertencente a IBM, mas não dependente de Bancos de Dados da mesma. É escalável e tem suporte para várias plataformas. Permite a utilização de algoritmos de mineração de forma individual ou combinada, para solucionar os problemas de KDD (www.ibm.com).
- Wiz Rule: Software de mineração de dados israelense, muito útil em aplicações envolvendo detecção de desvios. As relações entre dados identificadas pelo Wiz Rule podem ser representadas na forma de regras de produção ou como fórmulas matemáticas. Possui bom desempenho na identificação de possíveis erros de escrita nos valores das variáveis analisadas (www.wizsoft.com).

As ferramentas relacionadas ao controle do processo de KDD contribuem de forma significativa, auxiliando o homem na condução do processo. Existem poucas ferramentas desenvolvidas com esse propósito, muitas delas se encontram em fase de elaboração.

Algumas publicações que tratam desse tema são: (LIVINGSTON et al., 2000), (BERNSTEIN et al., 2002), (SOARES et al., 2001), entre outros.

2.3 Ontologias

2.3.1 Introdução

O termo *ontologia* tem origem na Filosofia, onde é definida como uma ciência que estuda a natureza da existência (SOUZA, ALVARENGA, 2004). Porém, vem sendo utilizada em várias áreas de conhecimento, como por exemplo, na Inteligência Artificial e Engenharia de Software, bem como em subáreas da computação como a Descoberta de Conhecimento em Banco de Dados. A definição mais citada sobre ontologias, segundo Gruber (1993, p.1), é que: “ontologia é uma especificação formal e explícita de uma conceitualização compartilhada”. Existem dois pontos centrais nessa definição, o conceito de *formal*, significando que uma ontologia deve ser passível de processamento automático; e outro ponto seria a palavra *compartilhada*, que mostrará a capacidade de uma ontologia capturar o conhecimento consensual, aceito por um grupo de pessoas (DAVIES, STUDER, WARREN, 2006).

Guarino (1998) reuniu e publicou as seguintes definições para o termo ontologia, propostas na literatura:

1. Ontologia é uma disciplina da Filosofia.
2. Ontologia é uma conceitualização formal de um sistema.
3. Ontologia é um relato semântico formal.
4. Ontologia é uma especificação de uma conceitualização.
5. Ontologia é uma representação de um sistema conceitual através de uma teoria da Lógica.
6. Ontologia é um vocabulário utilizado por uma teoria Lógica.
7. Ontologia representa o meta-nível da especificação de uma teoria de Lógica.

As ontologias facilitam a reutilização do conhecimento e sua compreensão pelas pessoas. Definem conceitos de diferentes domínios, seus relacionamentos e atributos, podendo então fazer inferências sobre os conceitos que elas armazenam e aumentar a comunicação entre humanos e computadores. Segundo Jasper et al. (1999), os três principais

usos das ontologias são: (1) prover comunicação entre humanos; (2) possibilitar a interoperabilidade e comunicação entre os softwares, e (3) possibilitar maior qualidade e design para os sistemas. Dessa forma as ontologias facilitarão a reutilização do conhecimento e sua compreensão pelas pessoas, mostrando sua capacidade de representar formalmente o conhecimento.

2.3.2 Linguagens de Representação de Ontologias

Os percussores das linguagens para representar ontologias tiveram início na década de 90 a partir dos paradigmas de representação do conhecimento. Linguagens como KIF, Ontolingua, LOOM, OKBC, foram os primeiros mecanismos de representação dessa época, todos baseados em lógica de primeira ordem. Desta forma, no contexto da Web Semântica, nos finais dos anos 90 apareceram as primeiras linguagens para representação de ontologias baseadas em XML (*eXtensible Markup Language*). Essa linguagem de marcação se tornou padrão e base para construção das outras linguagens que surgiram mais adiante. A primeira foi chamada SHOE (*Simple HTML Ontology Extensions*) (1996), seguida da XOL (1999) e finalmente aquela que foi adotada pelo W3C, chamada de RDF (*Resource Description Framework*). O RDF é utilizado para descrever propriedades, informações sobre páginas, classificar figuras, descrever conteúdo para sistemas de busca, entre outras utilizações. Sua estrutura básica compreende o estabelecimento de triplas compostas por sujeito (recurso), predicado (propriedade) e objeto (valor), onde é visível que se formaram frases para representar as informações necessárias para descrever um recurso (FENSEL et al. 2007).

Após o surgimento do RDF, foi criada uma extensão chamada de RDF *Schema*, permitindo o uso de classes e propriedades e assim possibilitando a construção de ontologias bastante simples. Na Europa criaram a OIL (*Ontology Inference Language*), que tinha uma maior capacidade descritiva, assim como a DAML (*DARPA Agent Markup Language*) que foi criada na mesma época. Houve uma fusão das duas linguagens e logo depois surgiu a linguagem padrão para construção de ontologias, chamada de OWL (*Web Ontology Language*) (McGUINNESS; HARMELEN, 2003).

2.3.2.1 OWL (*Web Ontology Language*)

A OWL (*Web Ontology Language*) (OWL; 2004) é uma linguagem de definição de ontologias, recomendada pela W3C, e é projetada para ser usada por aplicações que precisam

processar o conteúdo de informação, em vez de apenas apresentar esta informação. Seu desenvolvimento teve início em março de 2002, e consiste em uma revisão da linguagem DAM + OIL.

OWL facilita uma melhor interpretação de máquina do conteúdo da *web*, sendo suportada por XML, RDF e RDF Schema, provendo vocabulário adicional junto com uma semântica formal. É usada para explicitar as representações de significados de termos em vocabulários, e os relacionamentos entre estes termos (OWL; 2004).

De uma forma geral, para oferecer a interoperabilidade dos dados, duas condições são necessárias:

- Utilização de uma sintaxe comum: Habilitando as aplicações a analisarem sintaticamente os dados. O XML provê esta característica de uma sintaxe comum;
- Adoção de meios para compreensão da semântica: habilitando as aplicações a utilizarem os dados. A OWL provê uma forma padrão de expressar a semântica.

A sintaxe OWL é baseada em XML e RDF, compatível com RDFS. Ela é uma revisão da linguagem DAM + OIL.

A OWL é um conjunto de elementos e atributos XML, com significados padronizados, que são utilizados para definir termos e seus relacionamentos. Ela estende o RDF Schema, ou seja, torna-o mais amplo, acrescentando outros axiomas e construtores prontos, como, por exemplo, *equivalentClass* e *TransitiveProperty* (OWL; 2004).

A OWL possui três sub-linguagens:

- OWL Lite: Fornece um suporte básico aos usuários que necessitam classificar uma hierarquia de classes e definições simples de propriedades. Por exemplo, a OWL Lite possibilita restringir cardinalidade para propriedades somente com os valores 0 ou 1, enquanto que as OWL DL e Full não possuem essa restrição. Dessa forma, ela facilita o desenvolvimento de ferramentas que lhe dêem suporte, visto que não apresenta grande complexidade para a implementação de suas funcionalidades.
- OWL DL: Fornece suporte aos usuários que necessitam de uma maior expressividade para o processamento, incluindo todas as construções da linguagem OWL.

- OWL Full: Suporte mais complexo da linguagem, permitindo classificar e projetar classes com o mais alto grau de semântica.

A linguagem OWL permite definir classes, com seus devidos atributos, relações com propriedades de cardinalidade, simetria e transitividade. Um pequeno trecho, do cabeçalho de uma ontologia para CDs de música, escrita em linguagem OWL, pode ser vista a figura 5.

```
<rdf:RDF xmlns:owl = http://www.w3.org/2002/07/owl#
  xmlns:rdf = http://www.w3.org/1999/02/22-rdf-syntax-ns#
  xmlns:rdfs = http://www.w3.org/2000/01/rdf-schema#
  xmlns:xsd = http://www.w3.org/2001/XMLSchema#>
  <owl:Ontology rdf:about="">
    <rdfs:comment>Ontologia OWL para CD de Música</rdfs:comment>
    <owl:priorVersion
      rdf:resource="http://www.exemplo.org/oldVersion"/>
    <owl:imports
      rdf:resource="http://www.exemplo.org/musicCD"/>
    <rdfs:label>MusicCD Ontology</rdfs:label>
  </owl:Ontology>
</rdf:RDF>
```

Figura 5 - Demonstrativo da linguagem OWL.

O mais básico dos conceitos acerca de um domínio corresponde às classes, que são os nodos das árvores de taxonomia. Cada objeto na linguagem OWL é membro da classe *owl:Thing* por padrão, a qual já é predefinida na OWL. Dessa forma cada classe definida pelo usuário é implicitamente subclasse de *owl:thing* (OWL; 2004). Após definir todas as classes e subclasses, deve-se criar as propriedades que descrevem características e relacionamentos entre as classes.

Atualmente, a maioria das linguagens para construção de ontologias na web se baseiam em RDF e RDF *Schema*, que por sua vez têm como base a linguagem XML. Isso se torna muito vantajoso pois é mais fácil de interoperar com outras linguagens de programação. Para a linguagem Java, por exemplo, a API (*Application Programming Interface*) mais utilizada é a Jena¹ (CARROL et al., 2004) que foi elaborada pela *Hewlett Packard* (HP) e permite trabalhar tanto em RDF como em OWL.

Na figura 6, é mostrada a estrutura das distintas linguagens para uso de ontologias na web, cada linguagem se baseando em outra que se encontra uma camada inferior.

¹ Disponível no endereço: jena.sourceforge.net

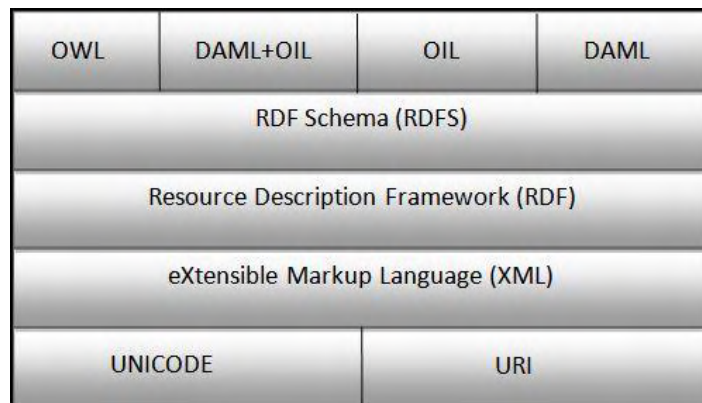


Figura 6 - Estrutura das Linguagens de Representação de Ontologias
Fonte: Adaptado de Breitman (2005).

A Figura 6 representa uma forma simplificada o “*layer cake*” de camadas da Web Semântica para facilitar o entendimento.

2.3.3 Classificação das Ontologias

As ontologias podem ser classificadas de diferentes formas e geralmente essa classificação é elaborada levando-se em conta a estrutura interna e o conteúdo das ontologias (LASSILA; MCGUINNESS, 2001).

A classificação leva em conta três fatores de classificação: Expressividade (*Expressiveness*), Especificidade (*Specificity*) e Propósito (*Purpose*) (Figura 7).

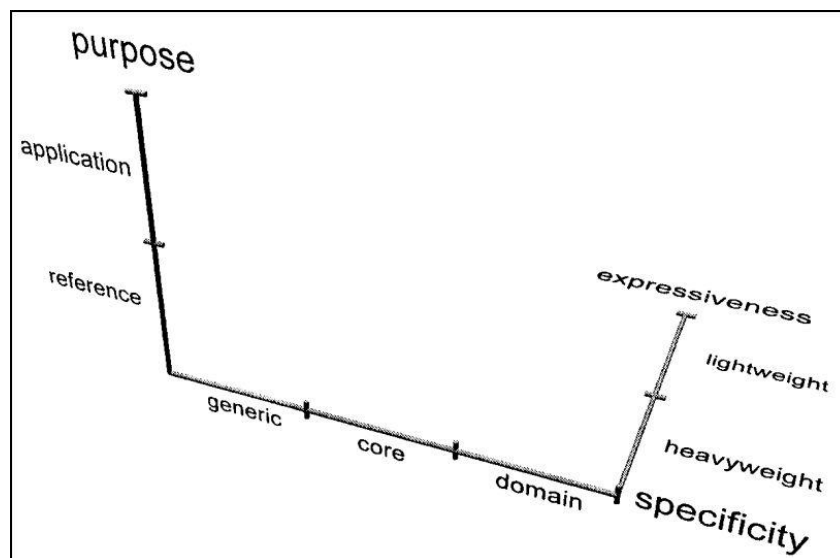


Figura 7 - Classificação das Ontologias.
Fonte: Oberle (2005).

Primeiramente, levando em conta a expressividade, as ontologias se classificam da seguinte forma:

- *Pesadas (heavyweight)*: tentam fazer uma classificação com o maior número de detalhes possíveis. Apresentam muitos conceitos para evitar ambiguidades e más interpretações.
- *Leves (Lightweight)*: diferentemente das “pesadas”, apresentam o mínimo de axiomas possíveis, diminuindo com isso seu nível de expressividade. Geralmente são reduzidas a simples taxonomias com poucas associações.

A classificação das ontologias quanto à especificidade pode ser dividida em quatro critérios e de acordo com Guarino (1998) são:

- *Ontologias de nível superior (Generic Ontology)*: apresentam conceitos muito genéricos, como espaço, tempo e eventos. São projetadas para serem independentes de domínio e podendo ser reutilizadas na criação de novas ontologias. Exemplos desses tipos de ontologias são a WordNet² e a Cyc³.
- *Ontologias de Domínio (Domain Ontology)*: descrevem um vocabulário relativo a um domínio específico, especializando os conceitos presentes na ontologia de alto nível.
- *Ontologias de Núcleo (Core Ontology)*: abrangerão conceitos genéricos a um grupo de domínios.

Por último, a classificação das ontologias levando em conta seu propósito (Borgo et al., 2002):

- *Ontologias de Aplicação (Application Ontology)*: são usadas durante a execução de uma aplicação específica, trazendo restrições e regras para execução das tarefas envolvidas.
- *Ontologias de Referência (Reference Ontology)*: Usadas no desenvolvimento de aplicações para entendimento mútuo entre agentes (humanos ou artificiais) pertencentes a diferentes comunidades, estabilizando um consenso entre as linguagens adotadas na comunicação.

² Encontrado em: <http://www.cogsci.princeton.edu/~wn/>.

³ O nome dessa ontologia foi derivado da palavra enciclopédia, que no inglês significa *enCYClopædia*.

2.3.4 Ferramentas para Criação de Ontologias

Existem várias ferramentas para criar e manusear ontologias, onde a maioria delas apresenta uma interface amigável e grande poder para manipular RDF e OWL de uma forma mais simples e eficiente do que criar passo a passo, manualmente. Logo abaixo pode ser visto um quadro comparativo com algumas ferramentas existentes para manipulação de ontologias e suas características principais.

C&L http://sl.les.inf.puc-rio.br/cel	Provê apoio semi-automático ao processo de construção de ontologias, sendo desenvolvido e disponibilizado utilizando software livre.
OilEd http://oiled.man.ac.uk	Editor simples que oferece suporte a ontologias desenvolvidas nas linguagens DAML+OIL e OWL. Não provê suporte para nenhuma metodologia de desenvolvimento de ontologias, mas permite a construção e verificação de inconsistências com o FACT. Apresenta um visualizador de hierarquia de classes chamado OilViz que fica incorporado na ferramenta principal.
Protégé HTTP://protege.stanford.edu	É um dos mais completos ambientes para criação e edição de ontologias. Livre, feito em Java e que tem arquitetura aberta, permitindo a adição de <i>plugins</i> . Também apresenta um mecanismo de inferência para verificação de inconsistências chamado: Pellet.
OntoEdit	Essa ferramenta implementa um processo específico para criação de ontologias, segmentando em três fases: especificação de requisitos, refinamento e avaliação. Na primeira fase são coletados os requisitos aos quais a ontologia dará suporte, em seguida a ontologia é refinada e finalmente na validação é verificado se corresponde às especificações do documento de requisitos.
Chimaera	Software que apóia a manutenção e a criação de ontologias. Tem funções para combinar múltiplas ontologias e diagnosticar ontologias individuais ou múltiplas.

Quadro 1 - Ferramentas para criação de Ontologias.
Fonte: Breitman (2005).

2.4 KDD e Ontologias

Mineração de dados, que nesse caso está se referindo ao processo de Descoberta de Conhecimento em Bases de Dados, enfrenta um grande problema até hoje: a forma como conhecimento prévio pode ser armazenado e utilizado em uma análise posterior ao processo. A existência de uma estrutura que facilite e ajude na manipulação desse tipo de conhecimento pode ajudar na seleção das informações e com isso diminuir bastante a quantidade de informações que precisam ser analisadas. As ontologias são um modelo conceitual satisfatório para armazenar e manipular o conhecimento, resultando assim em uma melhor automação da Descoberta de Conhecimento, seja ela em qualquer uma das fases que constitui o processo de KDD (ORALLO et al., 2004) (HAND; KAMBER, 2001).

De uma forma geral, o uso das Ontologias e a Mineração de Dados, são divididos pelos estudiosos da área de duas formas distintas (NIGRO; CÍSARO; XODO, 2008):

- Partindo das Ontologias para a Mineração de Dados – Nesse caso será incorporado conhecimento ao processo com a adição de Ontologias.
- Partindo da Mineração de Dados para as Ontologias – Dessa forma é adicionado conhecimento de domínio a informação de entrada ou até mesmo usar essas Ontologias para representar os resultados. As aplicações que mais utilizam esse tipo de abordagem se situam em áreas como Medicina, tais como na representação dos genes; Biologia, representando as taxonomias; Dados Espaciais, especificamente em Bancos de dados Geográficos nas Geociências.

Existem trabalhos que utilizam de forma satisfatória ontologias na Mineração de Dados e de certa forma eles podem ser representados genericamente pelo framework mostrado logo abaixo (NIGRO; CÍSARO; XODO, 2008).

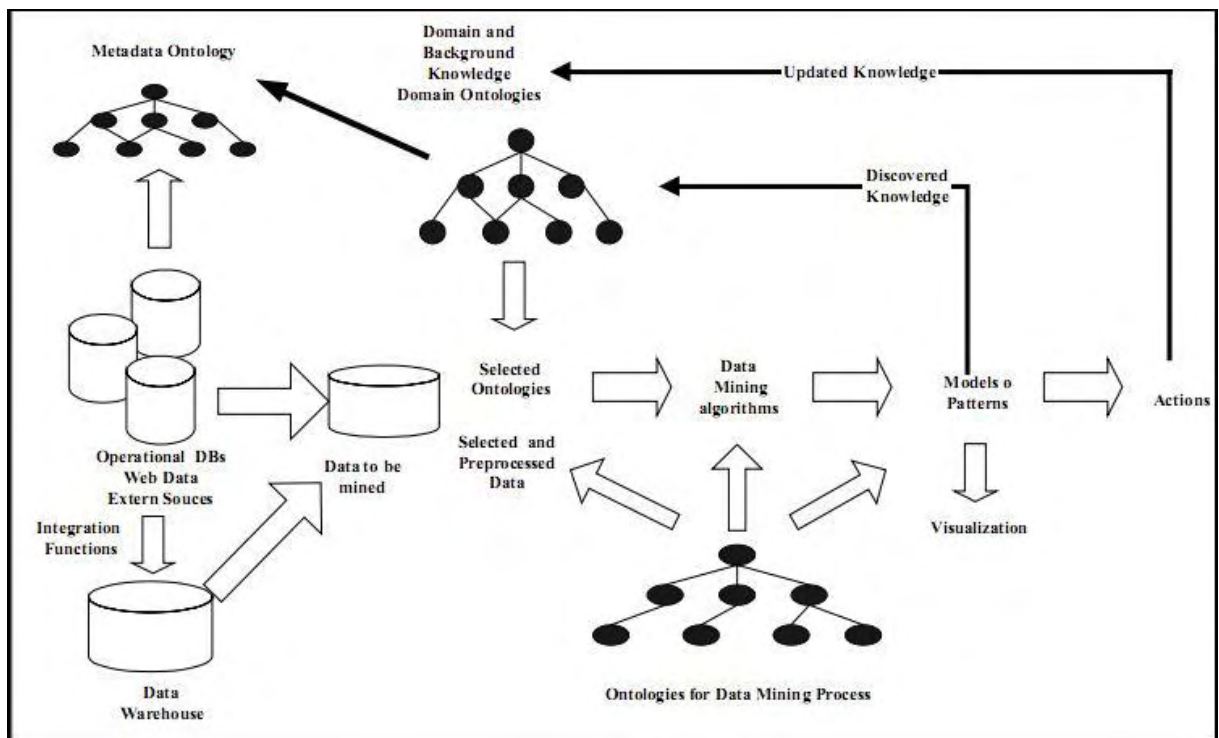


Figura 8 - Framework Genérico de Mineração de Dados com Ontologias.

Fonte: Nigro et al. (2008).

Na figura acima, foram representadas três tipos de ontologias, cada uma com uma determinada função:

- Ontologia de Metadados (*Metadata Ontologies*): Essas ontologias especificam informação sobre os dados do problema.
- Ontologia de Domínio (*Domain Ontologies*): Especificam o conhecimento sobre o domínio da aplicação.
- Ontologia de Mineração de Dados (*Ontologies for Data Mining Process*): Codificam todo o conhecimento prévio sobre o processo, como por exemplo, os melhores algoritmos de acordo com as variáveis do problema.

De certa forma os pesquisadores da área tentam otimizar uma das etapas do processo de KDD com o uso das ontologias.

2.5 Estratégias de Apoio a Seleção de Atributos

As ontologias apresentam uma grande capacidade para armazenar informações que podem ser inferidas por ferramentas específicas. Essas informações podem ser utilizadas de

forma a especificar detalhes sobre determinado domínio e permitir com isso uma grande análise semântica. Já a descoberta de conhecimento em banco de dados possibilita analisar um grande conjunto de dados e obter novos conhecimentos que antes não eram conhecidos ou até mesmo fáceis de serem visualizados. Desta forma, é possível aliar a capacidade semântica das ontologias com a eficiência em descobrir novas informações existentes nas ferramentas de mineração de dados. Essa união pode ser feita de duas formas específicas: partindo da mineração de dados para as ontologias, ou de forma inversa, das ontologias para a mineração de dados (tópico 2.4). No âmbito da mineração de dados, especificamente na tarefa de seleção de atributos, existe uma grande demanda por novas tecnologias que auxiliem os especialistas a melhorar a qualidade dos atributos selecionados e também permitir a diminuição do tempo gasto para executar essa tarefa.

2.5.1 O Problema da Seleção de Atributos

Devido à grande importância que tem se dado ao processo de Descoberta de Conhecimento em Bases de Dados nos últimos anos, também foi preciso incrementar proporcionalmente a necessidade de um suporte formal para realização das fases do processo de KDD. Neste caso, a informação necessária para esse suporte refere-se tanto as tarefas existentes no processo como os dados que se desejam analisar. As ontologias podem trazer grandes benefícios praticamente para todas as fases do processo de KDD, seja auxiliando na escolha de algoritmos de Mineração de Dados, na elaboração de consultas mais eficientes, ou na visualização de novas regras de associação etc.

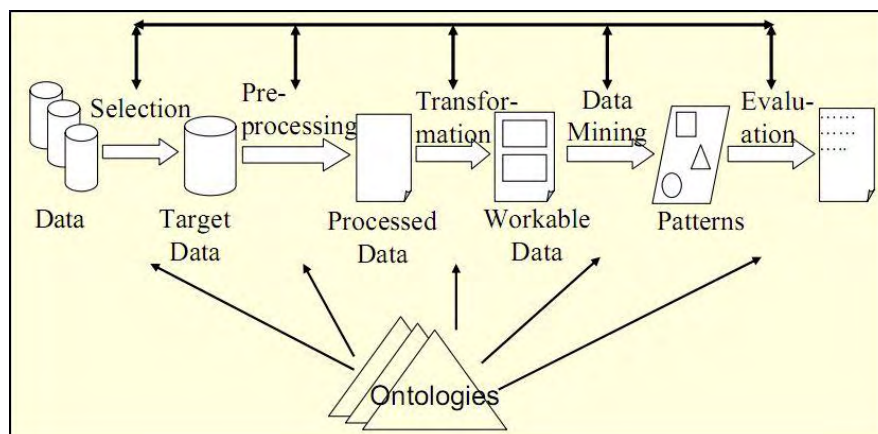


Figura 9 - Ontologias nas Etapas de KDD.
Fonte: (LIN, 2009).

A Figura 9 ilustra a possível utilização de ontologias em todas as fases do processo de KDD, onde cada fase utiliza a ontologia de uma maneira específica. Como visto no capítulo 2, existem trabalhos que utilizam *frameworks* integrando ontologias em todo o processo, e existem também aqueles que utilizam apenas em uma fase específica. Nesse trabalho é focado o uso das ontologias em uma fase anterior a Mineração de Dados, mais especificamente na Seleção de Atributos. A introdução de ontologias nessa fase tem como objetivo principal melhorar a qualidade dos atributos selecionados, levando em conta que essa fase do processo de Descoberta de Conhecimento em Bases de Dados é essencial para todas as fases que virão posteriormente.

A Seleção de Atributos permite, por exemplo, a ordenação dos atributos segundo algum critério de importância, a redução da dimensionalidade do espaço de busca de atributos, remoção de dados contendo ruídos (valores divergentes, fora do padrão, também conhecidos como *outliers*) entre outros. Intuitivamente, quanto maior o número de atributos em uma base de dados, maior o poder discriminatório do classificador e a facilidade de extrair modelos de conhecimento da base. Porém, levando esse conceito em termos práticos, isso nem sempre é verdade. Em primeiro lugar, existe o problema da dimensionalidade, fazendo com que o tempo computacional aumente de acordo com o número de atributos presentes, pois muitos dos métodos são gulosos (fazendo apenas escolhas ótimas locais) e não trabalham com otimização do conjunto de atributos selecionados. Além disso, atributos irrelevantes podem confundir o algoritmo de aprendizagem e esconder as distribuições de pequenos conjuntos de atributos realmente relevantes.

Para selecionar alguns atributos e descartar outros são necessários critérios que avaliem a importância de cada atributo no contexto em questão. Os vários critérios de importância de atributos existentes na literatura levam em conta medidas de distância, consistência ou informação, como também existem outros critérios que são fundamentados em medidas de dependência (HALL; HOMES, 2003).

2.5.2 Critérios Clássicos para Avaliação de Atributos

Para selecionar um subconjunto ótimo de atributos é necessário definir o que significa um atributo ser bom, ou seja, ter determinada importância em relação ao contexto

especificado. A questão da avaliação é complexa e multidimensional (LIU; MOTODA, 1998), e pode ser considerada em termos de:

- Os atributos selecionados trarão alguma melhora na precisão do classificador;
- Os atributos selecionados irão auxiliar a simplificar o modelo construído, de modo que ele seja mais compreensível.

Dessa forma, a definição da importância geral de um atributo pode ser dita como:

- ❖ Um atributo é importante se quando removido a medida de importância considerada em relação aos atributos restantes é deteriorada.

As principais medidas clássicas para verificar a importância de um atributo são:

Medidas de Informação: possibilitam determinar o ganho de informação a partir de um atributo. Esse ganho é definido como sendo a incerteza *a priori* e a incerteza *a posteriori*. Dessa forma, um atributo X_i é preferido ao atributo X_j se o ganho de informação a partir do atributo X_i é maior que a partir do atributo X_j . Esse tipo de verificação pode ser feita usando cálculo de entropia.

Medidas de Distância: podem ser chamadas de medidas de separabilidade ou discriminação. Se estiver comparando duas classes, um atributo X_i é preferido a X_j se X_i provê uma diferença maior que X_j entre as probabilidades condicionais das duas classes. A Distância Euclidiana pode ser usada para esse tipo de medida.

Medidas de Dependência: qualificam a habilidade de prever o valor de uma variável (atributo) a partir do valor de outra, ou seja, quantifica o quanto duas palavras estão associadas ou correlacionadas uma com a outra. Nesse tipo de verificação, pode ser citado o exemplo da comparação entre um atributo e uma classe de uma ontologia (método esse que será utilizado na metodologia deste trabalho para selecionar atributos com ontologias).

2.5.2.1 Distância de Levenshtein (Edit Distance)

A distância de Levenshtein, proposta pelo russo *Vladimir Iosifovich Levenshtein*, permite comparações que efetuam operações de inserção, deleção, substituição sobre *Strings* e ainda permite que essas *Strings* tenham tamanhos diferentes. Dentre os algoritmos existentes para o propósito de comparar strings (*Hamming distance*, *Episode distance*, *Longest common subsequence distance*), a distância de Levenshtein foi considerada a mais completa. Dessa

forma, foi a opção mais eficiente para efetuar comparações sintáticas entre *Strings* e resultar seus níveis de igualdade.

O uso da distância de Levenshtein não permite uma análise semântica, porém pode dar uma visão ao especialista de termos que apresentam grafias parecidas entre a ontologia e os atributos que estão sendo analisados e pode ser usada na etapa principal deste trabalho que será comentada nos próximos capítulos: mapeamento entre as classes da ontologia e os atributos do banco de dados. Uma descrição genérica do algoritmo pode ser feita da seguinte forma:

- 1 – Recebe dois nomes de entrada;
- 2 – Cria um vetor bidimensional do tamanho dos nomes recebidos;
- 3 – Captura o menor valor das seguintes operações até percorrer todo o vetor:

Operação 1: armazena o valor da linha anterior à atual acrescida de uma unidade;

Operação 2: armazena o valor da coluna anterior à atual acrescida de uma unidade;

Operação 3: armazena o valor da diagonal anterior à atual acrescida de 0 se os caracteres forem iguais ou 1 se forem diferentes;

- 4 – Retorna o último valor (linha x coluna) do vetor multidimensional.

Para um melhor entendimento, um exemplo prático comparando o número de operações necessárias para transformar a palavra *Isquemia* em *Infarto*:

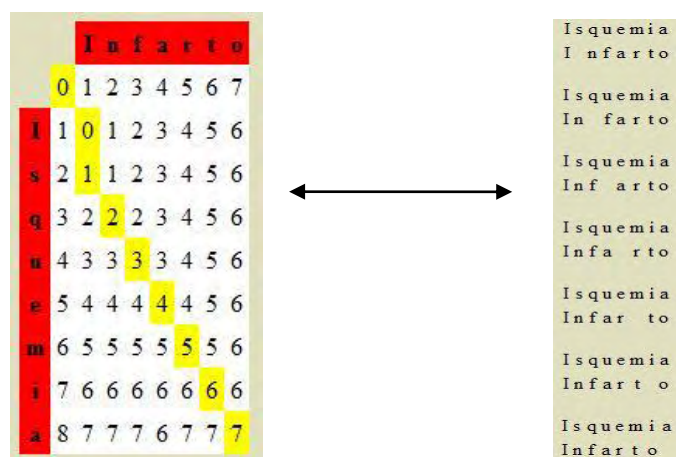


Figura 10 – Distância de Levenshtein na prática.
Fonte: Pesquisa própria.

Na comparação acima, a cada operação, o valor 1 é colocado para caracteres diferentes e 0 para os iguais. O processo é efetuado iterativamente e os valores calculados de acordo com a descrição do algoritmo. Esse algoritmo é utilizado para verificar erros de grafia em documentos do Word, e está inserido na verificação que o Google faz ao digitar uma palavra incorreta. O funcionamento específico geralmente é acrescentado de um dicionário ou banco de dados onde serão feitas as comparações. Neste projeto, no lugar de existir um dicionário, será utilizada a ontologia como base de informações.

2.5.3 Avaliação Inicial da Seleção de Atributos

Para selecionar atributos em um determinado conjunto de dados existem duas abordagens adotadas pelas ferramentas de mineração de dados: a primeira é uma abordagem dependente de modelo, ou seja, *Wrapper* (Seção 2.2.3.2); os algoritmos de aprendizado influirão também na escolha dos atributos e nesse caso os melhores atributos para o algoritmo KNN (*k-nearest neighbor*) e Redes Neurais podem não ser os melhores para as árvores de decisão (John et al., 1994). Evidenciando que a escolha do método de seleção de atributo também deverá levar em consideração o algoritmo que será utilizado posteriormente. A outra abordagem utiliza filtros, onde nesse caso, os atributos são selecionados de maneira independente, considerando apenas características dos mesmos, sem qualquer opinião de informações do algoritmo de aprendizado, como mostrado abaixo:

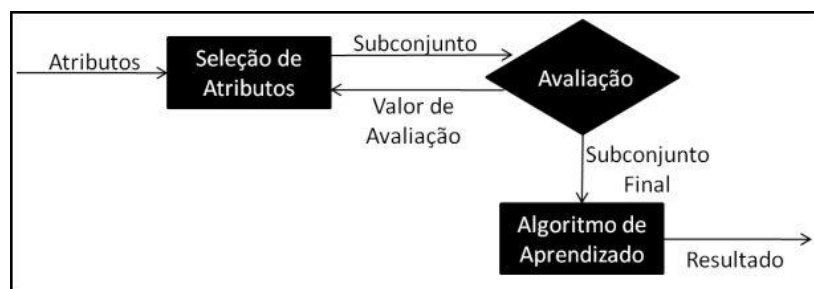


Figura 11 - Uso de Filtros para Seleção de Atributos.
Fonte: Adaptado de LIU et al. (2005).

No modelo que utiliza o *Wrapper*, o algoritmo de aprendizado é usado no processo de busca, onde cada subconjunto selecionado é aplicado no algoritmo utilizado. Dependendo do desempenho do algoritmo de aprendizado será adicionado ou não o subconjunto em questão.

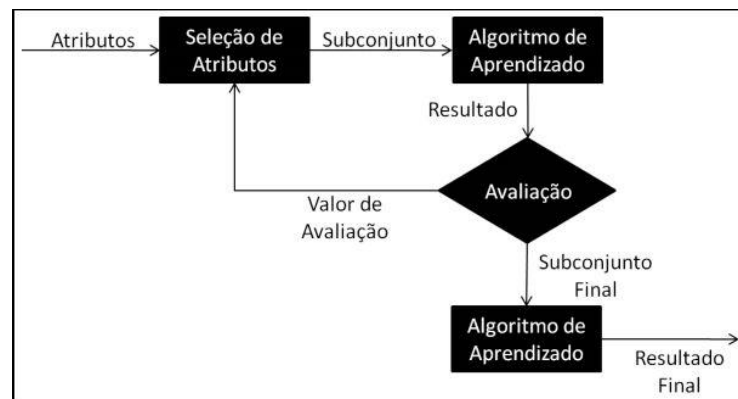


Figura 12 - Uso de Wrapper para Seleção de Atributos.
 Fonte: Adaptado de LIU et al. (2005).

A avaliação de cada subconjunto é relacionada ao desempenho do algoritmo de aprendizado, fazendo a seleção daquele que apresentar melhor resultado. De forma geral, os *Wrappers* tendem a apresentar melhores resultados de seleção que os filtros. Porém, apresentam um custo computacional maior e nesse trabalho a seleção de atributos não evidencia o algoritmo que será utilizado e se enquadra melhor como uma abordagem independente de modelo (filtros).

2.6 Considerações Finais

Este capítulo tratou de alguns conceitos relativos à Descoberta de Conhecimento em Bases de Dados (KDD) e as Ontologias. O processo de KDD foi abordado evidenciando suas diversas fases e mostrando suas características principais. A etapa de Seleção de Atributos foi evidenciada, mostrando seus principais conceitos e técnicas para se reduzir o conjunto de atributos inicial do problema. As ferramentas mais utilizadas para efetuar a Descoberta de Conhecimento em Banco de Dados foram discutidas e avaliadas sob suas características mais evidentes. As ontologias também foram discutidas e tratadas em seu aspecto formal, mostrando classificações e algumas linguagens usadas em sua construção.

O próximo capítulo irá especificar melhor os problemas da seleção de atributos e definir estratégias sintáticas e semânticas que podem ser incorporadas no processo de seleção de atributos.

3 TRABALHOS RELACIONADOS

3.1 Introdução

Existem diversos trabalhos na literatura sobre ontologias e KDD, porém a maioria deles não aborda os dois temas de forma cooperativa. As vantagens do uso das ontologias para auxiliar nas diversas fases do processo de KDD já foi explicada em tópicos anteriores, onde também são mostradas as formas de aplicá-las as diversas etapas que compõem o processo.

Utilizando ontologias e KDD em uma abordagem voltada para Bancos de Dados Geográficos, Bogorny et. al. introduz o conceito de mineração de padrões geográficos e regras de associações espaciais freqüentes em Bancos de Dados Geográficos. O trabalho é referenciado com o título: *Enhancing the Process of Knowledge Discovery in Geographic Databases Using Geo-Ontologies* (NIGRO; CÍSARO; XODO, 2008):. Sua fundamentação consiste em evitar que padrões descobertos e que são triviais, não interessantes e já óbvios para o especialista não tenham que ser descobertos novamente pelas ferramentas de mineração de dados. A proposta desse trabalho mostra como a grande quantidade de conhecimento existente nas Geo-Ontologias pode ser usado em conjunto com a descoberta de conhecimento em banco de dados e melhorar o desempenho dos algoritmos de mineração.

Outro trabalho com o título *An ontology driven approach for knowledge discovery in Biomedicine* (GOTTGTROY; KASABOV; MACDONELL), especifica o uso de bio-ontologias para compartilhar conhecimento entre diferentes comunidades. Para isso foi criada uma ontologia multidimensional que conecta diversas áreas da ciência e permite a aplicação da mineração de dados nas diferentes áreas. A proposta desse trabalho utiliza uma abordagem mais abrangente, e não segue as metodologias citadas no tópico 2.4, onde o processo caminha de KDD para ontologias ou de forma inversa, faz a junção e aplica a metodologia ontologia para kdd-kdd para ontologia (*Onto4KDD4Onto*).

Mais um trabalho envolvendo ontologias e mineração de dados tem o título: *Context-aware Data Mining using Ontologies* (SINGH; VAJIRKAR; LEE). Eles propõem um *framework* para mineração de dados sensível a determinados contextos e que pode se adaptar as mudanças graças ao uso de uma ontologia utilizada dinamicamente. Para conseguir esse objetivo foram tratados vários contextos como domínio, localização, dados, usuários etc.

Também foram utilizadas ontologias para representar o domínio e para especificar os dados existentes.

Na tabela 1 são mostrados mais alguns desses trabalhos e a fase que eles são aplicados no processo de KDD juntamente com as ontologias.

Fase de KDD	Tipo de Aplicação da Ontologia	Exemplos
Integração Seleção	De modo geral, as ontologias quando utilizadas nessas fases funcionam como repositórios de conhecimento em relação ao domínio aplicado. É a fase de maior aplicação de Ontologias dentro do processo de KDD.	(Silvescu et al., 2001), D2OMapper (Zu et al., 2006), SEMEDA (Köhler et al., 2003), KAON (Bozsak et al., 2002)
Limpeza dos Dados Redução dos Dados Transformação	Aqui existem duas opções para o uso das ontologias: (1) armazenando a informação necessária para transformar a instancia dos dados, (2) armazenando as instancias mediante uma representação formal.	(Phillips y Buchanan, 2001), (Bowers y Ludäscher, 2004), (Kedad y Métais, 2002), ONTOCLEAN (Wang et al., 2005)
Mineração de Dados Interpretação	São utilizadas pelos analistas para uma melhor escolha do algoritmo de mineração, levando em conta os dados e informações disponíveis. Os resultados da análise dos dados também podem ser armazenados nas ontologias, e dessa forma a representação formal do novo conhecimento facilita sua reutilização	PROTEUS (Cannataro et al., 2005), IDEA (Bernstein et al., 2005). Onto4KDD (Gottgroy et al., 2004), LISp-Miner (Svatek et al., 2005), MiningMart (Euler y Scholz, 2004)

Tabela 1- Trabalhos com ontologias e KDD.

A pequena quantidade de trabalhos envolvendo ontologias e KDD nas fases iniciais de preparação dos dados e seleção pode ser visualizada na tabela 1, onde apenas 4 trabalhos

podem ser citados de forma efetiva. As outras fases existentes no processo já foram bastante exploradas em diversos trabalhos na literatura, portanto a exploração de trabalhos relacionados diretamente ao tema de ontologias e KDD na fase de seleção de atributos é bastante restrita. Nos tópicos que seguem, mais alguns trabalhos serão analisados mais detalhadamente, especificando suas principais características.

3.2 D2OMapper

O D2OMapper permite criar um mapeamento entre o esquema do banco de dados e uma ontologia. São utilizadas regras heurísticas baseadas em correspondências conceituais entre o esquema e a ontologia, funcionando então de forma automática (Xu, Zhang, Dong; 2006).

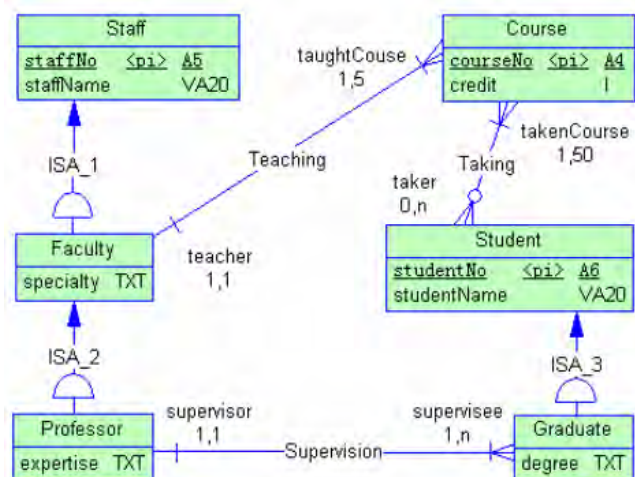


Figura 13 - Esquema utilizado como exemplo no D2OMapper.
Fonte: (Xu, Zhang, Dong; 2006).

A figura 13 ilustra o diagrama ER (Entidade-Relacionamento) utilizado, onde o mesmo pode ser utilizado no processo de mapeamento com a ontologia. Ao final do processo é possível verificar um mapeamento automático com os itens existentes em um arquivo XML.

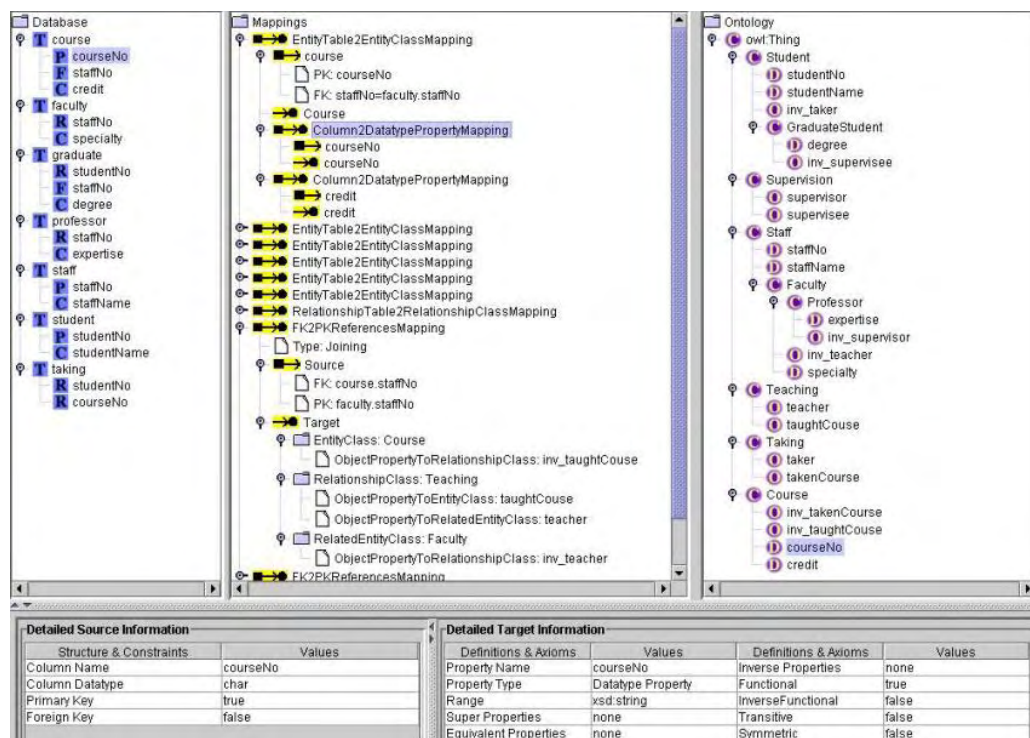


Figura 14 - Mapeamento entre Esquema do Banco de dados e Ontologia pelo D2OMapper.
Fonte: (Xu, Zhang, Dong; 2006).

O D2OMapper permite criar mapeamentos de forma simples entre o esquema do banco de dados e uma ontologia previamente criada para este propósito. A sua utilização e propósito se enquadra na construção e atualização de páginas web dinâmicas, com grande interação com usuários e mudanças externas.

3.3 KAON

KAON consiste em um número de vários módulos que providenciam diversas funcionalidades centradas na criação, armazenamento, recuperação e manutenção de aplicações com uso de ontologias.

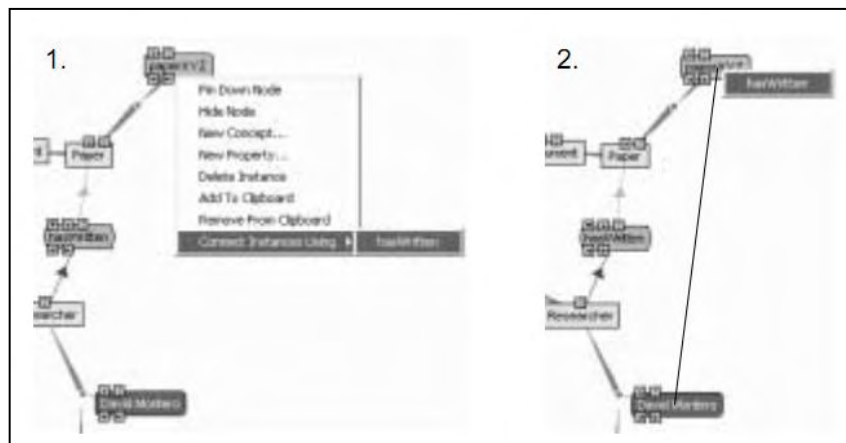


Figura 15 - Comunicação de Ontologia e Base de Dados.

A integração da ontologia e os dados do banco, com o uso do esquema, permite analisar cada regra gerada no processo de mapeamento dos termos presentes na ontologia. A figura 15 mostra o módulo do KAON responsável pela ligação entre os termos que serão analisados, onde finalmente serão mostrados conceitos e conectivos pertencentes à ontologia e o esquema do banco de dados.

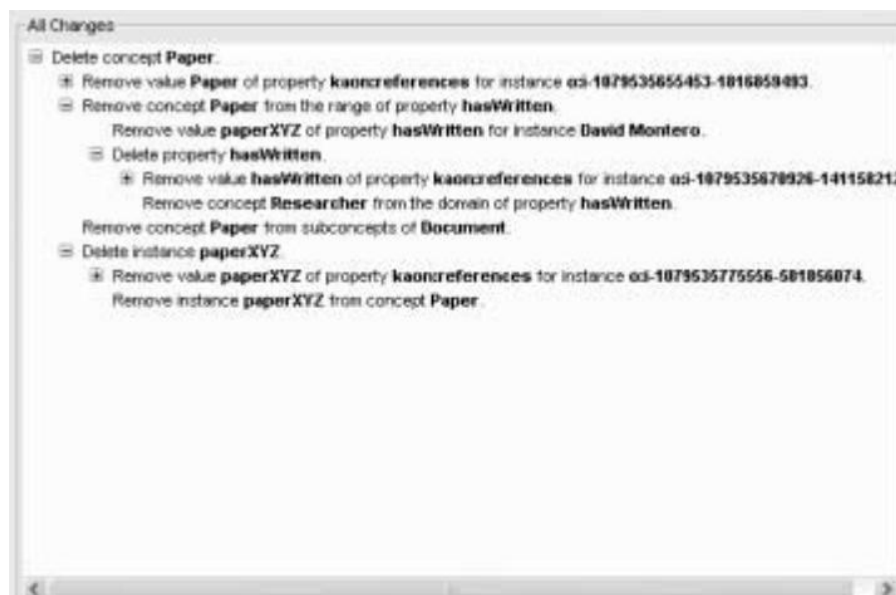


Figura 16 - Mapeamento final do KAON.

É possível o trabalho colaborativo com uma única instancia de ontologia através de um servidor, permitindo que mais de um cliente possa conseguir obter conectivos para mapear se respectivo esquema de banco de dados e sua ontologia.

3.4 Considerações Finais

Nesse capítulo foi analisado o contexto atual do uso das ontologias com KDD e alguns trabalhos relacionados. A análise introdutória mostra claramente cada fase do processo de KDD e vários trabalhos que se dedicaram a explorar essa fase do processo.

Por final foi analisado de forma mais específica, dois trabalhos que exploram especificamente a fase de seleção de atributos na junção das ontologias e KDD.

4 UMA ABORDAGEM SEMÂNTICA PARA SELEÇÃO DE ATRIBUTOS NO PROCESSO DE KDD

4.1 Introdução

Esse trabalho tem propósito de desenvolver uma ferramenta que auxilie o usuário ou especialista em KDD (ou Mineração de Dados) na etapa de Pré-Processamento, especificamente na Seleção de Atributos (Seção 2.2.2). Esses atributos são capturados e tratados a partir de um domínio semântico mais genérico especificado em uma ontologia. A utilização de uma ontologia de domínio propicia a utilização de um determinado conhecimento prévio que pode ser utilizado para auxiliar na análise dos dados existentes (neste caso, os atributos obtidos). O principal aspecto positivo dessa abordagem é possibilitar que o especialista possa fazer escolha de atributos por categorias relacionadas semanticamente, além de outras avaliações só possíveis depois de uma associação entre a ontologia e os atributos.

Foi desenvolvida uma ferramenta protótipo, chamada de SAAO (Seleção de Atributos com Auxílio de Ontologias), que busca auxiliar o analista dentro do processo de KDD na eliminação de atributos que pouco colaboram nas etapas posteriores ou até mesmo podem interferir nos resultados encontrados na Mineração de Dados. Serão evidenciadas de forma geral três maneiras para melhorar o conjunto de atributos escolhidos e auxiliar o especialista nessa fase de Pré-Processamento. São elas:

- Construção de uma categorização dos atributos com auxílio de uma ontologia de domínio e suporte ao processo com a análise sintática dos termos;
- Identificar possíveis redundâncias e realizar redução de dimensionalidade nos dados;
- Identificar e explorar atributos relacionados semanticamente.

Os atributos tratados inicialmente na fase de Seleção podem ser comparados as classes das ontologias e agrupados em categorias conforme a organização semântica existente na mesma. Dependendo da forma de representação da Ontologia, especificamente o tipo de linguagem OWL escolhida para construí-la, pode ser feito um mapeamento com conceitos, instâncias, propriedades, etc. Para fazer essa análise foi utilizada uma visão da ontologia

centrada em suas classes, permitindo assim uma maior facilidade para o usuário efetuar comparações com os atributos analisados.

4.2 Especificação do SAAO

4.2.1 Considerações Iniciais

Como a solução proposta nesse trabalho envolve ontologias e mineração de dados, qualquer ferramenta desenvolvida deve ter a capacidade de trabalhar com esses dois tipos de tecnologias. A idéia principal para ligação entre os dois conceitos é centrada no mapeamento entre os atributos obtidos na base de dados e as classes existentes na ontologia. Incluindo uma inferência básica na taxonomia da ontologia é possível a visualização de uma hierarquia de classes convencional acrescida de algumas mudanças proporcionadas por relacionamentos entre as classes. Tendo essa visão de atributos e classes, o especialista pode fazer o mapeamento de forma manual ou com o auxílio de um módulo de distância de edição (*Edit Distance*) entre os termos. Quando o mapeamento é concluído, além de possibilitar seleção de atributos categorizados, ainda permite visualizar redundâncias e efetuar alguns cortes em determinados níveis hierárquicos (redução de dimensionalidade).

A figura 15 ilustra a interação do especialista com a interface gráfica da ferramenta proposta:

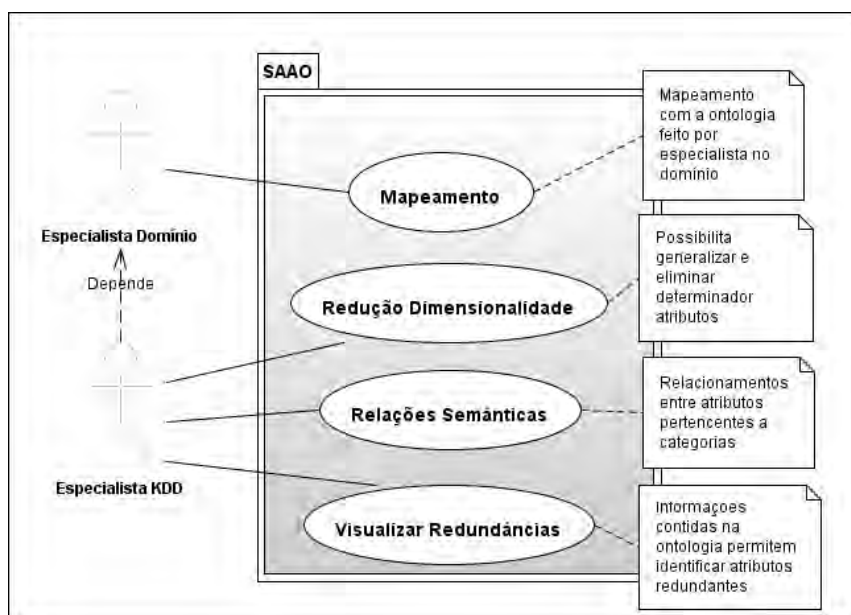


Figura 17 - Diagrama de Caso de Uso do SAAO

De uma forma genérica, para desenvolver a ferramenta foi usada a linguagem Java e as ontologias foram construídas com a ferramenta Protégé.

4.2.2 Arquitetura

Com o objetivo de permitir uma análise semântica entre a ontologia e os atributos candidatos à seleção, foi desenvolvida uma ferramenta protótipo que auxilia o especialista em KDD na fase de seleção de atributos com auxílio de ontologias. Foi utilizada a linguagem Java, a IDE Netbeans e o banco de dados PostgreSQL. Com a linguagem Java foi criado todo o protótipo, usando o ambiente de desenvolvimento Netbeans e os dados para avaliação foram armazenados no banco de dados PostgreSQL. O sistema foi dividido em cinco camadas funcionais que são diretamente ligadas e utilizam os resultados obtidos na camada anterior.

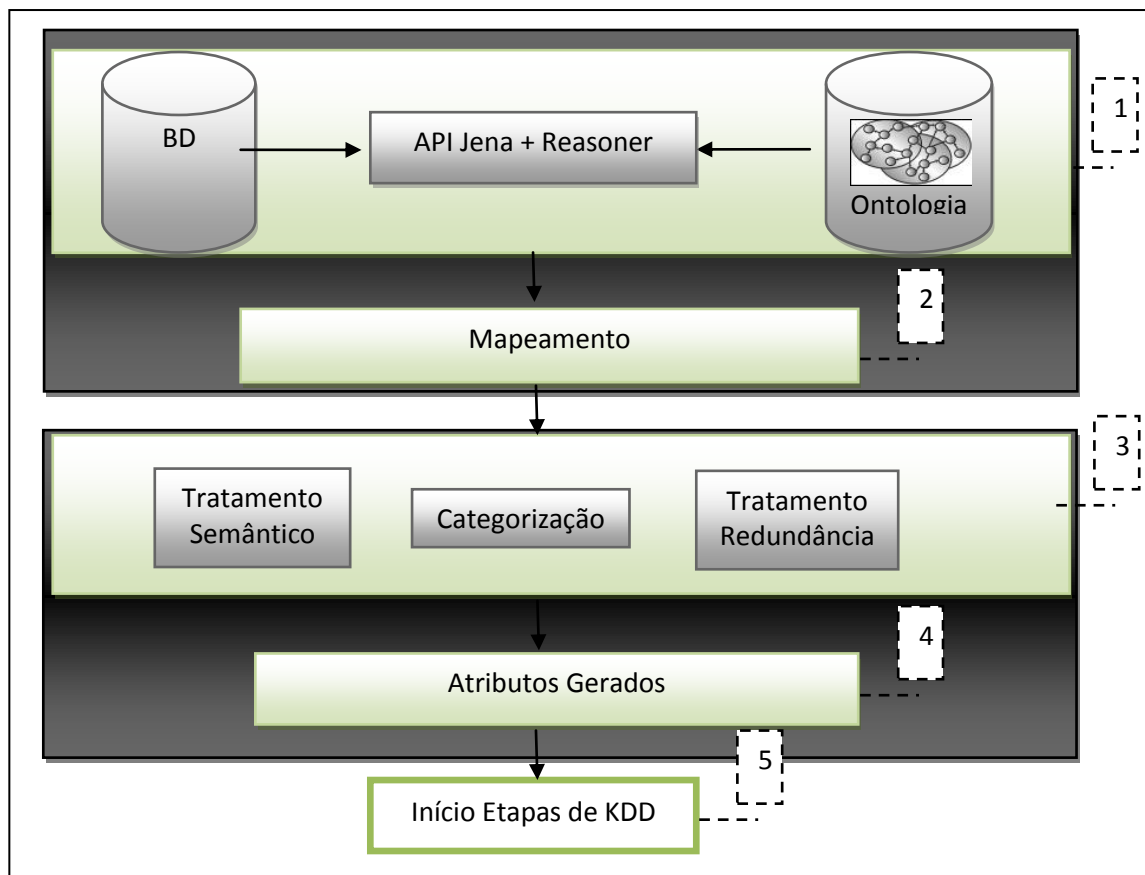


Figura 18 - Arquitetura Básica do Sistema.

A Figura 16 pode ser explicada da seguinte forma:

- **Camada 1:** Será responsável por capturar os atributos existentes no banco de dados e as classes da ontologia. O banco de dados deve fornecer todas as colunas em uma estrutura dimensional única. Provavelmente o especialista precisa agrupar o esquema do banco de dados em uma única tabela de avaliação. As ferramentas de mineração de dados como o Weka, na maior parte das vezes trabalham com um arquivo de extensão própria e organizado também em única estrutura. Então, para a finalidade de mineração de dados, a tabela única pode trazer algumas facilidades de transição para essas ferramentas. Para consultar as informações contidas na ontologia foi utilizada a API Jena em conjunto com o motor de inferências Pellet (referencia, ano).
- **Camada 2:** Essa etapa deve ser feita preferencialmente por um especialista no domínio que estiver sendo estudando. O usuário deverá ligar ou mapear os atributos às classes que melhor os represente semanticamente. Como mostrado no capítulo anterior, a ontologia foi construída de uma forma genérica e com capacidade para permitir uma organização categórica dos dados. O protótipo SAAO ainda conta com um sistema para calcular a similaridade entre a escrita do atributo e as várias classes da ontologia, possibilitando um suporte ao analista no momento do mapeamento de cada atributo.
- **Camada 3:** Essa é a camada de principal ação semântica do sistema, pois com as informações armazenadas após o mapeamento é possível visualizar de forma categorizada os atributos mapeados. Existe um módulo para o especialista fazer análises para cada atributo (redimensionar os dados, identificar e explorar atributos relacionados semanticamente, etc) e ainda possibilita uma melhor visão de atributos possivelmente redundantes no contexto estudado.
- **Camada 4:** Finalmente após todas as análises o usuário pode obter os atributos que servirão de entrada para outros processos de pré-processamento ou mesmo a mineração de dados. No protótipo é possível obter uma saída

com a consulta que será inserida no Weka para captura dos atributos selecionados no banco de dados.

- **Camada 5:** Apenas representa os atributos selecionados e prontos para serem usados na mineração de dados.

Com métodos específicos é possível manipular as ontologias e detalhar certos aspectos que podem contribuir na seleção de atributos. O funcionamento e detalhamento da ferramenta poderão ser mais bem entendidos nos próximos tópicos.

4.3 Protótipo para Seleção de Atributos com Auxílio de Ontologias (SAAO)

4.3.1 Contextualização do Problema

As metodologias convencionais para seleção de atributos, inclusive as aplicadas em ferramentas de mineração como o Weka, utilizam medidas matemáticas que buscam o grau de relacionamento entre os atributos avaliados. Como já discutido em vários tópicos anteriores, esses métodos não levam em consideração o domínio tratado e, principalmente, não possuem a capacidade de fazer análises levando em consideração conhecimentos prévios acerca do desse domínio. Outro fato é que a supressão de colunas (atributos) é muito mais delicada que a supressão de uma linha; então cada atributo deve ser bem avaliado antes de ser retirado do espaço de análise, dificultando mais ainda a tarefa que busca reduzir o espaço vertical dos atributos. Se esses atributos forem irrelevantes e permanecerem durante a análise dos algoritmos de mineração, independente da tarefa executada pode implicar na descoberta de padrões de baixa qualidade.

	C4.5	ReliefF	CFS	FCBF	CBF	SAAO
Avaliação Individual	X	X		X		
Avaliação de Subconjuntos			X	X	X	
Medida de Informação	X		X	X		
Medida de Distância		X				X
Medida de Dependência						
Medida de Consistência					X	
Avaliação de Redundância						X

Avaliação de Domínio						X
----------------------	--	--	--	--	--	---

Tabela 2 - Características dos algoritmos para seleção de atributos.

Fonte: Adaptado de Lee (2005).

Na tabela 2 são feitas algumas avaliações para cada tipo de característica presente em alguns dos algoritmos mais utilizados para a seleção de atributos. O SAAO não é especificamente um algoritmo, mas trabalha em conjunto com os algoritmos existentes para seleção de atributos, proporcionando uma melhor avaliação das redundâncias e principalmente relacionamentos semânticos no domínio estudado. O algoritmo C4.5 é o sucessor do antigo ID3, e apresenta função de classificação na descoberta de conhecimento, atuando na seleção de atributos com auxílio das árvores de classificação. com seleção de atributos através. É importante comentar que a utilização de um método de seleção não exclui outro método, ou seja, mesmo após selecionar atributos com o SAAO o usuário ainda pode tentar refinar seu resultado com os algoritmos já existentes. O mesmo se diz da execução prévia de um algoritmo antes mesmo de se executar o SAAO (o que seria menos eficiente, pois a categorização é mais apurada em um espaço maior de atributos).

4.3.2 Ferramenta Desenvolvida

O usuário pode interagir com o protótipo por meio de uma interface gráfica desenvolvida para visualizar, mapear, categorizar e relacionar atributos semanticamente. Com os resultados prontos o sistema ainda permite visualizar a consulta resultante e disponibilizá-la para o Weka pela área de transferência.

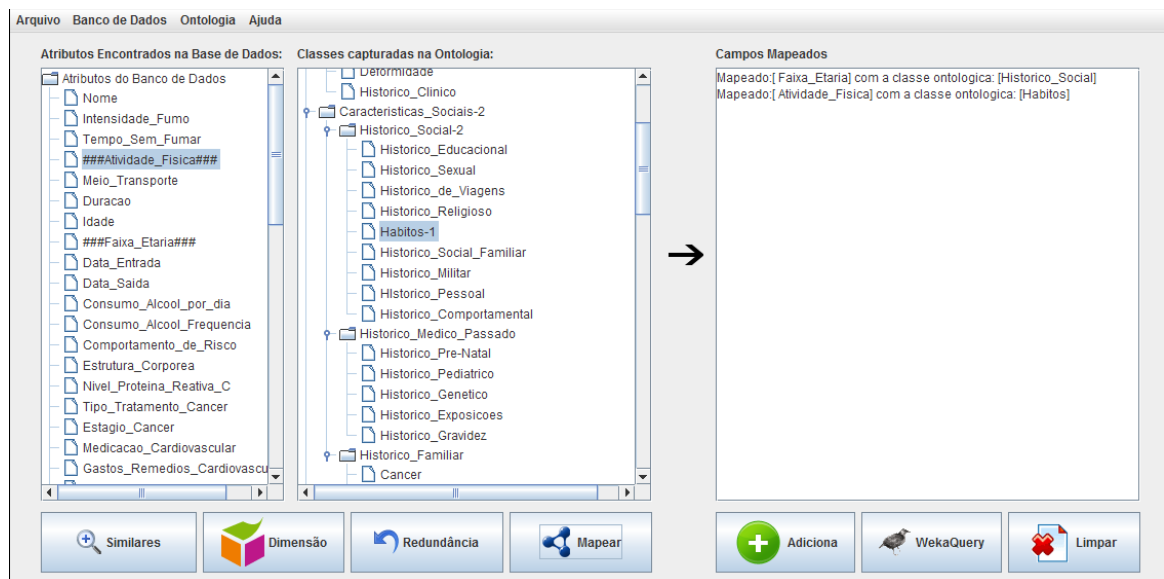


Figura 19 - Interface Gráfica SAAO.

Os atributos são carregados no seu painel respectivo e a classificação hierárquica da ontologia carregada em uma árvore em outro painel. O último painel do lado direito mostra todos os termos que foram e estão sendo mapeados. As funções para cada item da interface gráfica serão explicadas nos próximos tópicos.

4.4 Efetivação Semântica

4.4.1 Adaptação de Ontologia de Domínio

Através da junção de alguns métodos para construção de ontologias, como: Método 101 (NOY, MCGUINESS; 2001), Methontology (FERNANDEZ et al.; 1997), foi possível adaptar ontologias de domínio para o propósito de mapeamento de atributos. Como não existe uma maneira correta de se modelar um domínio – sempre existem várias alternativas, a melhor solução depende da aplicação e possíveis extensões. O desenvolvimento de ontologias não é um processo linear, necessitando de muitas iterações e refinamentos para chegar a um modelo adequado. Um processo de validação com especialistas de domínio e futuros usuários é muito proveitoso no final de cada versão. Como resultado é possível que novas modificações tenham de ser incorporadas à ontologia, fazendo com que as fases iniciais tenham de ser repetidas. Na Figura 18 é ilustrado o método 101 e detalhado cada um dos seus passos para utilização neste trabalho:

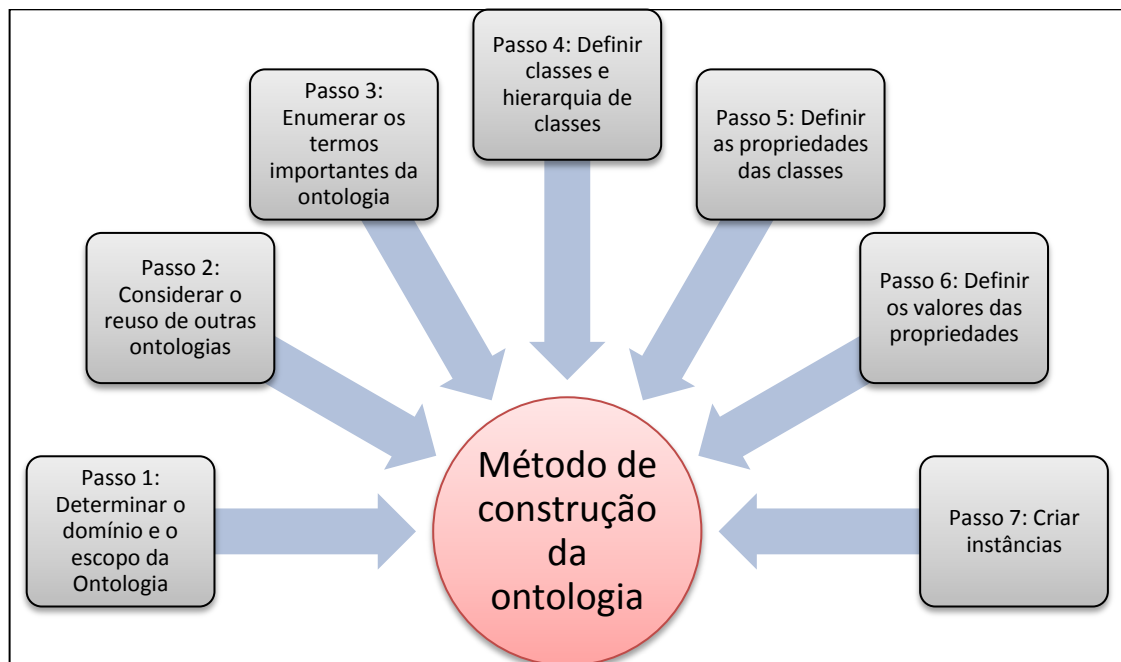


Figura 20 - Método de construção da ontologia.

Seguindo cada um desses passos a ontologia será adaptada para uso na seleção de atributos. Todos os passos serão detalhados na sequência e suas construções demonstradas.

4.4.1.1 Determinar o Domínio de Escopo da Ontologia

Para se determinar o domínio e o escopo da ontologia, os métodos sugerem alguns questionamentos (NOY, MCGUINESS; 2001),:

- Que domínio se deseja cobrir com a ontologia?
- Com que propósito será utilizada a ontologia?
- Para que informações a ontologia de fornecer respostas?
- Quem vai utilizar e manter a ontologia?

Seguindo esses questionamentos, foi necessário escolher um domínio para se trabalhar, mas a ferramenta proposta pode ser aplicada para qualquer domínio, bastando apenas ter a ontologia modelada e a base de dados para fazer mapeamento. Foi escolhido para esse trabalho o domínio médico, pois apresenta uma grande quantidade de pesquisas sobre ontologias e muitas ontologias já prontas para uso e modificação.

O propósito de utilização da ontologia será para auxiliar na seleção de atributos, então ela deve prover algum mecanismo que auxilie o especialista nessa tarefa. De certa forma a ontologia auxiliará o especialista a obter respostas de quais atributos podem ser utilizados em determinado contexto da mineração de dados.

4.4.1.2 Considerar o Reuso de outras Ontologias

Sempre é importante verificar se alguém já codificou os termos da ontologia ou se é possível refinar um modelo existente para o domínio aplicado (NOY, MCGUINESS; 2001). Muitas ontologias estão disponíveis para o domínio médico e existem editores que permitem lidar facilmente com os formalismos existentes nas mesmas e alterá-las sem muitas dificuldades. As bibliotecas consultadas para encontrar ontologias que tivessem utilidade para este trabalho foram: DAML⁴, Swoogle⁵ e os próprios repositórios de ontologias da ferramenta Protégé.

Então, vamos supor que os dados que estão sendo analisados pertencem a um repositório que guarda informações referentes a várias avaliações de pacientes atendidos por unidades móveis (um exemplo poderia ser o SAMU – Serviço de Atendimento Móvel de Emergência). Nesse contexto o especialista de mineração de dados irá fazer algumas suposições sobre esses dados colhidos e obter novos conhecimentos, sejam eles através de regras de associação, classificação ou outras tarefas inerentes a KDD; aplicando os diversos algoritmos embutidos em ferramentas como o WEKA. Independente do algoritmo que será aplicado, pois a ferramenta SAAO utiliza a abordagem de filtros, o especialista precisa selecionar os atributos importantes para sua análise. A ontologia pode ser construída a partir de diversas outras ontologias de forma a abranger esse domínio e possibilitar diversas melhoras na descoberta de conhecimento. Nos exemplos anteriores, foi citado o SAMU, mas o uso pode ser bem mais abrangente e especificar outras tarefas no domínio médico. Portanto, a inserção da ontologia permite que os diversos atributos existentes na estrutura unidimensional analisada (formada pela integração das diversas tabelas presentes em esquema do banco de dados) possam ser organizados em categorias semânticas (mesmo que em nível de inferência taxonômica).

⁴ Encontrada no endereço www.daml.org/ontologies/

⁵ Encontrado no endereço <http://swoogle.umbc.edu/>

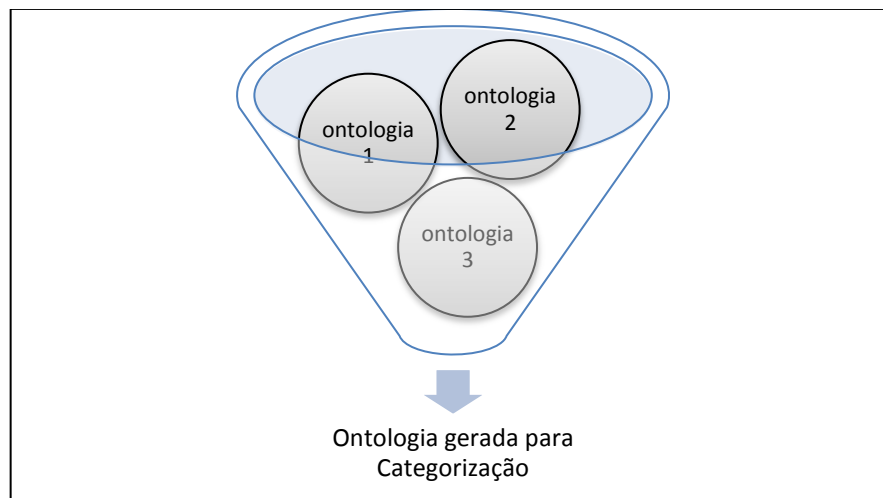


Figura 21 – Construção/Adaptação da Ontologia de Categorização.

A abstração indicada na Figura 19 pode de início parecer com um alinhamento de ontologias, mas a ontologia construída para categorização não evidencia termos comuns em ambas as ontologias. A adaptação da ontologia absorve conceitos que serão úteis na análise de domínio que está sendo feita, ou seja, só serão utilizados conceitos: classes, propriedades, restrições; que forem úteis para a tarefa de categorização.

4.4.1.3 Enumerar os Termos Importantes da Ontologia

É importante fazer uma lista de termos que se deseja definir ou explicar e que serão úteis no contexto da categorização. Devem-se responder as seguintes perguntas:

- Quais os termos que têm utilidade para a análise?
- Quais as propriedades necessárias para serem incluídas nesses termos?

Então, para criar ou utilizar termos de outras ontologias é preciso entender que tipo de análise será feita. Continuando com o exemplo anterior sobre avaliações de pacientes atendidos pelo SAMU, imagine agora que o especialista sabe previamente que existem muitos dados que se enquadram no domínio de cardiologia e muitos pacientes são socorridos com suspeita de I.A.M (Infarto Agudo do Miocárdio). Então, ao analisar as ontologias foram adicionados conceitos que auxiliem na categorização desses termos. Primeiro é preciso verificar que serão especificados diversos problemas ligados ao contexto de doenças cardiovasculares, logo deverá existir uma classe: *doenças_cardiovasculares*. Como serão vistos diversos problemas relacionados a essas doenças também deverão ser acrescentados os principais tipos de doenças que são relatadas em históricos prévios de pacientes, por exemplo:

isquemia do miocárdio, infarto do miocárdio, hipertensão entre outros. O importante nesse tipo de avaliação é a existência de termos genéricos que permitam agrupar os atributos adquiridos do banco de dados.

4.4.1.4 Definir Classes e a Hierarquia de Classes

Como já mencionado em tópicos anteriores, o reuso de ontologias de domínio já existentes ajuda a criar uma nova ontologia para o propósito de categorização. Para adicionar novos termos é seguida uma estratégia *top-down* (topo-para-baixo) que se inicia com conceitos mais gerais e segue através de um processo de decomposição, onde se coloca abaixo do termo mais abrangente os termos mais específicos (subclasses) através de relacionamentos de especialização. Após a definição de toda hierarquia da ontologia (taxonomia), os termos restantes provavelmente representam características (propriedades das classes). Para cada propriedade deve-se determinar a que classe(s) pertence(m).

Para montar a ontologia necessária para análises de categorização foi utilizada a base de dados semântica da *National Library of Medicine*⁶ que contém um acervo com mais de 800.000 conceitos do domínio médico incorporados no Sistema Unificado de Linguagem Médica (*Unified Medical Language System* - UMLS). Dessa forma, foram utilizadas algumas ontologias para avaliação de taxonomia, removidos conceitos inválidos para o domínio e incluídos novos termos obtidos da UMLS. O aspecto mais importante em relação à construção da ontologia é o domínio que ela trata, pois deverá especificar genericamente os atributos analisados. Então, para suprir essa necessidade também foi utilizado um sistema de nomenclatura sistematizada dos termos médicos chamado SNOMED CT⁷.

⁶ Disponível no endereço: <http://www.nlm.nih.gov/>

⁷ Disponível no endereço: http://www.nlm.nih.gov/research/umls/Snomed/snomed_main.html

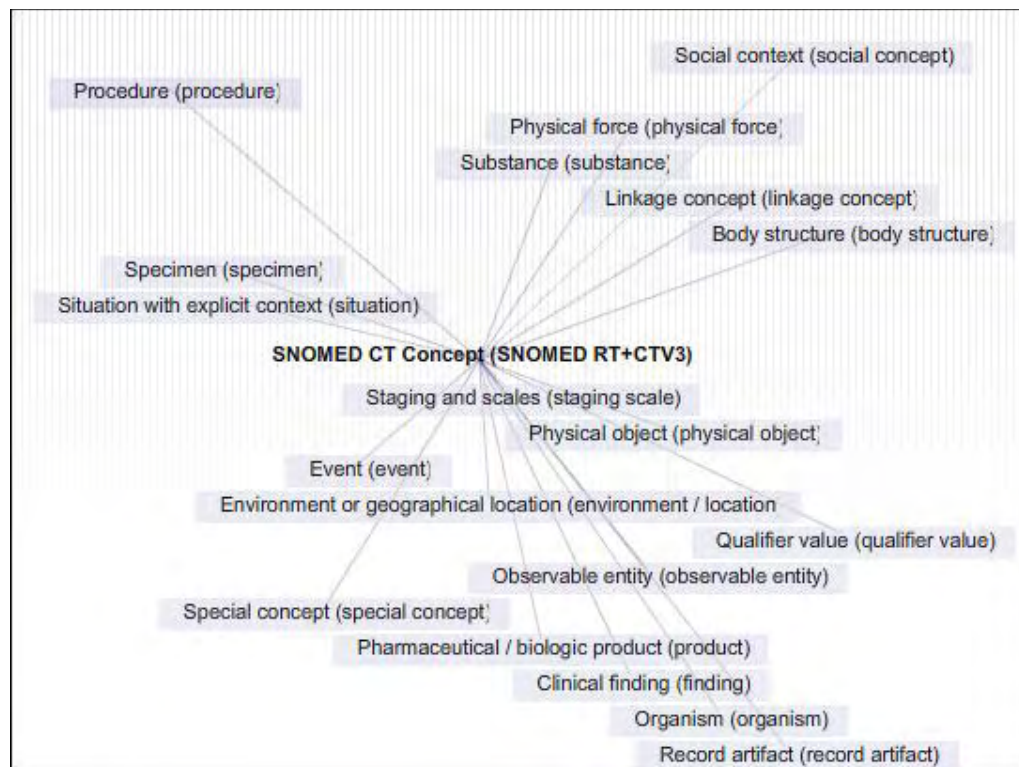


Figura 22 - Nomenclatura sistematizada dos termos médicos.
Fonte: (SnomedCT)

Antes de inserir alguns termos, suas nomenclaturas foram traduzidas e incorporadas no contexto de classes da ontologia. Então, na construção da ontologia a sua taxonomia principal foi derivada dos termos dos vários ambientes de nomenclaturas médicas. A representação taxonômica principal é representada pelas seguintes classes:

- *Diagnósticos*: Inclui alguns exames e procedimentos utilizados em uma consulta ou atendimento médico, auxiliando na identificação de uma doença.
- *Achados Clínicos*: Agrupa algumas manifestações objetivas definidas pelo médico ou subjetivas definidas pelo paciente.
- *Procedimentos*: Ações executadas para determinados diagnósticos.
- *Características Sociais*: Informações pessoais, incluindo atividades, hábitos e outros temas.

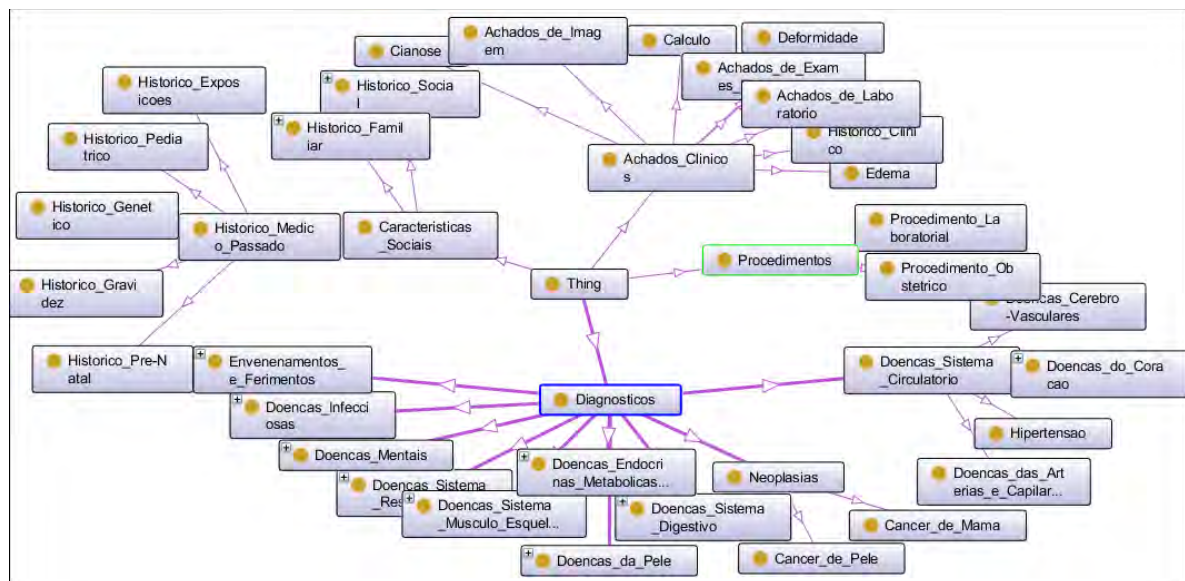


Figura 23 - Hierarquia Principal.

Na Figura 21 é possível perceber que todas as classes principais estão ligadas a classe *Thing*, pois é superclasse padrão em qualquer Ontologia OWL. Essas classes definidas tentam agrupar o maior número possível de características genéricas usadas no domínio médico.

4.4.1.5 Outros Passos Incorporados na Construção da Ontologia

Para seguir com a estruturação da ontologia também é necessário adicionar propriedades e restrições. Para acrescentar algumas propriedades ao invés de classes deve-se entender a sua importância para o domínio em questão. Se a ontologia for do domínio médico, porém, específica para a especialidade da cardiologia, algumas classes podem se tornar propriedades ou propriedades formarem classes. Por exemplo, a ontologia inicial é genérica no domínio médico, logo apresenta uma classificação das diversas categorias de doenças existentes em várias especialidades. Se ela fosse especificar a especialidade cardiologia ela deixaria de representar vários problemas de outras especialidades e especificaria mais as suas características. Em uma ontologia mais genérica o conceito de doenças cardiovasculares apresenta apenas as categorias mais genéricas de doenças e distúrbios, porém a ontologia mais específica detalha a classe doenças cardiovasculares e especifica melhor seus conceitos e propriedades.

Outro tipo de informação relevante neste trabalho é a definição de classes equivalentes e a adição das propriedades de igualdade. Com essas informações é possível usar o motor de inferência para capturar todos os conceitos relacionados a determinada classe e auxiliar na visualização de redundâncias. Na ontologia para definir que uma classe equivale a outra se deve especificar o conceito *owl:equivalentClass*, como no exemplo abaixo:

```
<owl:Class rdf:about="Cavidade Atrial">
  <owl:equivalentClass>
    <owl:Class>
      <rdf:Description rdf:about="Cavidade Atrial Esquerda"/>
      <rdf:Description rdf:about="Cavidade Atrial Direita"/>
    </owl:Class>
  </owl:equivalentClass>
</owl:Class>
```

Figura 24 - Classes Equivalentes.

Dizer que classes são equivalentes não significa necessariamente que sejam iguais, pois esse tipo de avaliação só é possível com a propriedade *owl:sameAs* (propriedade de igualdade). Classes equivalentes apresentam a mesma extensão de conceitos, ou seja, na figura 19 tanto a classe que representa a cavidade atrial esquerda quanto a que representa a direita apresentam o mesmo conjunto de indivíduos. Já a propriedade de igualdade denota conceitos sinônimos e também pode ajudar na identificação de redundâncias.

Para tirar proveito dos relacionamentos existentes em uma ontologia é preciso de ferramentas que além de manipular os diversos termos contidos internamente, possam tirar alguma conclusão sobre esses relacionamentos. A ferramenta utilizada para esse caso se trata de uma API já comentada em alguns tópicos anteriores chamada JENA, e para realizar o processo de conclusão sobre os relacionamentos ou dizendo de uma melhor forma, inferência. Os aplicativos utilizados se chamam motores de inferência. Eles são utilizados em diversas áreas da inteligência artificial e existem para diversos propósitos: trabalhar com agentes, sistemas especialistas, Inferências em Ontologias, etc.

O nível semântico existente em uma ontologia é passível de análise por meio das ferramentas de inferência, podendo ser avaliado em três níveis: (1) Sem inferência, (2) Inferência fundamentada em RDFS (RDF Schema) e (3) Inferência descrita com OWL. Para o nível (1), veremos exatamente o conjunto de indivíduos e classes sem nenhuma afirmação adicionada, já o nível (2) relacionamentos de subclasses e sub-propriedades são evidenciados. Finalmente, o nível (3) adiciona uma quantidade bem maior de expressividade usando

restrições, classes avançadas e descrições de propriedades. Com essas adições é possível trabalhar em um domínio específico e descobrir relacionamentos que ajudem a extrair diversas conclusões. Para exemplificar, três instâncias (indivíduos) da ontologia criada serão inferidos nos três níveis citados. A sub-hierarquia estudada é a seguinte:

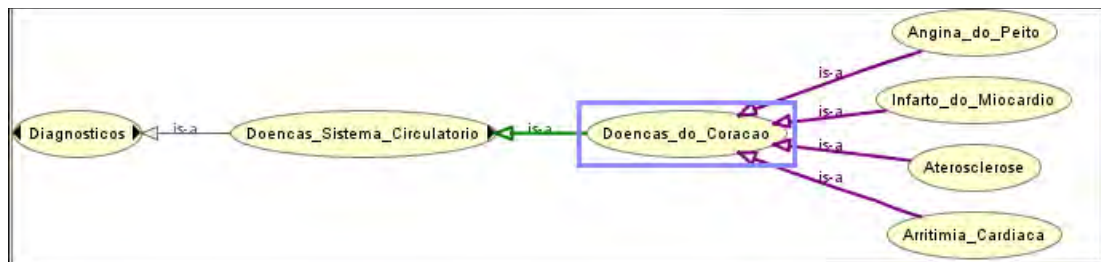


Figura 25 - Sub-hierarquia da Ontologia SAAO.

Para adicionar um relacionamento semântico entre as classes, como discutido em tópicos anteriores, devem-se acrescentar propriedades, restrições, cardinalidade, axiomas etc. Cada instancia tem suas propriedades e pode se relacionar de diversas formas na ontologia, constituindo assim certa interligação semântica.

No primeiro nível de inferência temos o seguinte resultado da listagem de informações de cada individuo.

```
Individual: PacienteAngina2
  TemPredisposicao : :PacienteAngina2
  label : Diagnostico convencional^^http://www.w3.org/2001/XMLSchema#string
  type : :Angina_do_Peito
  type : owl:NamedIndividual
Individual: PacienteAngina1
  TemConjuntoPatologico : :PacienteAngina1
  label : Diagnostico mais raro^^http://www.w3.org/2001/XMLSchema#string
  type : :Angina_do_Peito
  type : owl:NamedIndividual
Individual: PacienteArritmia1
  type : :Arritmia_Cardiaca
  type : owl:NamedIndividual
```

Figura 26 - Indivíduos pertencentes a classes.
Fonte: Pesquisa direta/2010.

Na figura acima, cada indivíduo foi caracterizado com propriedades e relacionamentos que ele apresenta na ontologia. Por exemplo, o individuo *PacienteAngina2* pertence a classe *Angina_do_Peito*, ou seja, é um tipo dessa classe. Apresenta também a propriedade *TemPredisposicao* e outras características. Esse nível não efetuou nenhuma

adição nos dados já existentes. O próximo nível, também conhecido como RDFS (RDF Schema), já consegue mostrar relacionamentos diretos, indiretos e outras propriedades.

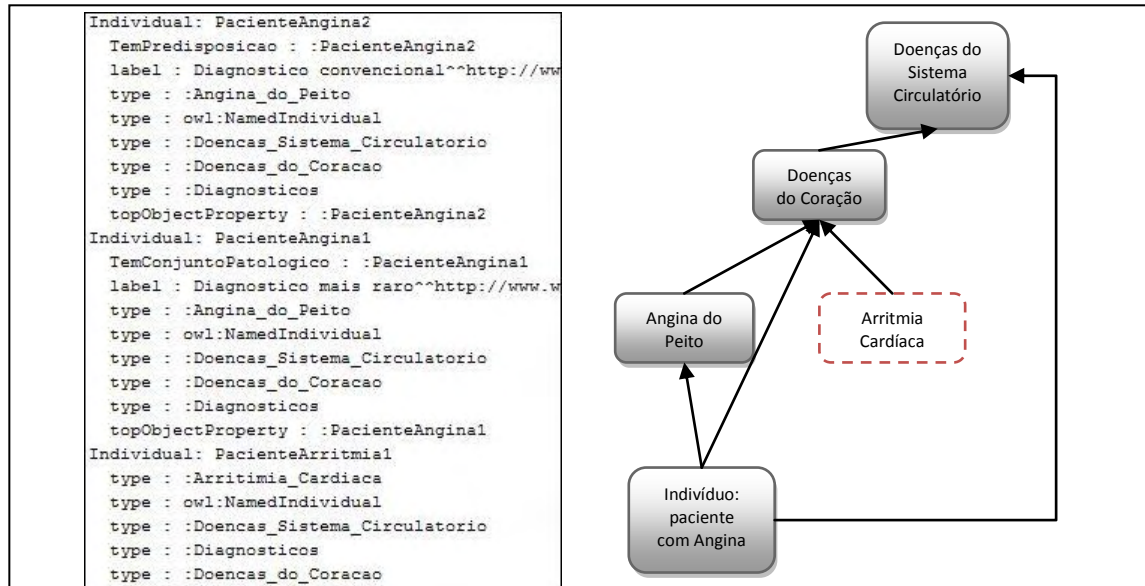


Figura 27 - Indivíduos Pertencentes a classe RDFS.

Fonte: Pesquisa direta/2010.

No tipo de inferência feito acima surgem os relacionamentos diretos e indiretos, ou seja, se a classe *Angina_do_Peito* é um tipo de *Doencas_do_Coracao* e individuo é um tipo dessa classe, logo ele também será um tipo de *Doencas_do_Coracao*. Esse mesmo raciocínio pode ser feito para a ligação direta com a classe *Doencas_do_Sistema_Circulatorio*. Por fim, o nível de inferência mais eficiente e com o maior número de itens:

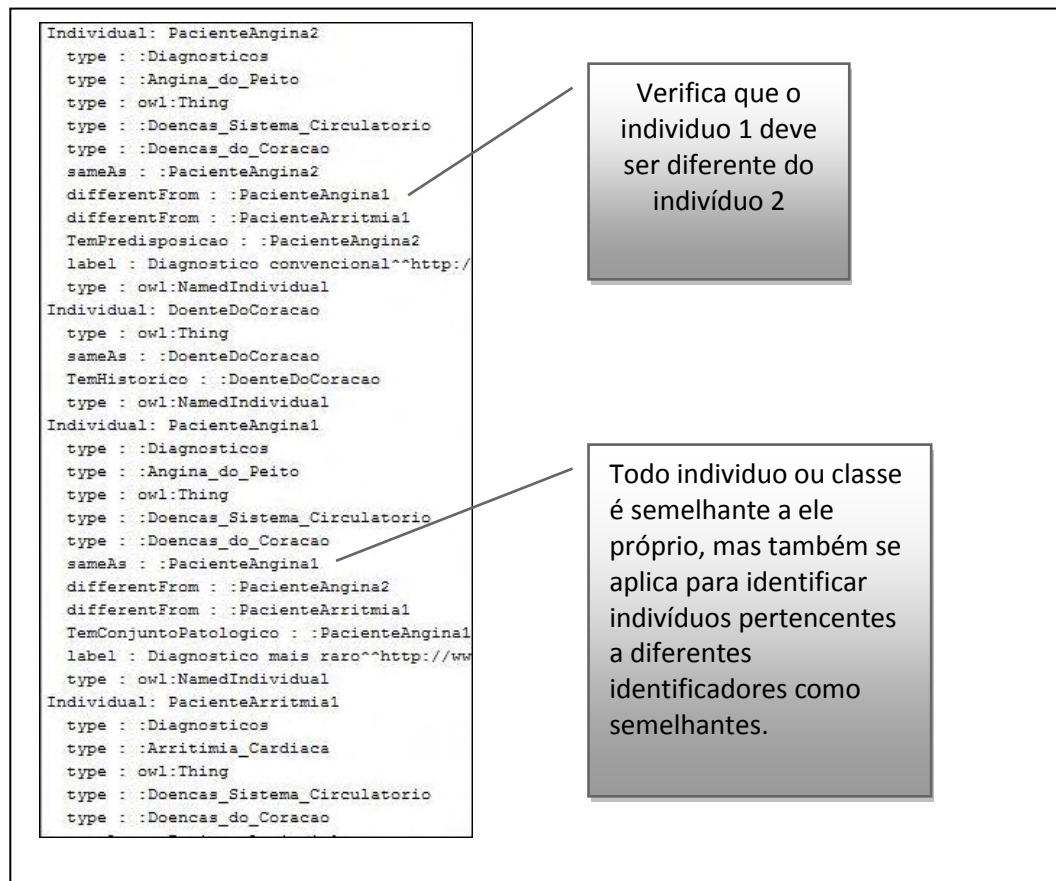


Figura 28 - Último nível de inferência.

Pelo tamanho das propriedades inferidas não foi possível agrupar todas na figura. É importante comentar que apesar de existirem mais informações no modo de inferência acima, a capacidade pode ser bem maior bastando apenas existirem relacionamentos, restrições e outros recursos que permitam o motor de inferência buscar novos relacionamentos.

4.4.2 Incorporando Ontologias na Seleção de Atributos

Os métodos convencionais para seleção de atributos tratam o conjunto existente sem nenhuma especificação prévia do domínio que está sendo tratado. Em uma metodologia convencional, o especialista antes de iniciar o processo específico de mineração de dados deve selecionar as variáveis que serão trabalhadas pelos algoritmos de descoberta de conhecimento. Esse processo é bastante trabalhoso e muitas vezes repetitivo, precisando-se efetuar diversas análises sobre o conjunto de dados e assim conseguir um subconjunto de atributos que represente de forma satisfatória o domínio específico. Usar a ontologia como uma forma de armazenamento prévio de informações sobre o domínio pode ser de grande utilidade, trazendo informações que levariam tempo para serem coletadas (NOY, 2001).

Para usar as ontologias no armazenamento desse conhecimento prévio e permitir uma estruturação genérica do domínio tratado, podem-se seguir determinados passos:

1 – Deve-se buscar ontologias de domínio já criadas e fazer reuso das informações contidas nas mesmas. O especialista pode adicionar mais conceitos ou remover outros que não se enquadram na análise.

2 – Se essas ontologias não existirem ou não trouxerem informações satisfatórias o especialista deve construir a mesma.

3 – A construção da ontologia deve ser feita levando em conta o nível de informação necessário para análise, por exemplo, uma ontologia especificada na linguagem OWL Full seria desnecessária para avaliar e armazenar atributos em determinadas categorias, pois traz propriedades, restrições e outras categorias que podem não ser utilizadas.

Para se usar uma ontologia na seleção de atributos deve-se pensar qual a análise que será feita nos atributos com essa ontologia, ou seja, como o especialista deve interagir com as informações disponíveis e usá-las de modo a permitir visualizar a importância de determinados atributos em seus devidos contextos. Ferramentas como Protégé (PROTÉGÉ, 2006), OntoEdit (NOY, 2001), ajudam a visualizar as informações contidas na ontologia e permitem análises visuais nas mesmas.

Para incorporar ontologias na mineração de dados, levando em conta a metodologia explicada na Seção 2.4, esse processo pode ser feito das ontologias para a mineração de dados ou de forma inversa da mineração de dados para as ontologias. Neste trabalho, é necessário o uso da primeira metodologia, podendo assim incorporar conhecimento sobre o domínio antes mesmo do início da mineração de dados.

Para usar a expressividade da ontologia foram feitos os seguintes estudos:

1. O relacionamento entre a ontologia e os atributos deve ser de tal forma que possibilite extrair algum conhecimento sobre a qualidade dos atributos.
2. A análise deve contar com o auxílio de uma ferramenta de inferência para efetuar uma classificação na taxonomia original da ontologia.
3. O auxílio em alguma fase da mineração de dados deve ser de tal forma que diminua o esforço do especialista ao escolher atributos para o mesmo domínio, principalmente quando o trabalho é feito diversas vezes de forma iterativa.

Para o primeiro caso, como a abordagem feita neste trabalho efetua um mapeamento entre ontologia e atributos capturados, é de extrema importância que a ontologia possa ser

simples e ao mesmo tempo expressar o domínio tratado. Para isso, deve-se transformar ou readaptar a ontologia de domínio para o propósito de categorização dos atributos, e estes por fim podem ser visualizados de uma forma bem mais intuitiva e com a taxonomia semântica existente na ontologia.

No segundo estudo, a ferramenta de inferência permite capturar a taxonomia convencional da ontologia e refazê-la de modo a evidenciar as propriedades, permitindo que classes que pertençam a determinados grupos possam ser agrupadas de forma conjunta.

Por último, a diminuição do esforço do especialista se dá principalmente com a categorização dos atributos que estão sendo avaliados. Esse processo permite:

- Organizar os atributos por categorias semânticas;
- Permitir a captura de vários atributos relacionados semanticamente com apenas um clique;
- Guardar essa categorização para uma posterior mineração de dados e permitir a diminuição do esforço do especialista para escolher atributos de forma manual.
- Adaptar o processo de categorização para uma possível redução de dimensionalidade dos dados, podendo efetuar generalizações antes muito difíceis de serem percebidas.

Categorizar significa separar em categorias, e essas categorias quando pertencentes a uma ontologia estão relacionadas semanticamente, sejam através do conceito de subclasse, classe equivalente, propriedades que relacionam e restringem aquela determinada classe. Quando o especialista mapeia seus atributos a determinados grupos de classes, ele automaticamente pode ter uma visão mais apurada do domínio e dos dados envolvidos no problema. Em sites de busca da web específicos para ontologias, citados no próximo capítulo, podem-se encontrar diversas ontologias de domínio prontas e fazer o aproveitamento das informações contidas na mesma. Na figura 13 pode ser visualizada a taxonomia da parte de uma ontologia médica para o processo de categorização:

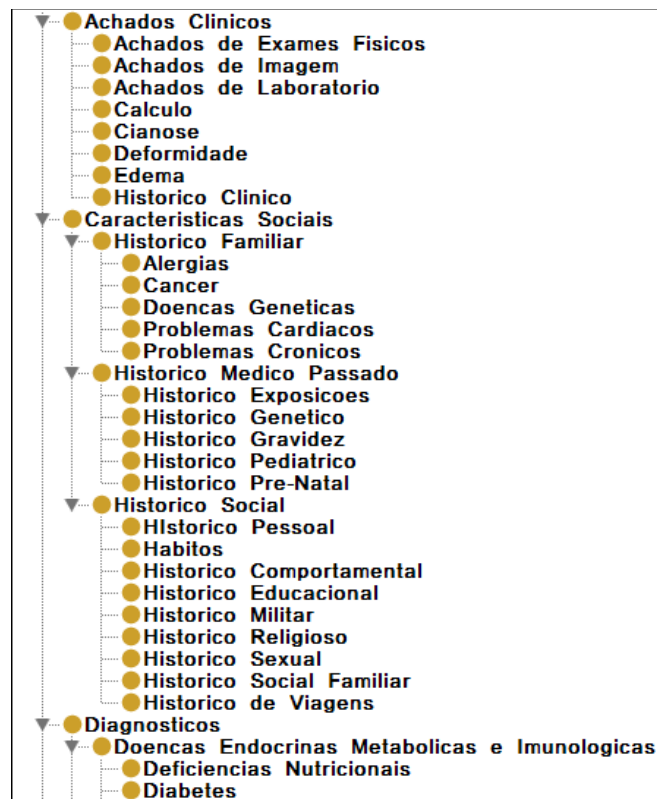


Figura 29 – Parte da ontologia remodelada para efetuar o mapeamento dos dados

Essa ontologia foi remodelada a partir de repositórios já existentes na web e locais especializados em termos médicos e que serão comentados no próximo capítulo.

Tendo essa visualização da ontologia o especialista precisa ter os atributos que precisam ser mapeados e categorizados. Para capturar os atributos de um banco de dados que possui um determinado esquema, suas várias tabelas devem ser consultadas e os dados agrupados em uma única estrutura dimensional (o que é um fato comum na mineração de dados). Por exemplo, entre os vários atributos disponíveis em uma tabela que continha informações sobre fumantes, dentre elas existiam vários atributos (colunas) indicando o tipo de cigarro fumado. E esses atributos são mapeados para o conceito fumantes, que já pertence à categoria grupo de pessoas de risco. Em outra tabela existem atributos que indicam o consumo de álcool, vinho etc. Esses atributos são mapeados dentro do conceito determinado na ontologia. Finalizado a categorização, o usuário poderá visualizar o resultado do mapeamento:

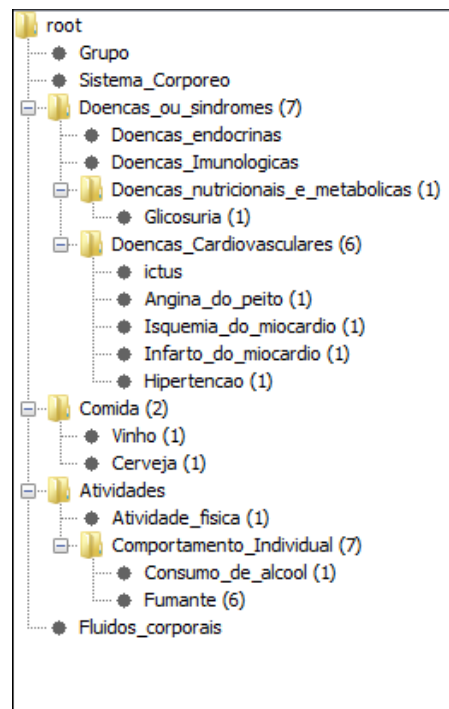


Figura 30 – Parte da ontologia com Atributos Categorizados.

Na figura 14 é possível visualizar os atributos categorizados na ferramenta SAAO. Dentro da categoria de doenças cardiovasculares, por exemplo, foram mapeados 6 atributos do banco de dados, onde um deles está relacionado à *Angina do peito*, outro a *Isquemia do Miocárdio* e assim por diante. Muitos outros conceitos foram mapeados para as correspondentes classes da ontologia, permitindo que o especialista possa selecionar determinado(s) grupo(s) de atributos com relativo compartilhamento semântico. Esse foi apenas um exemplo da utilização da categorização proposta neste trabalho e será especificado melhor no capítulo seguinte e no capítulo contendo experimentações e cenários de utilização.

4.4.3 Categorizando Atributos

As análises semânticas nos atributos com o auxílio de uma ontologia podem ser feitas de diversas formas e cada uma delas ajuda o especialista no momento da escolha, modificação ou eliminação desses atributos. A primeira e mais importante análise se trata do próprio mapeamento entre os atributos e as classes da ontologia, pois a visualização desses atributos categorizados permite fazer escolhas de uma forma mais clara no domínio que está sendo tratado. A categorização dos atributos com a ontologia também ajuda a identificar

atributos relacionados semanticamente (principalmente ao nível de nomenclatura e taxonomia).

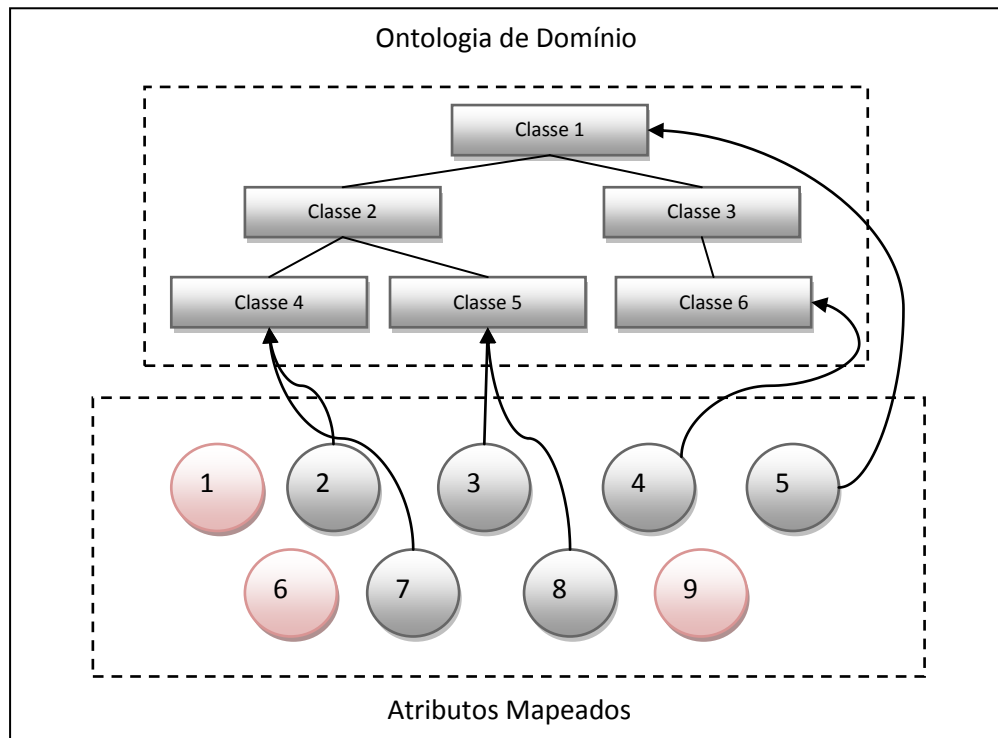


Figura 31 - Mapeamento Atributo – Classe.

Na figura acima temos atributos mapeados com as respectivas classes. Primeiramente, cada classe contém um determinado número de atributos mapeados, e esses atributos também pertencerão ao pai e assim por diante. Por exemplo, a classe quatro apresenta dois atributos mapeados: atributo 2 e 7, ou seja, $[classe\ 4 \leftarrow (2,7)]$. Então, se $[classe\ 5 \leftarrow (3,8)]$, pode-se concluir que a classe pai das duas terá um domínio capaz de conter todos os atributos, logo 2: $[classe\ 2 \leftarrow (2, 3, 7, 8)]$. A categoria representada pela classe 2 pode substituir as classes abaixo de sua hierarquia dependendo do contexto utilizado. O especialista terá a possibilidade de visualizar essa relação que antes poderia ser difícil de perceber nas tabelas do banco de dados. Existem alguns atributos que estão sendo representados em vermelho e não foram mapeados, pois são atributos que não colaboram ao processo de descoberta de conhecimento. Esses atributos podem representar chaves auto-incrementais do banco de dados ou até mesmo atributos inconsistentes e com falhas de modelagem. Então, além de categorizar os atributos ainda se elimina do processo os atributos errôneos. A categorização diminui bastante o esforço na fase de seleção de atributos, principalmente pela característica de poder selecionar um grupo de atributos com apenas um comando. Pela figura

abaixo é possível perceber como seria complicado analisar o esquema do banco de dados (a figura tem o objetivo de mostrar as diversas tabelas interligadas formando o esquema e não fazer leitura de nomes existentes) e selecionar os atributos importantes em um determinado contexto.

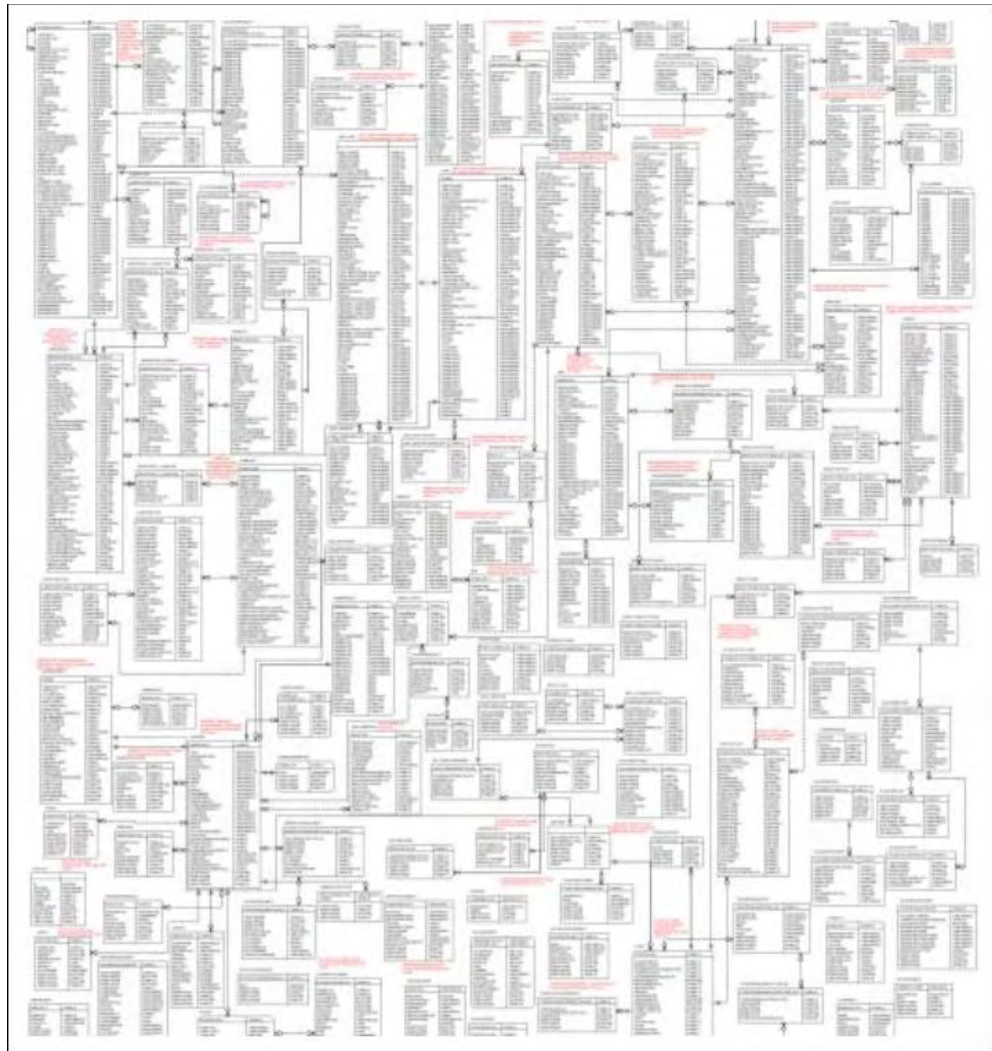


Figura 32 - Extenso Esquema de um Banco de Dados

No protótipo SAAO o processo de mapeamento funciona selecionando cada atributo e mapeando com a sua correspondente classe na ontologia. Cada atributo mapeado terá uma ligação direta com a classe da ontologia e o usuário poderá usufruir dos relacionamentos semânticos da classe para avaliar o enquadramento do atributo na classe selecionada. Esse tipo de avaliação pode ser feita de forma mais apurada e será mais bem explicada no tópico sobre redução de dimensionalidade. Para realizar o mapeamento o usuário conta com ajuda de

um módulo de distância de edição, mostrando dentre as classes da ontologia, quais as que apresentam maior semelhança sintática com o atributo selecionado. Veja o exemplo a seguir:

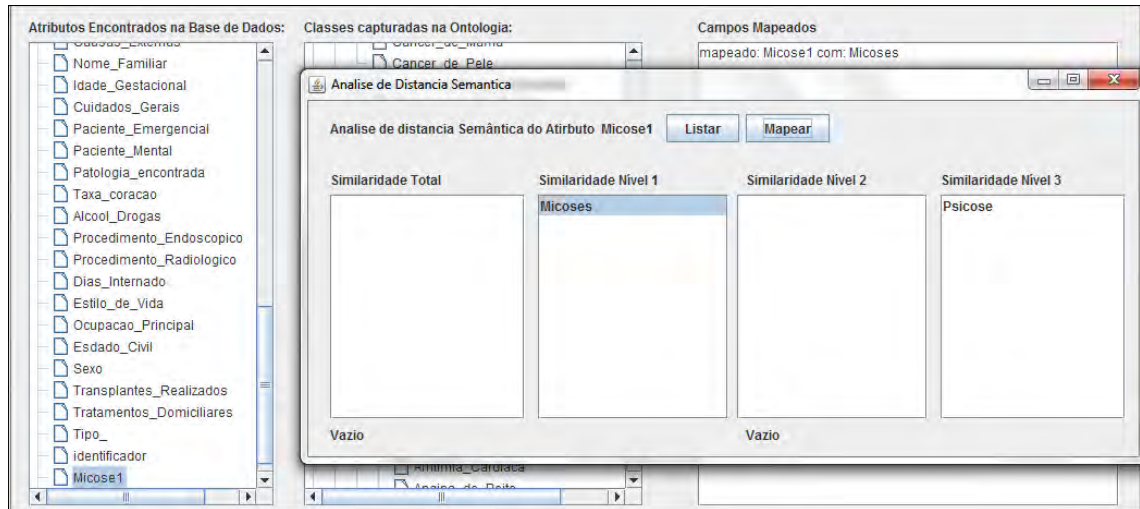


Figura 33 - Similaridade Sintática.

Levando em conta os resultados acima, o atributo do banco de dados *Micose1* foi verificado em três níveis de similaridade. O primeiro ocorre quando a escrita é semelhante e os outros são destacados dependendo das quantidades de transformações necessárias para se transformar um nome em outro. O resultado na similaridade nível 1 (*Micoses*) é considerado satisfatório e pode ser mapeado, porém no nível 3 foi encontrada a classe *Psicose* na ontologia e logicamente não é a mais indicada para mapear o atributo *Micose1*. Então, esse módulo ajuda bastante no momento do mapeamento, mas deve ser utilizado com cuidado para não mapear atributos em classes que não apresentam nenhuma relação semântica com o atributo do banco de dados.

Uma informação importante é que cada classe pode conter vários atributos, mas o atributo só pode ser mapeado para uma única classe. Esse tipo de restrição teve de ser feita para evitar que atributos selecionados em um determinado conjunto de classes sejam selecionados novamente em outro conjunto e assim gerar inconsistência na consulta SQL. Após o mapeamento, o usuário é capaz de selecionar um grupo de atributos relacionados semanticamente apenas com um clique e todos são adicionados na saída que será gerada para a ferramenta de mineração Weka.

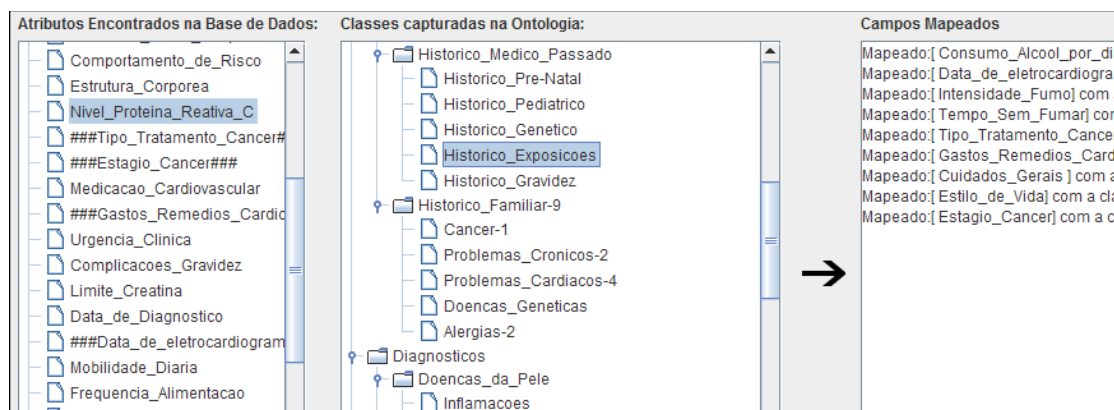


Figura 34 - Atributos categorizados.

Na figura acima, por exemplo, a classe *Histórico_Familiar* apresenta nove atributos mapeados, onde quatro deles estão na sub-classe *Problemas_Cardiacos*, dois em *Problemas_Cronicos* e assim por diante. Se o usuário tivesse interesse em atributos relacionados a todo o histórico familiar ele simplesmente selecionaria a classe e o programa automaticamente adicionaria os nove atributos correspondentes. Porém, se estivesse interessando apenas nos problemas crônicos ele simplesmente selecionava a classe *Problemas_Cronicos* e os dois atributos correspondentes seriam adicionados na saída de consulta.

4.4.4 Visualizando Redundâncias

Atributos redundantes em bancos de dados são aqueles que podem ser derivados a partir da execução de procedimentos de busca de dados e/ou cálculos sobre o banco de dados (HEUSER, 2009). Na mineração de dados considera-se que os atributos são redundantes entre si, parcial ou completamente, quando seus valores estão correlacionados. O surgimento de atributos redundantes na mineração se deve principalmente a junção de base de dados diferentes, pois nesse tipo de caso não existe uma normalização dos dados na disponibilização dos atributos para mineração. Neste trabalho, a redundância é determinada quando os atributos são altamente correlacionados entre si. Verificando a classe onde esse atributo foi mapeado e todos os outros atributos mapeados em classes equivalentes ou com instâncias semelhantes à existente na classe do atributo é possível comparar o grau de semelhança semântico entre eles. Em KDD a verificação de redundância pode ser bastante complicada. Em uma análise subjetiva de um usuário (especialista, projetista etc.) um atributo pode parecer redundante, porém esse mesmo atributo não é classificado como redundante por outro usuário em sua avaliação. Outro fator que pode influenciar é a finalidade da descoberta de

conhecimento, pois em alguns casos atributos possivelmente redundantes em determinado nível de informação podem não ser em outro. Alguns atributos são redundantes devido ao próprio projeto de banco de dados, um exemplo seria ter em uma mesma tabela atributos data de nascimento e idade, onde nesse caso idade pode ser calculada facilmente a partir da data de nascimento.

Na busca por redundâncias, o usuário após efetuar a categorização dos atributos, pode ser capaz de identificar atributos relacionados semanticamente e esse é o primeiro passo para que as informações contidas na ontologia possam auxiliar na visualização de redundâncias. Outro passo importante é a ontologia conter informações que permitam identificar classes equivalentes ou instancias semelhantes. No primeiro caso, quando as classes são equivalentes elas contém características em comum e qualquer instância pertencente a uma delas pertence à outra, e no segundo caso, instâncias classificadas como iguais identificam conceitos semelhantes. Finalmente, todos os atributos altamente correlacionados entre si são mostrados para o usuário decidir sua remoção.

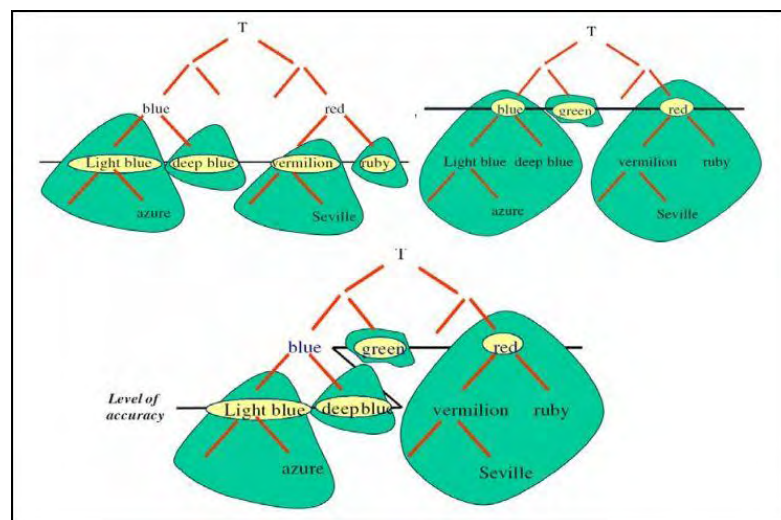


Figura 35 - Hierarquia usando árvores com atributos redundantes.

A figura acima é apenas um exemplo genérico mostrando a representação de diversas cores em uma árvore. Dependendo do nível de informação necessária na mineração de dados, alguns dos atributos podem ser considerados redundantes. Por exemplo, o vermelho (*red*) apresenta as seguintes sub-classes (*vermilion*, *ruby*, *Seville*) que poderão ser eliminadas pois representam a mesma cor mais genérica: o vermelho.

Como já comentado em tópicos anteriores, a ontologia usada para a maioria dos exemplos desse trabalho foi montada com auxílio de outras ontologias prontas, de domínio

médicos, já existentes na *Web* para o propósito de consulta. Então, a quantidade de classes equivalentes é suficiente para determinar redundâncias em determinados atributos, porém para uma maior listagem de equivalências esse deve ser um processo incremental. A ontologia deve sempre evoluir, melhorar seus termos e relacionamentos, possibilitando que um conhecimento mais rico sobre um determinado domínio seja agrupado. Alguns projetos, como o CYC, tentam agrupar todo conhecimento disponível em uma única ontologia, já outros expressam a necessidade de descentralizar o conhecimento e torná-lo disponível em diversas ontologias pela *Web*. Portanto, existem diversas vertentes sobre a montagem de domínios em ontologias e nessa pesquisa a amostragem de termos na ontologia foi suficiente para indagar alguns questionamentos e montar cenários explicativos no capítulo sobre cenários de utilização.

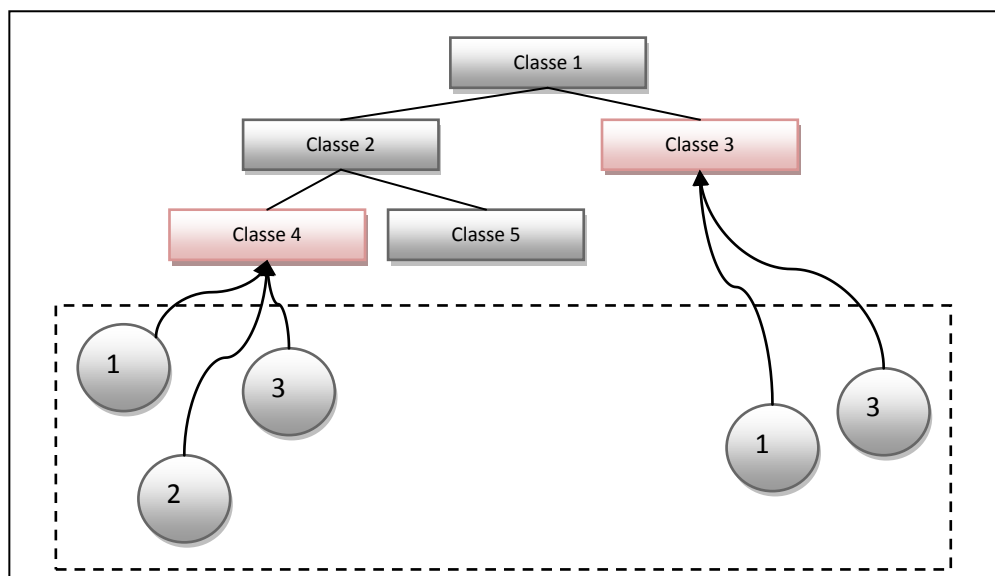


Figura 36 - Atributos candidatos a redundância.

No exemplo mostrado na figura acima a classe 4 é equivalente a classe 3, portanto todos os atributos mapeados nas duas classes podem ser correlacionados entre si. O usuário deverá analisar os resultados e avaliar se alguns desses atributos não serão necessários devido a sua similaridade com outro já mapeado em sua classe equivalente. Na ferramenta temos a seguinte visualização para o usuário:

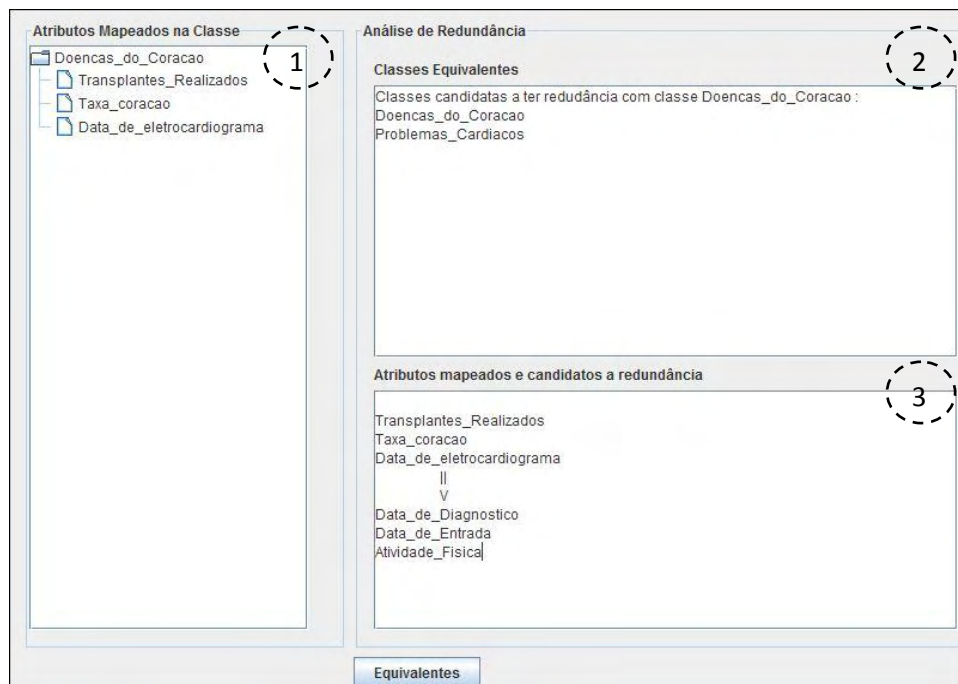


Figura 37 - Visualização de Possíveis Redundâncias.

Quando o usuário faz o mapeamento dos atributos e acha necessário verificar redundâncias, ao clicar no botão correspondente obterá a tela mostrada na figura 33. Dependendo da classe que estiver selecionada na aba correspondente da ontologia, todos os atributos mapeados para essa classe serão mostrados na área 1. Todas as classes que forem consideradas equivalentes às selecionadas serão mostradas na área 2, inclusive a própria classe. Finalmente na área 3 os atributos candidatos a redundância são listados. É importante ressaltar que os atributos mostrados nessa área podem estar correlacionados um com o outro mais nem por isso serão redundantes. Então, é fundamental a análise visual do usuário para determinar a real correlação semântica entre esses atributos para o propósito da mineração desejado.

4.4.5 Reduzindo Dimensionalidade dos Dados

A redução de dimensionalidade permite diminuir a quantidade de atributos em um conjunto, pois atua substituindo rótulos (nesse trabalho constituem as classes da ontologia) de nível inferior por rótulos de nível superior. Nas Seções 2.2.3.1 e 2.2.3.2 tratadas anteriormente foram explicadas dois tipos de reduções, vertical e horizontal. Ambas as reduções se enquadram como reduções de dimensionalidade e de forma geral são feitas manualmente pelo

usuário ou especialista na etapa de pré-processamento em KDD. Dependendo da quantidade de atributos pode ser um processo bastante trabalhoso e exigir diversas visualizações sobre os dados para ser concluída, portanto a existência de uma ferramenta para auxiliar nessa fase é de grande utilidade.

Por eliminar linhas (tuplas), a redução horizontal não chega a ser tratada nesse trabalho. Então, a eliminação de atributos faz uma redução vertical dos dados e é bem mais delicada que a remoção de tuplas, pois pode comprometer uma quantidade bem maior de informações. Para auxiliar nesse processo de redução dos dados a idéia central é utilizar os relacionamentos taxonômicos da ontologia e as propriedades individuais de cada classe. Por exemplo, em um processo de mineração de dados o especialista está selecionando atributos referentes às infecções respiratórias de determinado paciente e no banco de dados existem os atributos: tosse crônica, bronquite crônica, influenza, sinusite crônica, etc. Todos informam nominalmente que o paciente teve ou não o tipo de infecção. A depender da necessidade da mineração ou até mesmo do algoritmo que será executado não se faz necessário a inclusão de atributos específicos como tosse crônica ou sinusite crônica, apenas um atributo que represente esse tipo de infecção. Mas, o usuário teria previamente que saber que essas são infecções respiratórias ou mesmo que fazem parte do grupo de doenças do sistema respiratório. Então, com a ontologia já mapeada, a verificação desse tipo de generalização é bastante simples. Supondo ainda que ele esteja analisando a incidência de determinadas doenças respiratórias em alguns estados brasileiros, então atributos mais específicos vão apenas aumentar o espaço de dados a serem analisados pelos algoritmos de mineração de dados. Esses atributos podem ser removidos sem afetar a qualidade das descobertas que serão efetuadas pela ferramenta de mineração.

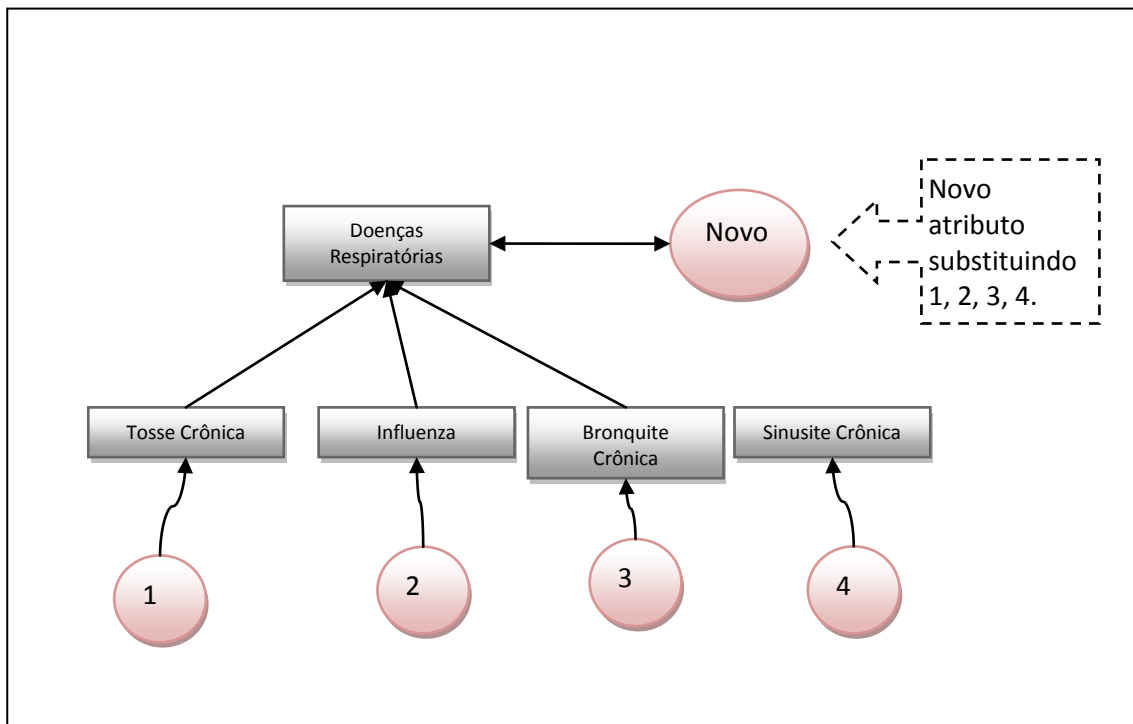


Figura 38 – Exemplo Redução de dimensionalidade.

A figura acima demonstra exatamente o que foi explicado anteriormente, ou seja, os atributos mapeados nas subclasses serão transformados em um único atributo que representa a classe “doenças respiratórias”. Porém, como no mapeamento é possível incluir mais de um atributo por classe, esse processo pode ser especificado ainda mais. Se em “Tosse Crônica” existissem mais de um atributo mapeado, então essa classe já se enquadraria como uma classe redutora se fosse necessária na avaliação do usuário. O usuário poderia fazer o mesmo processo que fez para doenças respiratórias e criar um único atributo representando a classe “Tosse Crônica”.

Com esse tipo de redução a quantidade de atributos pode diminuir drasticamente e permitir a descoberta de padrões mais interessantes para a avaliação em questão.

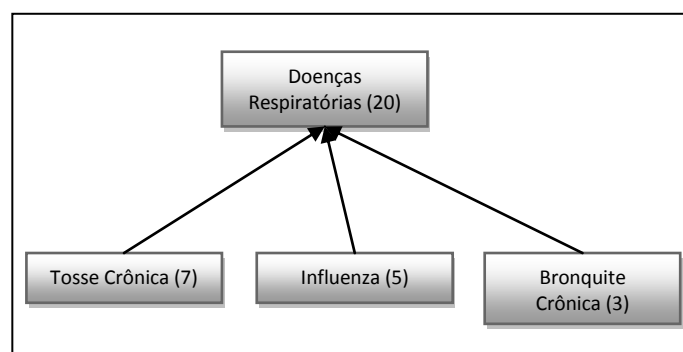


Figura 39 - Reduzindo Mapeamento.

Pela configuração do mapeamento da figura 35, a classe “Doenças Respiratórias” apresenta 20 atributos, onde 7 pertencem à “Tosse Crônica”, 5 a “Influenza”, 3 a “Bronquite Crônica” e os 5 restantes estão mapeados na própria classe “Doenças Respiratórias”. A redução nesse caso permitirá a eliminação de 15 atributos, pois todos os atributos pertencentes às subclasses de doenças respiratórias serão representados em um único atributo representado na classe doenças respiratórias que finalmente apresentará apenas 6 atributos finais. Esse tipo de redução pode ser efetuada em qualquer domínio, não apenas para o domínio médico tratado nesse trabalho.

Na ferramenta SAAO, para efetuar a redução de dimensionalidade basta selecionar a classe para verificação e depois clicar no botão <semântica>. Os resultados obtidos vão depender da localização da classe dentro da ontologia, da quantidade de superclasses que ela tiver, da quantidade de instancias da superclasse, dos relacionamentos entre as instancias e outras classes etc. O usuário vai obter todas as classes irmãs da classe selecionada e as superclasses serão candidatas a generalizar os atributos. No lado esquerdo ainda é possível visualizar informações sobre a classe selecionada e que podem ser úteis para tomar a decisão se realmente é necessário generalizar os atributos existentes na mesma.

Para a classe Infarto_do_Miocardio foram efetuadas as seguintes recomendações:	
Redução de Dimensionalidade Os atributos mapeados nas classes: Nothing Arritmia_Cardiaca Angina_do_Peito Infarto_do_Miocardio Aterosclerose Quantidade de atributos mapeados nas classes acima: 3 podem ser generalizados por um atributo que represente: 1 - Diagnosticos 2 - Thing 3 - Doencas_Sistema_Circulatorio 4 - Doencas_do_Coracao	Relacionamento dos Indivíduos Descrição e propriedades da classe selecionada: Dor ou desconforto transitório localizado na região anterior do tórax, percebido como uma sensação de pressão, aperto ou queimação . Relacionamentos: doença arterial coronariana doenças do músculo cardíaco

Figura 40 - Módulo para auxílio na redução de dimensionalidade.

Na avaliação acima o usuário selecionou a classe *Angina_do_Peito* e a ferramenta retornou que existem as classes: *Aterosclerose*, *Infarto_do_Miocardio*, *Arritmia_Cardiaca*, como irmãs da classe *Angina_do_Peito*. No total existem 3 atributos mapeados entre essas classes e a sugestão é que o usuário generalize todos esses atributos em um único atributo podendo se chamar *doenças_do_coração* ou *Doenças_do_Sistema_Circulatório*. Ainda, para auxiliar na decisão são disponibilizadas algumas descrições da classe *Angina_do_Miocardio* e algumas propriedades importantes para essa classe encontradas na ontologia.

4.4.6 Método de Salvamento e Utilização dos Resultados do SAAO

Um fato importante de ser comentado, explicado no Seção 2.2.3, é que os atributos do problema serão selecionados por junção orientada, ou seja, atributos com maior potencialidade para influir no processo de KDD serão escolhidos. Diferindo assim da junção não orientada, que utiliza todos os atributos e os envia sem nenhuma modificação para o algoritmo de mineração que deverá de alguma forma eliminar alguns desses atributos (alguns algoritmos possuem características que eliminam atributos de forma nativa). Esse é exatamente o papel do SAAO, auxiliar na diminuição do conjunto de atributos iniciais, provendo as funcionalidades de categorização, avaliação de redundância e redução de dimensionalidade. Porém, o que acontece depois que o especialista seleciona seus atributos e usa a ferramenta de mineração de dados? Logicamente, esse mapeamento deve ficar salvo para que possa ser utilizado novamente em outras minerações de dados nesse domínio específico. Então, todo o mapeamento é salvo em um arquivo no formato XML e pode ser lido novamente em qualquer momento. Foi escolhido esse formato, pois além de ser a linguagem de base de qualquer ontologia é o padrão para compartilhamento de informações em programas e na Web, permitindo que o mapeamento possa ser utilizado por outros aplicativos futuramente. O arquivo com os dados salvos é descrito da seguinte forma:

```

<mapeamento>
  <classe>
    <nome>Classe1</nome>
    <atributo1>AtributoMapeado1</atributo1>
    <atributo2>AtributoMapeado2</atributo2>
    <atributo3>AtributoMapeado3</atributo3>
    ...
  </classe>
  <classe>
    <nome>Classe2</nome>
    <atributo1>AtributoMapeado1</atributo1>
    <atributo2>AtributoMapeado2</atributo2>
    <atributo3>AtributoMapeado3</atributo3>
    ...
  </classe>
  ...
</mapeamento>

```

Figura 41 - Salvando o Mapeamento.

Outro aspecto importante e que não foi comentado nos tópicos acima é como os atributos selecionados serão mostrados para o usuário e como ele pode os utilizar na ferramenta de mineração. Depois que todos os atributos estão mapeados e já foram verificadas as redundâncias e reduções de dimensionalidade o usuário deve clicar no botão *WekaQuery* e vai obter um simples *select* com todos os atributos escolhidos.

Atributos selecionados para a ferramenta de Mineração de dados (Weka padrão):

```

SELECT ("Nome", "Intensidade_Fumo", "Tempo_Sem_Fumar", "Atividade_Fisica",
"Meio_Transporte", "Duracao", "Idade", "Data_Entrada", "Data_Saida",
"Consumo_Alcool_por_dia", "Consumo_Alcool_Frequencia", "Comportamento_de_Risco",
"Estrutura_Corporea", "Nivel_Proteina_Reativa_C", "Tipo_Tratamento_Cancer",
"Estagio_Cancer") FROM "Doencas_Desastres"

```

Clipboard Descartar

Figura 42 - Atributos selecionados.

Com os resultados da seleção prontos o usuário só precisa adicionar a consulta na ferramenta de mineração de dados, por exemplo, no Weka ao abrir o Explorer, seleciona a opção “Open DB” e insere a consulta obtida acima.

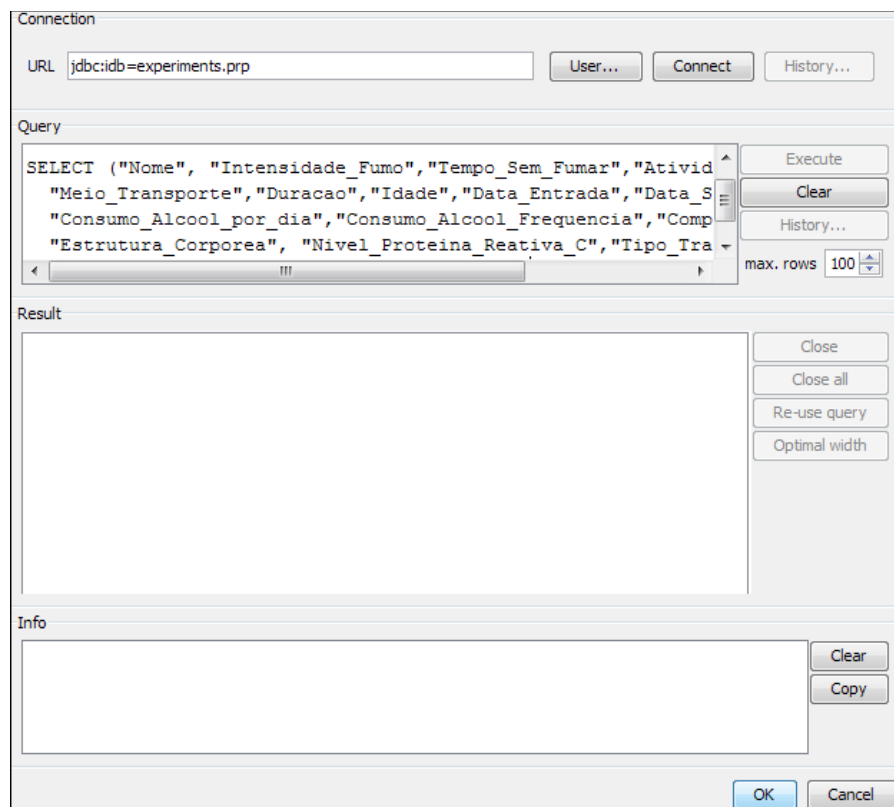


Figura 43 - Captura de atributos na ferramenta WEKA.

4.5 Considerações Finais

Neste trabalho foi apresentada a proposta de um protótipo para auxiliar o especialista em KDD a melhorar a seleção de atributos na fase de pré processamento. O foco principal está em usar ontologias para prover semântica que possa ser utilizada durante o processo de seleção e com isso verificar atributos que não terão grande importância para a etapa de Mineração de Dados. Esse processo é feito mapeando os atributos do banco de dados para as classes da ontologia. Com esse mapeamento pronto, o especialista consegue visualizar todos os atributos em categorias semânticas e selecionar grupos de atributos com apenas um clique. Outro fato importante é a possibilidade de analisar redundâncias e reduzir a dimensionalidade dos dados, pois a ontologia especifica algumas informações que podem ser úteis na definição desses processos. Com a diminuição da quantidade de atributos, diminui-se a dimensionalidade total das amostras e aperfeiçoa as etapas de KDD posteriores a seleção de

atributos. No próximo capítulo serão estudados alguns estudos de caso com o uso do protótipo.

5 CENÁRIOS DE APLICAÇÃO DO PROTÓTIPO SAAO

5.1 Considerações Iniciais

Com o objetivo de mostrar alguns exemplos de utilização do protótipo SAAO, nesse capítulo serão abordados alguns cenários de utilização. Cada cenário terá uma pergunta base que será utilizada pelo usuário antes de efetuar a mineração de dados, ou seja, “eu estou fazendo a mineração de dados com que objetivo?”. Feito isso é importante também destacar em que tarefa de mineração é viável seu uso, portanto tarefas como descobertas de associações, classificação dos dados podem ser avaliadas nesses cenários. Embora as características gerais de um conjunto de dados possam promover uma idéia de que o subconjunto de algoritmos poderiam produzir melhores resultados, na há análise matemática capaz de determinar que um algoritmo de aprendizado construirá melhores modelos que outros (Dietterich, 1989) ,(Langley, 1988). O mesmo ocorre quando se trata de algoritmos de seleção de atributos. Desse modo, avaliações experimentais constituem um importante instrumento de estimativa de quais algoritmos seriam mais apropriados em relação ao conjunto de dados trabalhado. Ou seja, para analisar e comparar resultados dos algoritmos de seleção já existentes e o SAAO deve-se optar por uma abordagem experimental, com dados para se visualizar as melhorias em cada processo. Mais a frente o processo de comparação será mais bem explicado, porém convém mencionar que irá tratar tanto aspectos quantitativos como qualitativos. Em aspectos quantitativos se trabalha com quantidade de atributos selecionados e os aspectos qualitativos representam o ganho semântico e significativo da inclusão ou remoção de determinados atributos.

O objetivo principal desse trabalho é auxiliar o especialista em KDD na seleção de atributos, porém não eliminando nenhuma das abordagens ou algoritmos que também fazem a seleção de atributos. Ou seja, ao executar a seleção de atributos proposta neste trabalho ainda é possível executar algoritmos para seleção existentes no Weka, ajudando relativamente a diminuir a quantidade de atributos para avaliação.

Como descrito no capítulo anterior, o processo de seleção no SAAO utiliza uma ontologia como ponto central para efetuar mapeamento entre os atributos e as classes dessa ontologia. Essa ontologia não pode ser muito genérica, nem mesmo especializar muitos os termos. Se as classes forem muito genéricas não existirá um verdadeiro suporte semântico

para seleção, pois não será possível analisar os atributos em contextos mais específicos e que poderiam trazer alguns benefícios. Em contraponto, se a ontologia for muito específica não dará suporte para o mapeamento de diversos atributos que possam existir no banco de dados utilizado. Então, a correta especificação das áreas de determinado domínio é importante, visto que a ontologia dará um melhor suporte nesse conjunto de dados. O processo de construção ou melhoramento de ontologias existentes deve levar em conta esses critérios, elaborando uma nova ontologia se for partir para um novo tipo de mineração ou fixando a mesma ontologia e reaproveitando o mapeamento já feito anteriormente, se for considerar o mesmo domínio já trabalhado.

5.2 Coleta de Dados

Para viabilizar a seleção de atributos, antes disso, logicamente, deve-se ter os dados que vão ser avaliados. A informatização na medicina nos dias atuais está bem avançada, porém esse processo ainda está em fase de implantação em muitos hospitais, clínicas e centros universitários. Advindo desse fator vem a primeira dificuldade: conseguir um banco de dados no domínio médico para ser usado no SAAO. Existem vários modelos que podem ser considerados compatíveis com a ontologia criada nesse trabalho para mapeamento, como por exemplo: prontuários eletrônicos, pesquisas no domínio médico, históricos de pacientes etc. Por ainda estar em estágio de avaliação e prototipagem foi preferido elaborar um banco de dados para efetuar os estudos de caso, dividindo a experimentação em alguns cenários.

Outra forma de efetuar a construção de uma base de dados poderia ser a partir de prontuários médicos, porém a maioria dos locais pesquisados não informatizou esse procedimento. Além disso, o tratamento dos dados e transposição para o formato eletrônico poderia ser muito custoso e precisar de muitos outros métodos para realizar limpeza e padronização sobre os dados coletados.

Os cenários descritos utilizam atributos médicos criados a partir de questionários relativos à medicina preventiva. Esse tipo de questionário geralmente envolve aspectos importantes a determinado tema médico e o questionário escolhido pertence a centro de tratamento de câncer da Universidade do Texas (MDAnderson Cancer Center⁸).

⁸ Encontrado no endereço eletrônico: <https://my.mdanderson.org/>

THE UNIVERSITY OF MICHIGAN
MD ANDERSON CANCER CENTER

Patient History Database

Institutional _____ **Patient** _____ **Print Date** _____ **MDA #** _____
ACCT# _____ **DOB** _____ **FC** _____ **SEX** _____ **Location** _____

PLEASE fill in WHITE AREAS, FRONT AND BACK OF ALL PAGES, BEFORE YOUR APPOINTMENT. Your answers will help the staff plan and provide your care, as well as help us with our research to better understand the risk factors for cancer. Leave blank any parts you are unsure of, or do not wish to answer. We will review the form with you. Any information we gather will be kept confidential. PRINT AND USE INK. Thank you.

PERSON COMPLETING THIS FORM: _____
☐ PATIENT ☐ OTHER (Indicate Relationship in Patient) _____
PRIMARY LANGUAGE: ☐ English ☐ Spanish ☐ Other _____ ☐ Used Certified Translator: _____

CURRENT MEDICAL HISTORY:
 What is your medical reason for coming to M. D. Anderson? (Chief Complaint) _____
 Please give the history of your current problem: (when it started, symptoms, treatment) _____

 Are you ALLERGIC to anything? ☐ Yes ☐ No
 List all ALLERGIES and describe your reaction: _____

PAST MEDICAL HISTORY: Please check ALL previous illnesses or conditions below:
☐ Bleeding problems ☐ High blood pressure ☐ Psychological/Psychiatric problems
☐ Circulation problems ☐ HIV/AIDS ☐ Seizure
☐ Diabetes or sugar in urine ☐ Kidney/urine problems ☐ Stroke
☐ Frequent infections ☐ Liver problems ☐ Thyroid problems
☐ Heart problems ☐ Lung problems ☐ Tuberculosis ☐ Other _____
 Please provide more information below for any of the conditions or illnesses you checked above: _____

Figura 44 - Demonstrativo do Modelo de Pesquisa com Históricos dos Pacientes

Para adaptar o questionário que está em formato digital no padrão PDF e extrair os atributos importantes para avaliação, foram aplicadas as seguintes estratégias:

- Partindo de determinadas perguntas existentes no questionário, com o auxílio de um especialista de domínio e ferramentas existentes na Web como o *National health data dictionary*⁹ e *UMLS Knowledge Source Server*¹⁰ foi construído um dicionário com os atributos considerados mais importantes no questionário estudado.
- Etapa 2: Com o dicionário construído essa etapa consiste em fazer correspondências entre os dados do questionário e os dados do dicionário criado. Dessa forma é possível obter os atributos finais para inclusão no banco de dados.

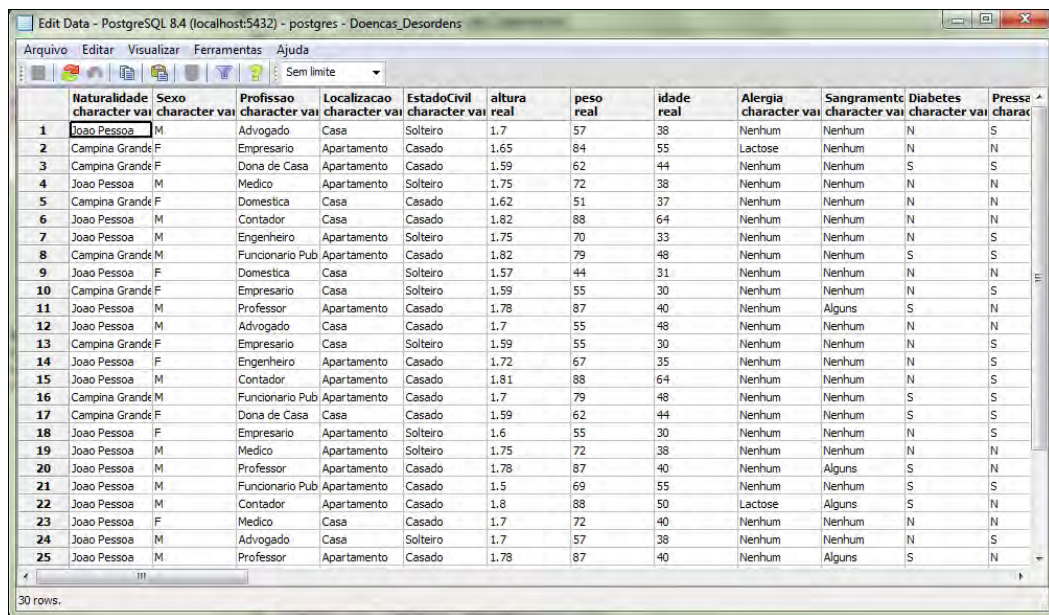
Agora que os atributos já foram escolhidos basta montar a base de dados. O povoamento dessa base foi feito partindo de dados hipotéticos, seguindo critérios especificados pelo especialista de domínio e optando por aleatoriedade em vários dos atributos. Então, quando os dados apresentarem vícios, estes devem ser evidenciados nos processos de descoberta de associações. A partir do momento que esses dados forem

⁹ Acessível no endereço eletrônico: <http://www.aihw.gov.au/publications/hwi/107/>

¹⁰ Acessível no endereço eletrônico: <http://umlsks.nlm.nih.gov>

aleatorizados as associações descobertas anteriormente devem ficar comprometidas ou até mesmo não aparecer. Esse tipo de verificação torna possível usar os dados hipotéticos para comprovar a eficiência de alguns algoritmos em relação a outros, e será dessa forma que funcionará a análise qualitativa.

Para permitir a efetivação do processo de seleção de atributos, depois do processo de analogia entre os dados do dicionário criado e os dados do questionário, foram escolhidos 50 atributos para serem selecionados dependendo do objetivo requerido na mineração de dados. Esses atributos foram agrupados em uma única estrutura (tabela) e preparados para o processo de mapeamento com a ontologia. O que é interessante comentar é que o processo de seleção de atributos com auxílio de ontologias, para esse protótipo, pode ser feito para qualquer domínio, bastando existir uma ontologia capaz de abordar os temas existentes e permitir que os mesmos sejam agrupados em suas diversas classes.



The screenshot shows a PostgreSQL database window titled 'Edit Data - PostgreSQL 8.4 (localhost:5432) - postgres - Doencas_Desordens'. The window displays a table with 13 columns and 25 rows of data. The columns are: Naturalidade, Sexo, Profissao, Localizacao, EstadoCivil, altura, peso, idade, Alergia, Sangramento, Diabetes, and Pressao. The data is organized into 25 rows, each representing a patient's attributes.

	Naturalidade	Sexo	Profissao	Localizacao	EstadoCivil	altura	peso	idade	Alergia	Sangramento	Diabetes	Pressao
	character vai	character vai	character vai	character vai	character vai	real	real	real	character vai	character vai	character vai	charac
1	Joao Pessoa	M	Advogado	Casa	Solteiro	1.7	57	38	Nenhum	Nenhum	N	S
2	Campina Grande	F	Empresario	Apartamento	Casado	1.65	84	55	Lactose	Nenhum	N	N
3	Campina Grande	F	Dona de Casa	Apartamento	Casado	1.59	62	44	Nenhum	Nenhum	S	S
4	Joao Pessoa	M	Medico	Apartamento	Solteiro	1.75	72	38	Nenhum	Nenhum	N	N
5	Campina Grande	F	Domestica	Casa	Casado	1.62	51	37	Nenhum	Nenhum	N	N
6	Joao Pessoa	M	Contador	Casa	Casado	1.82	88	64	Nenhum	Nenhum	N	N
7	Joao Pessoa	M	Engenheiro	Apartamento	Solteiro	1.75	70	33	Nenhum	Nenhum	N	S
8	Campina Grande	M	Funcionario Pub	Apartamento	Casado	1.82	79	48	Nenhum	Nenhum	S	S
9	Joao Pessoa	F	Domestica	Casa	Solteiro	1.57	44	31	Nenhum	Nenhum	N	N
10	Campina Grande	F	Empresario	Casa	Solteiro	1.59	55	30	Nenhum	Nenhum	N	S
11	Joao Pessoa	M	Professor	Apartamento	Casado	1.78	87	40	Nenhum	Alguns	S	N
12	Joao Pessoa	M	Advogado	Casa	Casado	1.7	55	48	Nenhum	Nenhum	N	N
13	Campina Grande	F	Empresario	Casa	Solteiro	1.59	55	30	Nenhum	Nenhum	N	S
14	Joao Pessoa	F	Engenheiro	Apartamento	Casado	1.72	67	35	Nenhum	Nenhum	N	S
15	Joao Pessoa	M	Contador	Apartamento	Casado	1.81	88	64	Nenhum	Nenhum	N	S
16	Campina Grande	M	Funcionario Pub	Apartamento	Casado	1.7	79	48	Nenhum	Nenhum	S	S
17	Campina Grande	F	Dona de Casa	Casa	Casado	1.59	62	44	Nenhum	Nenhum	S	S
18	Joao Pessoa	F	Empresario	Apartamento	Solteiro	1.6	55	30	Nenhum	Nenhum	N	N
19	Joao Pessoa	M	Medico	Apartamento	Solteiro	1.75	72	38	Nenhum	Nenhum	N	N
20	Joao Pessoa	M	Professor	Apartamento	Casado	1.78	87	40	Nenhum	Alguns	S	N
21	Joao Pessoa	M	Funcionario Pub	Apartamento	Casado	1.5	69	55	Nenhum	Nenhum	S	S
22	Joao Pessoa	M	Contador	Apartamento	Casado	1.8	88	50	Lactose	Alguns	S	N
23	Joao Pessoa	F	Medico	Casa	Casado	1.7	72	40	Nenhum	Nenhum	N	N
24	Joao Pessoa	M	Advogado	Casa	Solteiro	1.7	57	38	Nenhum	Nenhum	N	S
25	Joao Pessoa	M	Professor	Apartamento	Casado	1.78	87	40	Nenhum	Alguns	S	N

Figura 45 - Tabela de Atributos.

A tabela acima mostra alguns dos atributos utilizados para cada paciente (linha) e será consultada pelo protótipo SAAO e os algoritmos de mineração de dados existentes no Weka para definir os atributos importantes para determinado tipo de descoberta ou tarefa de mineração de dados.

5.2 Elaborando o Mapeamento entre a Ontologia e os Atributos Capturados

Antes de realizar qualquer procedimento com os atributos é necessário mapeá-los com as classes da ontologia criada neste trabalho, então esse tópico irá tratar do mapeamento que servirá como base para todos os possíveis cenários analisados. Foi montado um grupo inicial contendo 50 atributos e 30 instancias (linhas) que representam cada paciente analisado e que hipoteticamente devem se referir aos históricos de pacientes que estão realizando ou realizaram tratamentos para um determinado tipo de câncer.

Cada atributo teve de ser mapeado para uma classe que mais se adequasse a sua definição para que mais na frente pudesse possibilitar a categorização e seleção de atributos relacionados semanticamente. Uma função usada para auxiliar no processo de mapeamento, principalmente quando a ontologia trabalhada é muito grande, é a análise sintática comparativa entre o atributo e os termos da ontologia. Antes de mapear qualquer atributo, utiliza-se o analisador sintático para compará-lo com as classes e tentar listar as que mais se assemelham graficamente com o mesmo. Por exemplo, no momento do mapeamento foi analisado um atributo capturado e em seguida a distancia sintática (distancia de strings) obteve os seguintes resultados:

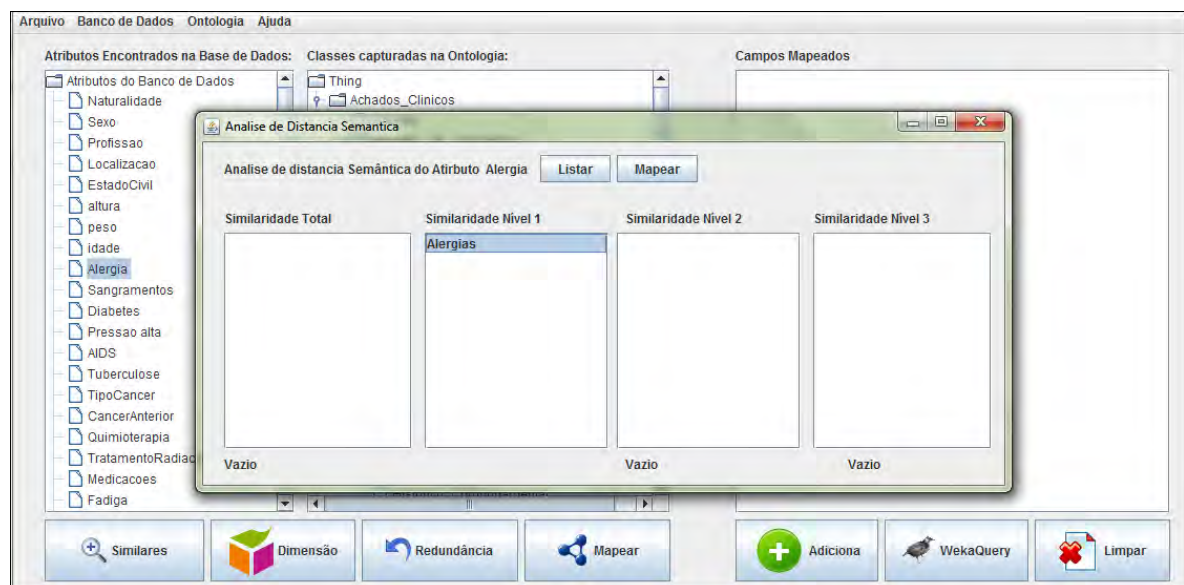


Figura 46 - Similaridade Sintática.

Na tentativa acima o analisador retorna que o atributo <Alergia_fa> apresenta nível de similaridade 1 (esse processo foi explicado no tópico 3.3.1) com a classe <Alergias> na

ontologia, e realmente nesse caso é a classe mais indicada para fazer o mapeamento. A função principal dessa etapa é buscar entre as classes da ontologia a que mais se assemelha sintaticamente com o atributo estudado no momento. Agora pela visualização da ontologia o usuário terá o atributo <Alergia_fa> mapeado na classe <Alergias>, que pertence a classe <Historico_Familiar> que por sua vez pertence a classe <Caracteristicas_Sociais>. Esse tipo de hierarquia além de ajudar o especialista a visualizar melhor o domínio ainda permite selecionar atributos relacionados semanticamente, pois todos os atributos contidos na classe <Historico_Familiar> devem ter alguma relação semântica com características da família desse paciente.

Todos os atributos devem ser mapeados para as suas classes correspondentes e sempre que necessário deve-se efetuar a procura sintática entre o atributo e a classe da ontologia. Ao final do mapeamento se obtém todos os atributos relacionados semanticamente em categorias prontas para serem selecionadas apenas com o clique do usuário.

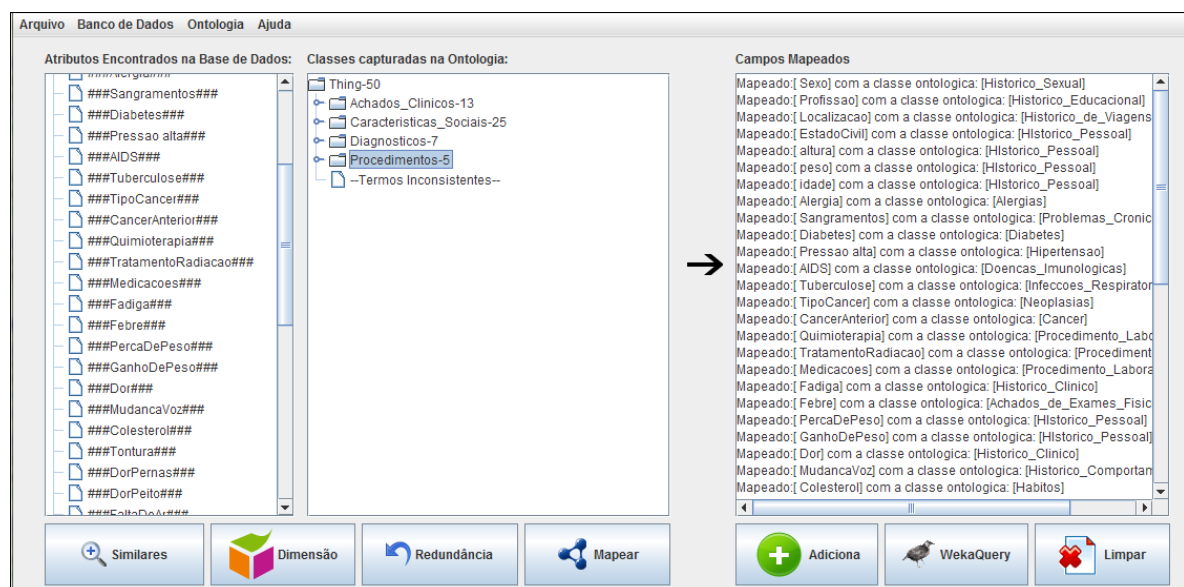


Figura 47 - Mapeando Atributos.

A visualização na imagem acima mostra que foram mapeados os 50 atributos, onde inclusive alguns deles foram considerados inconsistentes e adicionados a classe <Termos Inconsistentes>. A classe de termos inconsistentes deve abranger aqueles atributos que certamente não trarão nenhum benefício para o processo de descoberta de conhecimento, por exemplo: atributos identificadores, atributos constantes (valores repetidos não trazem nenhuma imposição nos algoritmos de mineração). Agora o módulo de redução de dimensão e

redundância pode melhorar e aperfeiçoar a qualidade dos dados utilizados, eliminando atributos redundantes ou mesmo reduzindo vários atributos a um único atributo. A partir desse momento, o especialista que estiver efetuando o processo de pré-processamento já pode fazer sua seleção de atributos por meio de categorias semânticas.

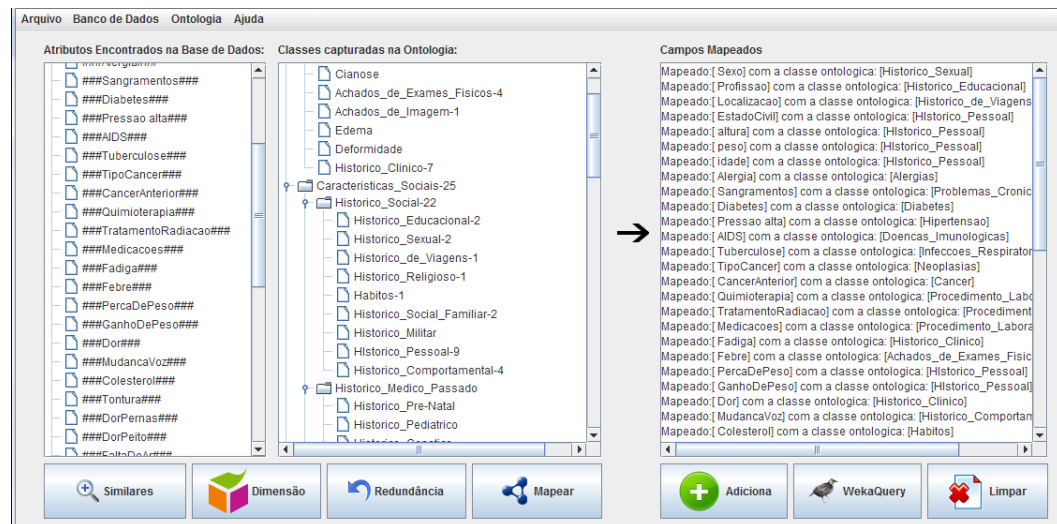


Figura 48 - Atributos mapeados (hierarquia expandida).

Em cada uma das classes mais genéricas existem subclasses que também podem ser selecionadas e seus atributos mapeados automaticamente são incluídos na seleção que será executada na ferramenta Weka. Nos próximos passos, o analista utiliza a pergunta base para efetuar a seleção entre os atributos mapeados para as várias classes. Ainda neste momento pode utilizar as avaliações de redução de dimensionalidade e tentar visualizar alguma redundância que ainda esteja presente entre os dados agrupados.

5.3 Cenários de Aplicação

Um fato que deve ser evidente é que a elaboração dos mapeamentos e principalmente a seleção das categorias realmente importantes para determinado contexto devem ser feitos por uma pessoa que entenda o domínio médico em conjunto com o especialista em KDD. Essa parceria é realmente importante para uma boa representação do domínio tratado e principalmente a correta utilização dos dados disponíveis.

Para demonstração do protótipo SAAO serão feitas comparações com algoritmos de seleção de atributos já existentes na ferramenta Weka e algoritmos para classificação, mas

que também realizam um processo de seleção de atributos, como o J48 (implementação do C4.5 no Weka). As análises realizadas entre o SAAO e os algoritmos de seleção serão:

- Primeira análise: Seleção comparativa utilizando apenas o protótipo SAAO ou algoritmos para seleção de atributos do Weka ou Algoritmos para classificação (esses algoritmos constroem árvores de decisão e os atributos não utilizados podem ser considerados irrelevantes).
- Segunda Análise: Comparação da utilização do protótipo SAAO em conjunto com algoritmos de seleção de atributos.

Como critério avaliativo para cada análise serão feitas avaliações quantitativas e qualitativas levando em conta os atributos selecionados, tarefas de associação e árvores de classificação. Os critérios qualitativos ainda serão avaliados segundo o especialista no domínio em três classificações: ótimo, bom, regular e ruim.

5.3.1 Primeira Análise: Seleção Comparativa

Um fator importante para a seleção semântica efetuada no SAAO é que os atributos incluídos são exatamente aqueles que apresentam a melhor relação semântica, visto que foram analisados por um especialista e mapeados em suas devidas classes. Enquanto isso, o Weka eliminará atributos apenas levando em conta análises matemáticas (ganhos de informação e outras análises estatísticas), ou seja, é complicado comparar um procedimento puramente semântico com outro procedimento efetivamente matemático e trazer estatísticas de qual foi o melhor método. Portanto, a primeira análise que mostra fatores quantitativos deve ser mostrada com certas ressalvas.

Essa primeira comparação tem o objetivo de mostrar os atributos selecionados no SAAO e os atributos selecionados nos algoritmos do Weka. Primeiramente utilizando todo o conjunto de atributos e executando alguns algoritmos de seleção de atributos do Weka como o Cfs (*Correlation-based Feature Subset Selection*), InfoGain, ReliefF etc. Todos os algoritmos de seleção de atributos efetuam a seleção levando em conta um atributo central, ou seja, o analista pretende descobrir quais os atributos que estão mais relacionados a esse atributo. Então, Como a base de dados que está sendo analisada contém informações sobre pacientes que tem ou tiveram câncer, então uma possível primeira análise seria: existe alguma relação entre as características sociais e o tipo de câncer apresentado pelo paciente? Não é possível

fazer esse tipo de análise semântica nos algoritmos do Weka, ou o especialista escolhe manualmente um a um cada atributo, ou especifica um atributo central e executa o processo de seleção. Então, nesse caso, o atributo mais adequado seria o tipo de câncer e ao ser executado com o algoritmo Cfs apresentou os seguintes resultados:

```

=== Attribute Selection on all input data ===

Search Method:
  Best first.
  Start set: no attributes
  Search direction: forward
  Stale search after 5 node expansions
  Total number of subsets evaluated: 689
  Merit of best subset found: 0.681

Attribute Subset Evaluator (supervised, Class (nominal): 15 TipoCancer):
  CFS Subset Evaluator
  Including locally predictive attributes

Selected attributes: 2,3,4,8,11,12,21,26,33,34,36,49 : 12
  Sexo
  Profissao
  Localizacao
  idade
  Diabetes
  Pressao alta
  Febre
  Colesterol
  SangueFezes
  Colica
  FrequenciaUrina

```

Figura 49 - Saída da seleção usando Cfs.
Fonte: Weka

Como pode ser visto na figura acima o Cfs selecionou 12 dos 50 atributos existentes no banco de dados, onde cada um deles foi listado. O resultado em termos quantitativos foi bastante satisfatório, resultando na eliminação de mais de 75% dos atributos. Porém, deve-se levar em consideração a importância desses atributos selecionados para o processo de descoberta de conhecimento e até mesmo para o objetivo colocado nessa mineração que era verificar o relacionamento entre o tipo de câncer e as características sociais do paciente. Para o algoritmo *PrincipalComponents* que trabalha primeiramente com uma matriz de correlação e depois ranqueia os atributos, obteve-se 14 atributos. Ou seja, novamente um número pequeno de atributos em relação a quantidade original. Logo abaixo será mostrada uma tabela com os resultados obtidos nesses e em outros algoritmos para seleção de atributos embutidos no Weka (todos os algoritmos foram executados com seus dados de configurações em formato padrão).

Algoritmo de Seleção de Atributos	Quantidade de Atributos selecionada
Cfs	12
Principal Components	14
ReliefF	17
InfoGain	16

Tabela 3 - Quantidade de atributos por algoritmo.

Pela análise da tabela acima se percebe que o melhor algoritmo em termos quantitativos foi o Cfs e o último lugar ficou com o ReliefF. Muitos desses algoritmos classificam os atributos por ranqueamento e não selecionam certa quantidade, ou seja, o especialista vai visualizar a classificação dada pelo mesmo e decidir a exclusão desse atributo dos dados que serão avaliados.

Para fazer a seleção dos atributos considerados importantes no SAAO o analista precisa apenas selecionar as categorias que mais se evidenciam no contexto das informações que serão estudadas. Partindo da pergunta feita na seleção anterior, no SAAO é necessário apenas selecionar as classes que apresentam maior relacionamento nesse contexto e o ambiente evidencia os atributos mapeados para que possam ser adquiridos na ferramenta Weka.

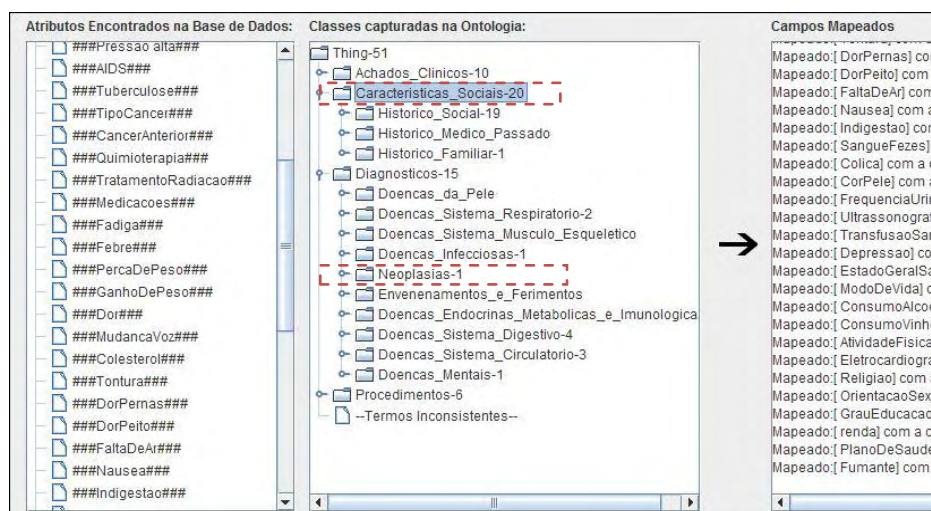


Figura 50 - Seleção de Categorias.

Na seleção feita na figura acima duas classes são adicionadas por meio do botão <Adiciona> e os atributos pertencentes as mesmas podem ser adquiridos clicando no botão <WekaQuery>, sendo devolvidos na forma de uma consulta SQL para ser usada no Weka.

Nesse caso, a classe <Caracterisricas_Sociais> e <Neoplasias> contém 21 atributos mapeados.

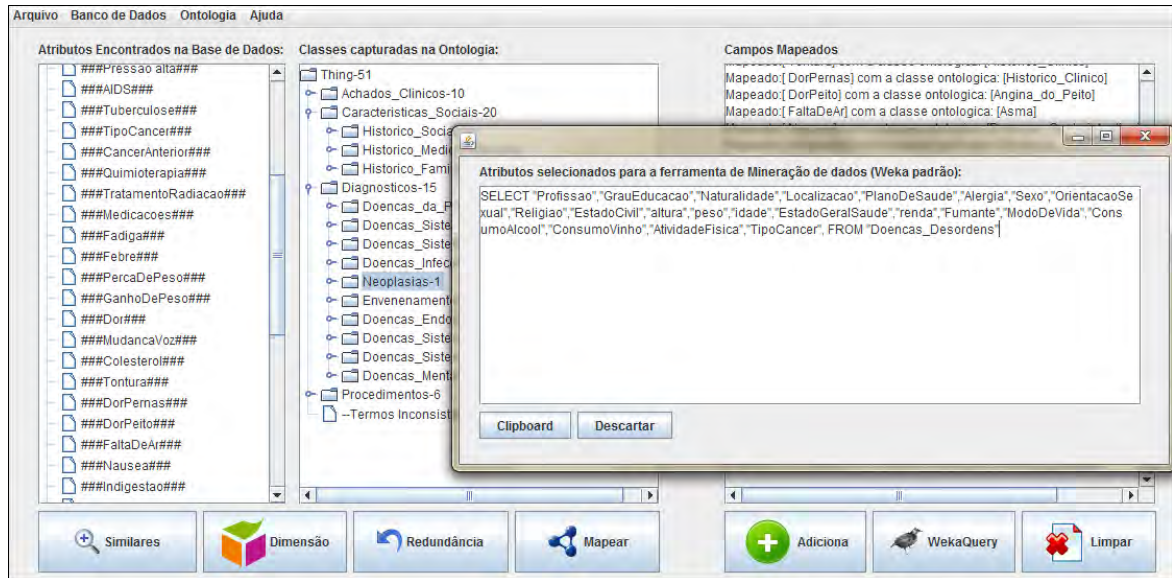


Figura 51 - Atributos resultantes da Consulta.

A primeira comparação quantitativa já pode ser feita, e como o Weka teve seus algoritmos para seleção de atributos com quantidade variando entre 12 e 16 seleções, o resultado do SAAO está um pouco acima da quantidade analisada pelos algoritmos, especificando 21 atributos no contexto especificado. Porém, os 21 atributos terão total relação com características sociais e o tipo de câncer apresentado pelo paciente, mesmo que a quantidade seja maior essa desvantagem pode trazer diversos benefícios e ampliar a interação semântica no processo de seleção de atributos. Na figura 48 é mostrado o processo de visualização dos atributos resultantes do protótipo SAAO e inserção dos mesmo no Weka por meio de uma conexão direta com o banco de dados e execução da consulta SQL resultado da *WekaQuery* (disponível no SAAO depois que os atributos mapeados são adicionados a saída).

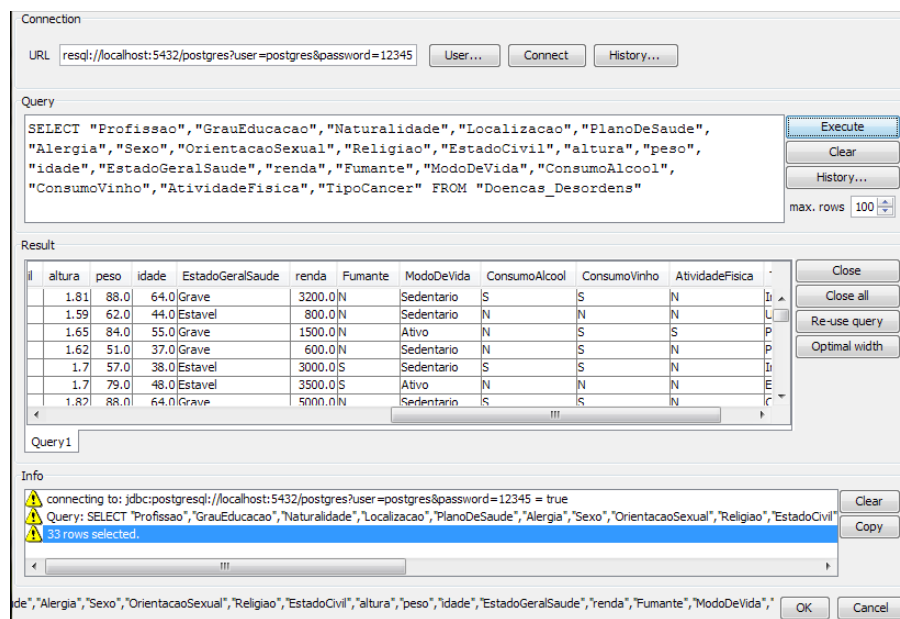


Figura 52 - SQL Viewer no Weka.

Como mais um critério avaliativo, um especialista de domínio (médico) foi consultado e em resposta ao relacionamento entre as características sociais e o tipo de câncer deveria especificar entre os atributos selecionados, por ambos os processos, quais os que ele considera mais correlacionados e possíveis de trazerem melhores resultados para os algoritmos de associação, classificação etc. Logo abaixo foi criada mais uma tabela com as classificações dadas:

Algoritmo	Ótimo	Bom	Regular	Ruim
Cfs			x	
Principal Components				x
ReliefF			x	
InfoGain		X		
SAAO	x			

Tabela 4 - Classificação do especialista de domínio (médico).

É possível concluir, pela tabela acima, que o desempenho dos algoritmos para seleção dos atributos levando em conta a pergunta base teve a média regular. Com exceção do algoritmo *InfoGain* que foi considerado bom, pois apresentou melhores atributos e não considerou alguns que os outros algoritmos inseriram e consecutivamente depreciaram o poder de classificação do modelo. Para o SAAO foi considerada a seleção ótima, visto que os atributos selecionados foram aqueles os quais os próprios especialistas classificaram. Por isso

mesmo, na fase inicial desse estudo de caso foi apresentado que a comparação de modelos matemáticos com modelos puramente semânticos são de difícil verificação. Realmente pode ser descrita nesse ponto uma pequena desvantagem do SAAO, pois o mapeamento dos atributos deve existir a priori para que o processo de seleção possa ser efetivado e esse procedimento trás certo custo que de certa forma é suprido pela riqueza de informações pós mapeamento.

5.3.2 Segunda Análise Comparativa: Usando o SAAO em Conjunto aos Algoritmos para Seleção de Atributos

Para essa segunda análise os resultados de saída do SAAO são aplicados nos algoritmos de seleção do Weka para uma nova seleção, avaliando assim a possibilidade de eliminação de atributos depois do processo semântico e as possíveis melhorias no conjunto de dados. Voltando ao modelo anterior e ao mesmo tipo de avaliação o SAAO resultou em 21 atributos que foram analisados. Para aplicar esses atributos nos algoritmos do Weka basta utilizá-los como entrada para a seleção e definir um atributo central. Como a análise continua sendo a mesma, o atributo central permanece sendo o *Tipo_Cancer* e será utilizado para cada um dos algoritmos abordados acima.

```
Attribute Subset Evaluator (supervised, Class (nominal): 19 TipoCancer):
  CFS Subset Evaluator
  Including locally predictive attributes

Selected attributes: 2,3,4,5,6,9,10,11,14,17,18,20 : 12
  Localizacao
  EstadoCivil
  CorPele
  renda
  PlanoDeSaude
  Sexo
  Religiao
  altura
  Fumante
  ConsumoVinho
  AtividadeFisica
  CancerAnterior
```

Figura 53 - Seleção de atributos pós SAAO.

Com a execução da seleção Cfs usando os atributos obtidos pelo SAAO foram encontrados 12 atributos e mais de 40% deles eram diferentes da seleção efetuada apenas pelo algoritmo Cfs de forma singular. Esse fator mostra o resultado das avaliações dos especialistas no estudo anterior, onde o Cfs ficou classificado como regular. O resultado dessa execução é importante, pois mostra que mesmo após a execução do SAAO o algoritmo para seleção de atributos do Weka ainda pode reduzir o conjunto de dados, restando apenas saber se esses dados tiveram um ganho ou uma perda da capacidade para descoberta de conhecimento. Para medir essa variação serão executadas descobertas de associações e classificação para os atributos diretamente herdados do SAAO e esses agora reduzidos com o auxílio do algoritmo Cfs.

Usando os 21 atributos selecionados no SAAO são obtidas algumas regras de associação válidas e outras que não apresentam nenhum significado real, por exemplo, a regra a seguir determina que a maioria dos tipos de câncer de estômago situa-se no grupo do sexo masculino. Logicamente essa característica foi condicionada no banco montado, mas nem sempre os algoritmos de associação vão encontrá-la, dependendo muito dos atributos envolvidos.

1.	TipoCancer=Estomago 8 ==> Sexo=M 8	conf:(1)
2.	TipoCancer=Estomago CancerAnterior=N 7 ==> Sexo=M 7	conf:(1)
3.	TipoCancer=Mama 5 ==> Sexo=F 5	conf:(1)
4.	TipoCancer=Intestino 5 ==> CancerAnterior=N 5	conf:(1)
5.	TipoCancer=Utero 4 ==> Sexo=F 4	conf:(1)
6.	TipoCancer=Utero 4 ==> CancerAnterior=S 4	conf:(1)
7.	Sexo=M TipoCancer=Intestino 4 ==> CancerAnterior=N 4	conf:(1)
8.	TipoCancer=Utero CancerAnterior=S 4 ==> Sexo=F 4	conf:(1)
9.	Sexo=F TipoCancer=Utero 4 ==> CancerAnterior=S 4	conf:(1)
0.	TipoCancer=Utero 4 ==> Sexo=F CancerAnterior=S 4	conf:(1)

Figura 54 - Regras de associação.

Veja também na figura acima que os pacientes que tiveram câncer de útero eram do sexo feminino. Essa regra deve ser óbvia, pois o sexo masculino não pode ter câncer de útero. Portanto, é visível que as regras podem ser úteis e outras já não trazem informações novas ou mesmo verdadeiras sobre os dados. Os atributos que foram selecionados condicionam diretamente os resultados das regras de associação e as árvores de classificação também são afetadas, pois também dependem dos atributos envolvidos.

Ao se aplicar os algoritmos do Weka na seleção e diminuir o número de atributos sempre serão condicionadas regras mais eficientes e precisas, pois os atributos que de certa forma enganam o classificador são eliminados do processo. A quantidade de atributos, mesmo sendo pequena em relação à quantidade original, ainda dificulta a descoberta de algumas

associações importantes pelo analista. Então, devem-se aplicar os algoritmos do Weka ao final do processo do SAAO para tentar obter melhores resultados, ou por tentativa eliminar um por um os atributos existentes e revelar valores que possam estar escondidos.

O uso da classificação também permite a eliminação de atributos, pois aqueles que não aparecem na formação dos nós podem ser considerados menos importantes no conjunto total. Logo abaixo está um exemplo de uma árvore de classificação formada com o algoritmo J48 a partir de atributos selecionados pós SAAO e com auxílio de algoritmos do Weka.

```

Sexo = M
|  AtividadeFisica = N
|  |  idade <= 38: Intestino (3.0)
|  |  idade > 38
|  |  |  idade <= 55: Estomago (7.0/1.0)
|  |  |  idade > 55: Intestino (2.0/1.0)
|  |  AtividadeFisica = S: Estomago (5.0/3.0)
Sexo = F
|  idade <= 38
|  |  ConsumoAlcool = S: Intestino (2.0/1.0)
|  |  ConsumoAlcool = N: Mama (8.0/3.0)
|  idade > 38
|  |  ModoDeVida = Sedentario: Utero (4.0)
|  |  ModoDeVida = Ativo: Pulmao (2.0)

Number of Leaves :      8
Size of the tree :     15

```

Figura 55 - Árvore de Classificação.

Evidente que as regras acima não são verdadeiras na realidade, mas mostram perfeitamente as características do banco de dados que foi povoado. Os atributos que participam do processo de construção podem ser considerados mais importantes que os que não apareceram. Esse tipo de análise permite uma seleção de atributos de forma indireta sobre os dados.

Ao final desses estudos de caso, fica visível que a ajuda semântica na seleção de atributos pode ser muito útil e se for auxiliada com os métodos existentes pode trazer resultados satisfatórios para a descoberta de conhecimento em banco de dados.

6 CONCLUSÃO E TRABALHOS FUTUROS

6.1 Conclusão

Este trabalho tratou de uma abordagem para selecionar atributos na fase de pré-processamento em KDD. Essa seleção é feita com o auxílio de uma ontologia criada especificamente para o mapeamento entre os atributos adquiridos da base de dados e as classes existentes na ontologia. O mapeamento já é uma das características mais importantes para o especialista visualizar os atributos, pois eles vão estar categorizados no mesmo molde da ontologia e automaticamente poderão ter relações semânticas. Esse tipo de relacionamento permite que o especialista selecione os atributos por categorias e com apenas um clique, o que na modelagem convencional necessitava de várias iterações para verificar atributos importantes e relacionados ao contexto pesquisado.

Outras características inerentes à ferramenta desenvolvida são a verificação de redundâncias e a redução de dimensionalidade. Ambas ajudam a diminuir o número de atributos. A primeira ajudando a identificar redundâncias através de conceitos existentes na ontologia, como as classes equivalentes. A segunda proporciona uma generalização de atributos específicos em atributos mais genéricos com a visualização de classes que representem e possam nomear um novo atributo que substitua os existentes.

6.2 Contribuições

As principais contribuições do SAAO para a Seleção de Atributos na Descoberta de Conhecimento em Banco de Dados podem ser as seguintes:

1. Estudo da interação entre ontologias e Mineração de Dados;
2. Adaptação de uma ontologia de domínio genérica e específica para mapeamento de atributos;
3. Implementação de um protótipo para manipulação e seleção de atributos com o auxílio de ontologias;
4. Manipulação de atributos através de categorias semânticas;
5. Possibilidade da visualização de atributos relacionados semanticamente;
6. Visualização de possíveis redundâncias nos atributos de entrada;
7. Redução de dimensionalidade no conjunto de atributos;

8. Consulta de saída com os atributos mapeados para ser utilizada em ferramentas de mineração de dados;
9. Criação de arquivo XML com mapeamento possível de utilização por diversas ferramentas.

6.3 Trabalhos Futuros

Como trabalhos futuros sugerem-se:

1. Melhoria e ampliação da ontologia criada para mapeamento, possibilitando melhores análises nos atributos;
2. Construção e adaptação do protótipo para a Web, seguindo assim os novos padrões e tendências;
3. Melhoria dos métodos para inferência de redundâncias e redução de dimensionalidade.

REFERÊNCIAS

- BELLINGER, G., CASTRO, D., & Mills, A. **Data, Information, Knowledge, and Wisdom**, 2005.
- BERNSTEIN, A.; HILL, S.; PROVOST, F. **Intelligent Assistance for the Data Mining Process**. CeDER Working Paper, Stern School of Business, New York University, 2002.
- BOGORNY, V.; KUIJPERS, B.; ALVARES, L.O.: **Reducing Uninteresting Spatial Association Rules in Geographic Databases using Background Knowledge: a Summary of Results**. In: IJGIS International Journal of Geographical Information Science (2008).
- BOGORNY, VANIA. **Enhancing Spatial Association Rule Mining in Geographic Databases**. Tese de Doutorado. Porto Alegre: PPGC da UFRGS, 2006.
- BORGO, N. GUARINO, C. Interfacing Ontologies and Lexical Resources. In Kaufmann, editor, **Proceedings of Knowledge Representation and Reasoning, 2002**.
- BOWERS, S.; LUDASCHER. **An Ontology-Driven Framework for Data Transformation in Scientific Workflows**. *Workshop on Data Integration in the Life Sciences (DILS'04)*, March 25-26, 2004.
- BOZSAK, E. et al. KAON: **Towards a Large Scale Semantic Web**. In: EC-WEB, 3, 2002, Aix-en-Provence. Aix-en-Provence: Lecture Notes in Computer Science, Springer, 2002. p. 304-313
- BREITMAN, K. **Web Semântica: A Internet do Futuro** –Rio de Janeiro : LTC. 2005.
- CARROL et al. **Jena: implementing the semantic web recommendations**. New York, NY, USA , 2004.
- DAVIES, J.; STUDER, R.; WARREN, P. **Semantic Web Technologies: Trends and Research in ontology-based system**. USA. Wiley. 2006.
- DAVIS, L. **Handbook if Genetic Algorihtms**. VNR Comp. Library, 1990.
- FAYYAD, U., PIATETSKY-SHAPIRO, G., and SMYTH, P. (1996a) **Knowledge discovery and data mining: towards a unifying framework.**, *Proceedings of KDD'96, Second International Conference on Knowledge Discovery & Data Mining*, pages 82-88. AAAI Press, Menlo Park, CA.
- FENSEL, D. ET AL. *Enabling Semantic Web Services the Web Service Modeling Ontology*: Springer, 2007.

FENSEL, D. ET AL. **Enabling Semantic Web Services** the Web Service Modeling Ontology: Springer, 2007.

FREITAS, A. **Data Mining and Knowledge Discovery with Evolutionary Algorithms**. Natural Computing Series, Springer-Verlag Berlin Heidelberg, 2002.

GOTTGTROY et. al. **An ontology driven approach for knowledge discovery in Biomedicine**. In: Zhang, C., Guesgen, H.W., Yeap, W.K. (Eds.), Lecture notes in artificial intelligence, Vol. 3157. Springer-verlag, Berlin.

GRUBER, T. R. **A translation approach to portable ontology specifications**. Knowledge Acquisition, v. 5, n. 2, p. 199-220, ISSN:1042-8143.1993.

GUARINO, N. (1998) **Formal Ontology and Information Systems**. in: N. Guarino, (Ed.) Formal Ontology in Information Systems. pp. 3-15, IOS Press, Amsterdam, Netherlands.

GUARINO, N. **Understanding, building and using ontologies**. In: PROCEEDINGS OF KNOWLEDGE ACQUISITION FOR KNOWLEDGE-BASED SYSTEMS WORKSHOP. 10. 1996.

HALL, M.; FRANK E.; HOLMES, G.; Bernhard Pfahringer, Peter Reutemann, Ian H. Witten (2009); **The WEKA Data Mining Software: An Update**; SIGKDD Explorations, Volume 11, Issue 1.

HAN, J. **Data Mining: Concepts and Techniques**, Simon Fraser University, 1996.

HAN, J.; KAMBER, M. **Data Mining - Concepts and Techniques**. 2a edição. Nova York: Morgan Kaufmann, 2005.

Hettich, S. and Bay, S. D. **The UCI KDD Archive**. <http://kdd.ics.uci.edu>, 1999.

HAND, D. J.; H. MANNILA, P. SMYTH. **Principles of data mining**. Cambridge, MA: MIT Press, 2005.

HARMELEN, F. MCGUINESS. **OWL Web ontology language overview**. <http://www.w3.org/TR/owl-features/>, August 2003.

HAYKIN, S. **Neural Networks: a Comprehensive Foundation**. Prentice Hall, 1997.

HAYKIN, S. **Neural Networks: a Comprehensive Foundation**. Prentice Hall, 1999.

HELLERSTEIN, J. M.; AVNUR, R; CHOU, A.; HIDBER, C.; OLSTON, C.; RAMAN, V.; ROTH, T.; HAAS, P.J. **Interactive Data Nanalysis: The Control Project**. IEEE Transactions on Computer Society.

JASPER, R.; USCHOLD, M. **A framework for understanding and classifying ontology applications.** IJCAI-99, *Ontology Workshop, 1999*, Stockholm, 1999.

JOHN, G. H., R. KOHAVI; P. PFLEGER. 1994. **Irrelevant features and the subset selection problem.** In H. Hirsh and W. Cohen, editors, *Proceedings of the Eleventh International Conference on Machine Learning*, New Brunswick, NJ. San Francisco: Morgan Kaufmann, pp. 121–129.

KEDAD, Z. and M'etais, E. **Ontology-based data cleaning.** In *NLDB '02: Proceedings of the 6th International Conference on Applications of Natural Language to Information Systems-Revised Papers*, pages 137–149, London, UK. Springer-Verlag, 2002.

LASSILA, O.; MCGUINNESS, D. **The Role of Frame-Based Representation on the Semantic Web**, Knowledge Systems Laboratory Report KSL-01-02, Stanford University, 2001; Also appeared as Linköping Electronic Articles in Computer and Information Science, Vol. 6 (2001), No. 005, Linköping University, 2001.

LIU H, MOTODA H. **Feature Selection for Knowledge Discovery and Data Mining.** Boston: Kluwer Academic Publishers, 1998.

LIVINGSTON, G.; ROSENBERGM, J. M.; BUCHANAN, B. G. **A Framework for Autonomously Performing Knowledge Discovery in Databases.** *Proceedings of sixth ACM SIGKDD International Conference on Knowledge discovery in Databases*. San Jose, 2001.

MCGUINESSs, D. L., HARMELEN, F. **OWL Web Ontology Language Overview** (W3C Candidate Recommendation 18 August 2003).

MITCHIE, D.; SPIEGELHALTER, D.; TAYLOR, C. **Machine Learning, Neural and Statistical Classifications.** Ellis Horwood, 1994.

NALIATO, F. C. **Data Mining: Técnicas e Aplicações para o Marketing Direto.** Editora Berkeley, 2001.

NIGRO, H. O.; CISARO S. G.; XODO, D. H. **Data Mining With Ontologies: Implementations, Findings and Frameworks**, Information Science Reference - Imprint of: IGI Publishing, Hershey, PA, 2007.

NOY, N.; MCGUINESS, D. **Ontology Development 101 – A guide to creating your first ontology.** KSL Technical Report, Stanford University, 2001.

ORALLO et al. **An experimental comparison of performance measures for classification.** Valencia, Spain 2004.

PEDRYCZ, W. GOMIDE, F. **An Introduction to Fuzzy Sets: Analysis and Design**, MIT Press, Cambridge. 1998.

PROTÉGÉ. **open source ontology editor and knowledge-base framework**. Stanford, California, 2006.

PYLE, D. **Data Preparation for Data Mining**. San Francisco: Morgan Kaufmann Publishers, 1999.

SILVESCU, H. et al. **Ontology-driven information extraction and knowledge acquisition from heterogeneous**, distributed biological data sources, 2001.

SIMOUDIS, E. **Discovering Data Mining**, foreword in Cabena, P et al., Upper Saddle River, Prentice Hall PTR, New Jersey, USA, 1998.

SOARES, C.JORGE, R.; MARCOS, D. **Monitoring the Quality of Meta-data in Web Portals Using Statistics, Visualization and Data Mining**, 2001.

SOUZA, F.J., VELLASCO, M.M.B.R., PACHECO, M.A. (2000). **Hierarchical Neuro Fuzzy BSP Model - HNFB**, Proceedings of the VIth Brazilian Symp. on Neural Networks Volume 2 - (ISBN 0-7695-0856-1), IEEE Computer Society, Rio de Janeiro, RJ, 22-25 Nov.

SRIKANT, R.; VU, Q.; AGRAWAL, R. **Mining association rules with item constraints**. In The third international conference on knowledge discovery in databases and data mining, Newport Beach, California, August 1997.

WEISS, S.; INDURKHYA, N. **Predictive Data Mining: a practical guide**. San Francisco: Morgan Kaufmann Publishers, 1998.

ZAKI, M. J. **Parallel and Distributed Data Mining: An Introduction**. Large-Scale Parallel Data Mining. Berlin: Springer-Verlag, 2000.

APÊNDICE(S)

APÊNDICE – Ontologia criada para utilização no protótipo SAAO

```

<?xml version="1.0"?>
<!DOCTYPE rdf:RDF [
  <!ENTITY owl "http://www.w3.org/2002/07/owl#" >
  <!ENTITY xsd "http://www.w3.org/2001/XMLSchema#" >
  <!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#" >
  <!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#" >
  <!ENTITY ProjetoSAAO3 "http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#"
>
]>
<rdf:RDF xmlns="http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#"
  xml:base="http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:ProjetoSAAO3="http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#">
  <owl:AnnotationProperty rdf:about="&rdfs;label"/>
  <!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#NaoApresentaSintomas --
>
  <owl:ObjectProperty rdf:about="&ProjetoSAAO3;NaoApresentaSintomas">
    <rdfs:subPropertyOf rdf:resource="&owl;topObjectProperty"/>
  </owl:ObjectProperty>

  <!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#TemConjuntoPatologico -
->
  <owl:ObjectProperty rdf:about="&ProjetoSAAO3;TemConjuntoPatologico">
    <rdfs:type rdf:resource="&owl;TransitiveProperty"/>
    <rdfs:subPropertyOf rdf:resource="&owl;topObjectProperty"/>
  </owl:ObjectProperty>

  <!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#TemHistorico -->
  <owl:ObjectProperty rdf:about="&ProjetoSAAO3;TemHistorico">
    <rdfs:subPropertyOf rdf:resource="&owl;topObjectProperty"/>
  </owl:ObjectProperty>

  <!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#TemPredisposicao -->
  <owl:ObjectProperty rdf:about="&ProjetoSAAO3;TemPredisposicao">
    <rdfs:subPropertyOf rdf:resource="&owl;topObjectProperty"/>
  </owl:ObjectProperty>

  <!-- http://www.w3.org/2002/07/owl#topObjectProperty -->
  <owl:ObjectProperty rdf:about="&owl;topObjectProperty"/>
  <!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#TemHistoricoFamiliar -->
  <owl:DatatypeProperty rdf:about="&ProjetoSAAO3;TemHistoricoFamiliar">
    <rdfs:subPropertyOf rdf:resource="&owl;topDataProperty"/>
  </owl:DatatypeProperty>

  <!-- http://www.w3.org/2002/07/owl#topDataProperty -->
  <owl:DatatypeProperty rdf:about="&owl;topDataProperty"/>
  <!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Achados_Clinicos -->
  <owl:Class rdf:about="&ProjetoSAAO3;Achados_Clinicos"/>

```

```

<!--
http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Achados_de_Exames_Fisicos -->
<owl:Class rdf:about="&ProjetoSAAO3;Achados_de_Exames_Fisicos">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Achados_Clinicos"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Achados_de_Imagem -->
<owl:Class rdf:about="&ProjetoSAAO3;Achados_de_Imagem">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Achados_Clinicos"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Achados_de_Laboratorio
-->
<owl:Class rdf:about="&ProjetoSAAO3;Achados_de_Laboratorio">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Achados_Clinicos"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Alergias -->
<owl:Class rdf:about="&ProjetoSAAO3;Alergias">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Historico_Familiar"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Angina_do_Peito -->
<owl:Class rdf:about="&ProjetoSAAO3;Angina_do_Peito">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Doencas_do_Coracao"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Arritmia_Cardiaca -->
<owl:Class rdf:about="&ProjetoSAAO3;Arritmia_Cardiaca">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Doencas_do_Coracao"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Asma -->
<owl:Class rdf:about="&ProjetoSAAO3;Asma">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Doencas_Sistema_Respiratorio"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Aterosclerose -->
<owl:Class rdf:about="&ProjetoSAAO3;Aterosclerose">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Doencas_do_Coracao"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Calculo -->
<owl:Class rdf:about="&ProjetoSAAO3;Calculo">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Achados_Clinicos"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Cancer -->
<owl:Class rdf:about="&ProjetoSAAO3;Cancer">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Historico_Familiar"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Cancer_de_Mama -->
<owl:Class rdf:about="&ProjetoSAAO3;Cancer_de_Mama">

```

```

    <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Neoplasias"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Cancer_de_Pele -->
<owl:Class rdf:about="&ProjetoSAAO3;Cancer_de_Pele">
    <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Neoplasias"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Caracteristicas_Sociais --
>
<owl:Class rdf:about="&ProjetoSAAO3;Caracteristicas_Sociais"/>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Cianose -->
<owl:Class rdf:about="&ProjetoSAAO3;Cianose">
    <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Achados_Clinicos"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Deficiencias_Nutricionais
-->
<owl:Class rdf:about="&ProjetoSAAO3;Deficiencias_Nutricionais">
    <rdfs:subClassOf
rdf:resource="&ProjetoSAAO3;Doencas_Endocrinas_Metabolicas_e_Imunologicas"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Deformidade -->
<owl:Class rdf:about="&ProjetoSAAO3;Deformidade">
    <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Achados_Clinicos"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Deformidades_Acquiridas
-->
<owl:Class rdf:about="&ProjetoSAAO3;Deformidades_Acquiridas">
    <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Doencas_Sistema_Musculo_Esqueletico"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Diabetes -->
<owl:Class rdf:about="&ProjetoSAAO3;Diabetes">
    <rdfs:subClassOf
rdf:resource="&ProjetoSAAO3;Doencas_Endocrinas_Metabolicas_e_Imunologicas"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Diagnosticos -->
<owl:Class rdf:about="&ProjetoSAAO3;Diagnosticos"/>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Doencas_Cerebro-
Vasculares -->
<owl:Class rdf:about="&ProjetoSAAO3;Doencas_Cerebro-Vasculares">
    <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Doencas_Sistema_Circulatorio"/>
</owl:Class>

<!--
http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Doencas_Endocrinas_Metabolica
s_e_Imunologicas -->
<owl:Class rdf:about="&ProjetoSAAO3;Doencas_Endocrinas_Metabolicas_e_Imunologicas">

```

```

    <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Diagnosticos"/>
  </owl:Class>

  <!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Doencas_Gastrointestinais -->
  <owl:Class rdf:about="&ProjetoSAAO3;Doencas_Gastrointestinais">
    <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Doencas_Sistema_Digestivo"/>
  </owl:Class>

  <!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Doencas_Geneticas -->
  <owl:Class rdf:about="&ProjetoSAAO3;Doencas_Geneticas">
    <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Historico_Familiar"/>
  </owl:Class>

  <!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Doencas_Imunologicas -->
  <owl:Class rdf:about="&ProjetoSAAO3;Doencas_Imunologicas">
    <rdfs:subClassOf
rdf:resource="&ProjetoSAAO3;Doencas_Endocrinas_Metabolicas_e_Imunologicas"/>
  </owl:Class>

  <!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Doencas_Infecciosas -->
  <owl:Class rdf:about="&ProjetoSAAO3;Doencas_Infecciosas">
    <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Diagnosticos"/>
  </owl:Class>

  <!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Doencas_Mentais -->
  <owl:Class rdf:about="&ProjetoSAAO3;Doencas_Mentais">
    <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Diagnosticos"/>
  </owl:Class>

  <!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Doencas_Pancreaticas -->
  <owl:Class rdf:about="&ProjetoSAAO3;Doencas_Pancreaticas">
    <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Doencas_Sistema_Digestivo"/>
  </owl:Class>

  <!--
http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Doencas_Sistema_Circulatorio -->
  <owl:Class rdf:about="&ProjetoSAAO3;Doencas_Sistema_Circulatorio">
    <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Diagnosticos"/>
  </owl:Class>

  <!--
http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Doencas_Sistema_Digestivo -->
  <owl:Class rdf:about="&ProjetoSAAO3;Doencas_Sistema_Digestivo">
    <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Diagnosticos"/>
  </owl:Class>

  <!--
http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Doencas_Sistema_Musculo_Esqueletico -->
  <owl:Class rdf:about="&ProjetoSAAO3;Doencas_Sistema_Musculo_Esqueletico">
    <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Diagnosticos"/>

```

```

</owl:Class>

<!--
http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Doencas_Sistema_Respiratorio --
>
<owl:Class rdf:about="&ProjetoSAAO3;Doencas_Sistema_Respiratorio">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Diagnosticos"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Doencas_da_Bile -->
<owl:Class rdf:about="&ProjetoSAAO3;Doencas_da_Bile">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Doencas_Sistema_Digestivo"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Doencas_da_Pele -->
<owl:Class rdf:about="&ProjetoSAAO3;Doencas_da_Pele">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Diagnosticos"/>
</owl:Class>

<!--
http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Doencas_das_Arterias_e_Capilar
es -->
<owl:Class rdf:about="&ProjetoSAAO3;Doencas_das_Arterias_e_Capilares">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Doencas_Sistema_Circulatorio"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Doencas_do_Coracao -->
<owl:Class rdf:about="&ProjetoSAAO3;Doencas_do_Coracao">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Doencas_Sistema_Circulatorio"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Edema -->
<owl:Class rdf:about="&ProjetoSAAO3;Edema">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Achados_Clinicos"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Envenenamento_Drogas -
->
<owl:Class rdf:about="&ProjetoSAAO3;Envenenamento_Drogas">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Envenenamentos_e_Ferimentos"/>
</owl:Class>

<!--
http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Envenenamento_Substancias -->
<owl:Class rdf:about="&ProjetoSAAO3;Envenenamento_Substancias">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Envenenamentos_e_Ferimentos"/>
</owl:Class>

<!--
http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Envenenamentos_e_Ferimentos -
->
<owl:Class rdf:about="&ProjetoSAAO3;Envenenamentos_e_Ferimentos">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Diagnosticos"/>
</owl:Class>

```

```

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Ernia_Abdominal -->
<owl:Class rdf:about="&ProjetoSAAO3;Ernia_Abdominal">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Doencas_Sistema_Digestivo"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Esquizofrenia -->
<owl:Class rdf:about="&ProjetoSAAO3;Esquizofrenia">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Doencas_Mentais"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Fibrose_Cistica -->
<owl:Class rdf:about="&ProjetoSAAO3;Fibrose_Cistica">
  <rdfs:subClassOf
rdf:resource="&ProjetoSAAO3;Doencas_Endocrinas_Metabolicas_e_Imunologicas"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Fratura -->
<owl:Class rdf:about="&ProjetoSAAO3;Fratura">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Envenenamentos_e_Ferimentos"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Fraturas_Patologicas -->
<owl:Class rdf:about="&ProjetoSAAO3;Fraturas_Patologicas">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Doencas_Sistema_Musculo_Esqueletico"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#HHistorico_Pessoal -->
<owl:Class rdf:about="&ProjetoSAAO3;HHistorico_Pessoal">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Historico_Social"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Habitos -->
<owl:Class rdf:about="&ProjetoSAAO3;Habitos">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Historico_Social"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Hipertensao -->
<owl:Class rdf:about="&ProjetoSAAO3;Hipertensao">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Doencas_Sistema_Circulatorio"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Historico_Clinico -->
<owl:Class rdf:about="&ProjetoSAAO3;Historico_Clinico">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Achados_Clinicos"/>
</owl:Class>

<!--
http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Historico_Comportamental -->
<owl:Class rdf:about="&ProjetoSAAO3;Historico_Comportamental">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Historico_Social"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Historico_Educacional -->

```

```

<owl:Class rdf:about="&ProjetoSAAO3;Historico_Educacional">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Historico_Social"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Historico_Exposicoes -->
<owl:Class rdf:about="&ProjetoSAAO3;Historico_Exposicoes">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Historico_Medico_Passado"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Historico_Familiar -->
<owl:Class rdf:about="&ProjetoSAAO3;Historico_Familiar">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Caracteristicas_Sociais"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Historico_Genetico -->
<owl:Class rdf:about="&ProjetoSAAO3;Historico_Genetico">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Historico_Medico_Passado"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Historico_Gravidéz -->
<owl:Class rdf:about="&ProjetoSAAO3;Historico_Gravidéz">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Historico_Medico_Passado"/>
</owl:Class>

<!--
http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Historico_Medico_Passado -->
<owl:Class rdf:about="&ProjetoSAAO3;Historico_Medico_Passado">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Caracteristicas_Sociais"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Historico_Militar -->
<owl:Class rdf:about="&ProjetoSAAO3;Historico_Militar">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Historico_Social"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Historico_Pediatrico -->
<owl:Class rdf:about="&ProjetoSAAO3;Historico_Pediatrico">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Historico_Medico_Passado"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Historico_Pre-Natal -->
<owl:Class rdf:about="&ProjetoSAAO3;Historico_Pre-Natal">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Historico_Medico_Passado"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Historico_Religioso -->
<owl:Class rdf:about="&ProjetoSAAO3;Historico_Religioso">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Historico_Social"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Historico_Sexual -->
<owl:Class rdf:about="&ProjetoSAAO3;Historico_Sexual">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Historico_Social"/>
</owl:Class>

```

```

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Historico_Social -->
<owl:Class rdf:about="&ProjetoSAAO3;Historico_Social">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Caracteristicas_Sociais"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Historico_Social_Familiar -->
<owl:Class rdf:about="&ProjetoSAAO3;Historico_Social_Familiar">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Historico_Social"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Historico_de_Viagens -->
<owl:Class rdf:about="&ProjetoSAAO3;Historico_de_Viagens">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Historico_Social"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Infarto_do_Miocardio -->
<owl:Class rdf:about="&ProjetoSAAO3;Infarto_do_Miocardio">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Doencas_do_Coracao"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Infeccao_Bacterias -->
<owl:Class rdf:about="&ProjetoSAAO3;Infeccao_Bacterias">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Doencas_Infecciosas"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Infeccao_Virus -->
<owl:Class rdf:about="&ProjetoSAAO3;Infeccao_Virus">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Doencas_Infecciosas"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Infeccoes_Respiratorias -->
<owl:Class rdf:about="&ProjetoSAAO3;Infeccoes_Respiratorias">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Doencas_Sistema_Respiratorio"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Inflamacoes -->
<owl:Class rdf:about="&ProjetoSAAO3;Inflamacoes">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Doencas_da_Pele"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Lupus -->
<owl:Class rdf:about="&ProjetoSAAO3;Lupus">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Doencas_Sistema_Musculo_Esqueletico"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Micoses -->
<owl:Class rdf:about="&ProjetoSAAO3;Micoses">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Doencas_Infecciosas"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Neoplasias -->

```

```

<owl:Class rdf:about="&ProjetoSAAO3;Neoplasias">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Diagnosticos"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Osteoporose -->
<owl:Class rdf:about="&ProjetoSAAO3;Osteoporose">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Doencas_Sistema_Musculo_Esqueletico"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Pneumonia -->
<owl:Class rdf:about="&ProjetoSAAO3;Pneumonia">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Doencas_Sistema_Respiratorio"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Problemas_Cardiacos -->
<owl:Class rdf:about="&ProjetoSAAO3;Problemas_Cardiacos">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Historico_Familiar"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Problemas_Cronicos -->
<owl:Class rdf:about="&ProjetoSAAO3;Problemas_Cronicos">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Historico_Familiar"/>
</owl:Class>

<!--
http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Procedimento_Laboratorial -->
<owl:Class rdf:about="&ProjetoSAAO3;Procedimento_Laboratorial">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Procedimentos"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Procedimento_Obstetrico
-->
<owl:Class rdf:about="&ProjetoSAAO3;Procedimento_Obstetrico">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Procedimentos"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Procedimentos -->
<owl:Class rdf:about="&ProjetoSAAO3;Procedimentos"/>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Psicose -->
<owl:Class rdf:about="&ProjetoSAAO3;Psicose">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Doencas_Mentais"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Queimadura -->
<owl:Class rdf:about="&ProjetoSAAO3;Queimadura">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Envenenamentos_e_Ferimentos"/>
</owl:Class>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Transtornos_Afetivos -->
<owl:Class rdf:about="&ProjetoSAAO3;Transtornos_Afetivos">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Doencas_Mentais"/>
</owl:Class>

```

```

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#Ulceras -->
<owl:Class rdf:about="&ProjetoSAAO3;Ulceras">
  <rdfs:subClassOf rdf:resource="&ProjetoSAAO3;Doencas_da_Pele"/>
</owl:Class>
<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#DoenteDoCoracao -->
<owl:NamedIndividual rdf:about="&ProjetoSAAO3;DoenteDoCoracao">
  <rdf:type rdf:resource="&ProjetoSAAO3;Doencas_do_Coracao"/>
  <rdf:type rdf:resource="&ProjetoSAAO3;Problemas_Cardiacos"/>
  <TemHistorico rdf:resource="&ProjetoSAAO3;DoenteDoCoracao"/>
</owl:NamedIndividual>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#PacienteAngina1 -->
<owl:NamedIndividual rdf:about="&ProjetoSAAO3;PacienteAngina1">
  <rdf:type rdf:resource="&ProjetoSAAO3;Angina_do_Peito"/>
  <rdfs:label rdf:datatype="&xsd:string">Diagnostico mais raro</rdfs:label>
  <TemConjuntoPatologico rdf:resource="&ProjetoSAAO3;PacienteAngina1"/>
</owl:NamedIndividual>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#PacienteAngina2 -->
<owl:NamedIndividual rdf:about="&ProjetoSAAO3;PacienteAngina2">
  <rdf:type rdf:resource="&ProjetoSAAO3;Angina_do_Peito"/>
  <rdfs:label rdf:datatype="&xsd:string">Diagnostico convencional</rdfs:label>
  <TemPredisposicao rdf:resource="&ProjetoSAAO3;PacienteAngina2"/>
</owl:NamedIndividual>

<!-- http://www.semanticweb.org/ontologies/2010/8/ProjetoSAAO3.owl#PacienteArritmia1 -->
<owl:NamedIndividual rdf:about="&ProjetoSAAO3;PacienteArritmia1">
  <rdf:type rdf:resource="&ProjetoSAAO3;Arritmia_Cardiaca"/>
</owl:NamedIndividual>
<rdf:Description>
  <TemHistoricoFamiliar></TemHistoricoFamiliar>
</rdf:Description>
<rdf:Description>
  <TemHistorico rdf:resource="&ProjetoSAAO3;PacienteAngina1"/>
</rdf:Description>
<rdf:Description>
  <rdf:type rdf:resource="&owl;AllDifferent"/>
  <owl:distinctMembers rdf:parseType="Collection">
    <rdf:Description rdf:about="&ProjetoSAAO3;PacienteAngina1"/>
    <rdf:Description rdf:about="&ProjetoSAAO3;PacienteAngina2"/>
  </owl:distinctMembers>
</rdf:Description>
<rdf:Description>
  <rdf:type rdf:resource="&owl;AllDifferent"/>
  <owl:distinctMembers rdf:parseType="Collection">
    <rdf:Description rdf:about="&ProjetoSAAO3;PacienteAngina2"/>
    <rdf:Description rdf:about="&ProjetoSAAO3;PacienteArritmia1"/>
  </owl:distinctMembers>
</rdf:Description>
<rdf:Description>
  <rdf:type rdf:resource="&owl;AllDifferent"/>
  <owl:distinctMembers rdf:parseType="Collection">
    <rdf:Description rdf:about="&ProjetoSAAO3;PacienteAngina1"/>
    <rdf:Description rdf:about="&ProjetoSAAO3;PacienteArritmia1"/>
  </owl:distinctMembers>
</rdf:Description>

```

```
    </owl:distinctMembers>
  </rdf:Description>
</rdf:RDF>
```