
**UNIVERSIDADE SALVADOR - UNIFACS**
**MESTRADO EM SISTEMAS E COMPUTAÇÃO**

**REINALDO DE FIGUEIRÊDO ALMEIDA**

# IDENTIFICAÇÃO DE TÉCNICAS PARA DETECÇÃO AUTOMÁTICA DE FRAUDES ATRAVÉS DE ONTOLOGIAS

Salvador
2008

**REINALDO DE FIGUEIRÊDO ALMEIDA**

# IDENTIFICAÇÃO DE TÉCNICAS PARA DETECÇÃO AUTOMÁTICA DE FRAUDES ATRAVÉS DE ONTOLOGIAS

Dissertação elaborada junto ao programa de Mestrado em Sistemas e Computação, Universidade Salvador – UNIFACS, como requisito parcial para obtenção do grau de Mestre em Sistemas e Computação.

Orientadora: Prof[a] Laís Salvador, Dsc.

Salvador
2008

FICHA CATALOGRÁFICA
(Elaborada pelo Sistema de Bibliotecas da Universidade Salvador - UNIFACS)

Almeida, Reinaldo de Figueiredo
Identificação de técnicas para detecção automática de fraudes através de ontologias / Reinaldo de Figueiredo Almeida. - 2009.
176 f.

Dissertação (Mestrado) - Universidade Salvador – UNIFACS.
Mestrado em Sistemas de Computação, 2008.
Orientador: Prof. Laís do Nascimento Salvador

1.Ontologia. I. Salvador, Laís do Nascimento, orient. II. Título.

CDD: 004

**REINALDO DE FIGUEIRÊDO ALMEIDA**

# IDENTIFICAÇÃO DE TÉCNICAS PARA DETECÇÃO AUTOMÁTICA DE FRAUDES ATRAVÉS DE ONTOLOGIAS

Dissertação aprovada como requisito parcial para obtenção do grau de Mestre em Sistemas e Computação, Universidade Salvador – UNIFACS, pela seguinte banca examinadora:

Laís Salvador – Orientadora _____

Doutora em Engenharia Elétrica, Universidade de São Paulo (USP)

Universidade Salvador – UNIFACS

Daniela Barreiro Claro – _____

Doutora em Informatique, Université d'Angers

Universidade Federal da Bahia – UFBA

André Santanchè – _____

Doutor em Ciência da Computação, Universidade Estadual de Campinas (UNICAMP)

Universidade Salvador – UNIFACS

18 de dezembro de 2008.

## RESUMO

As ocorrências de fraudes nas organizações têm se tornado algo complexo e desafiador, seja pelo aumento no número de casos identificados, seja pelo aumento no nível de sofisticação dos mesmos. Isto tem levado ao desenvolvimento por organizações públicas e privadas, de iniciativas tanto em relação à prevenção (medidas tomadas para inibir a ocorrência da fraude no primeiro momento) como em relação à detecção (medidas tomadas para que um evento fraudulento seja identificado o mais rápido possível). Estas iniciativas envolvem, a promulgação de leis específicas sobre o tema, o desenvolvimento de *frameworks* para a gestão do ciclo de vida de fraudes (com a identificação detalhada de todos os estágios envolvidos), indo até a implementação de técnicas para detecção automática de fraudes (baseadas em estatística, inteligência artificial e mineração de dados). Entretanto, se em relação às políticas de conformidade e aos *frameworks* para gestão de problemas de fraude, existe um movimento de consolidação e de padronização de conhecimentos e procedimentos, no que se refere a técnicas para detecção automática de fraudes, verifica-se uma lacuna quanto à efetiva relação entre a técnica empregada e o domínio do problema de fraude. Dentro deste contexto, esta dissertação propõe as seguintes ontologias: uma para problemas de fraude, uma para técnicas para detecção automática de fraudes e uma terceira ontologia, gerada a partir do *merging* entre as duas primeiras. Estas ontologias têm como objetivo, a disponibilização de bases de conhecimentos para que possam ser utilizadas em aplicações capazes de responder a questão fundamental: q*ual ou quais as técnicas para detecção automática são mais adequadas a um domínio de problema de fraude específico*? As ontologias implementadas por este trabalho foram elaboradas com base na metodologia Uschold & King´s *method*, codificadas na linguagem *Ontology Web Language* (OWL), e com apoio da ferramenta *Protégé* versão 4.0 beta. Também compõe esta dissertação, uma proposta de extensão do *framework* de avaliação de metodologias, elaborado por Gómez-Pérez, Fernández-López e Corcho, com a finalidade de se obter um esquema com capacidade de sugerir qual a metodologia mais adequada para construção de uma ontologia em um determinado contexto.



**Palavras-chave:** Ontologia. Gestão do conhecimento. Representação do conhecimento. Fraude. Técnicas para detecção automática de fraude.

**ABSTRACT**

The occurrences of fraud in organizations have become quite complex and challenging, is the increase in the number of cases identified, either by increasing the level of sophistication of them. This has led to the development by public and private organizations, initiatives regarding both the prevention (measures taken to inhibit the occurrence of fraud in the first instance) as for the detection (measures taken to an event that fraud is identified as soon as possible). These initiatives involve the enactment of specific laws on the subject, developing frameworks for managing the lifecycle of fraud (with the detailed identification of all stages involved), by going to the implementation of techniques for automatic detection of fraud (based on statistical, artificial intelligence and data mining). However, in relation to policies for compliance and the frameworks to manage problems of fraud, there is a movement of consolidation and standardization of knowledge and procedures in regard to techniques for automatic detection of fraud, there is a gap in the effective relationship between the technique and mastery of the problem of fraud. Within this context, this dissertation proposes: an ontology to problems of fraud, one for automatic detection techniques of fraud, and a third ontology generated from the merging between the first two. This ontologies aim at providing the bases of knowledge to be used in applications capable of meeting the fundamental question: *what or where the automatic detection techniques are more appropriate to a field of specific problem of fraud*? The ontologies implemented in this work were prepared based on the methodology Uschold & King's method, the coded language Web Ontology Language (OWL), and with support from the Protégé tool version 4.0 beta. Also composed this dissertation, a proposal to extend the framework of assessment methodologies, developed by Gómez-Pérez, Fernández-López and Corcho, in order to obtain a scheme with the ability to suggest what the most appropriate methodology for building an ontology in a given context.

**Keywords:** Ontology. Knowledge management. Knowledge representation. Fraud. Automatic etection techniques of fraud.

# SUMÁRIO

# CAPÍTULO 1

## 1 INTRODUÇÃO

Segundo a *Association of Certified Fraud Examiners* (ACFE) *apud* Silverstone e Davia (2005), “a questão das fraudes é um desafio mundial, que tem tomado múltiplas formas”. A partir de um *survey* conduzido em 2007 pela empresa americana Kroll, junto a novecentos executivos *seniors* de importantes organizações nos cinco continentes, foram obtidos os seguintes resultados (ECONOMIST INTELLIGENCE UNIT AND KROLL, 2007):

a) Entre os anos de 2005 e 2007, quatro dentre cinco organizações pesquisadas sofreram com algum tipo de fraude;

b) Neste período, as fraudes variaram desde o desvio de bens físicos (trinta e cinco por cento) até o roubo de informações (vinte por cento);

c) As perdas provocadas pelas fraudes no período pesquisado giraram em torno de trinta milhões de dólares em média por instituição fraudada.

Para minorar as ocorrências de fraudes nas organizações muitas iniciativas vêm sendo desenvolvidas, tanto no que tange à prevenção (medidas tomadas para inibir a ocorrência da fraude no primeiro momento) como em relação à detecção (medidas tomadas para que um evento fraudulento seja identificado o mais rápido possível) (BOLTON; HAND, 2002).

Estas iniciativas se iniciam com o aperfeiçoamento das políticas de conformidade para atuação das organizações, como a promulgação da lei norte-americana em 30 de julho de 2002, *Sarbanes-Oxley Act*, voltada para a prevenção de ocorrência de eventos fraudulentos a partir de uma maior transparência dos balanços contábeis das empresas (SILVERSTONE; DAVIA, 2005). Seguem através do desenvolvimento de *frameworks* para a gestão do ciclo de vida de fraudes, com a identificação detalhada de todas as fases envolvidas (WILHELM, 2004). Indo até a disponibilização de ferramentas baseadas em técnicas de estatística, de inteligência artificial e de mineração de dados para detecção automática de fraudes (PHUA *et al.*, 2005).

Entretanto, se em relação às políticas de conformidade e aos *frameworks* para gestão de problemas de fraude, existe um movimento de consolidação e de padronização de conhecimentos e procedimentos, no que se refere a técnicas para detecção automática de

fraudes, verifica-se uma lacuna quanto à efetividade da relação entre a técnica empregada e o domínio do problema de fraude (ALLEN, 1999; BOLTON; HAND, 2002).

Dentro deste contexto, esta dissertação propõe desenvolver ontologias com os seguintes domínios: problemas de fraude, técnicas para detecção automática de fraudes, e um *merge* entre as duas. Isto tem como objetivo, disponibilizar bases de conhecimentos para que possam ser utilizadas em aplicações capazes de responder uma questão fundamental:

> *Qual ou quais as técnicas para detecção automática são mais adequadas a um domínio de problema de fraude específico*?

## 1.1 MOTIVAÇÃO

Problemas de fraude têm dimensões globais e em todo o mundo os custos provocados pelas mesmas são repassados para a sociedade, principalmente sob a forma de elevação de preços de bens e serviços, do aumento das atividades criminosas financiadas por ganhos fraudulentos por um lado, e de enormes perdas financeiras e de imagem das instituições públicas e privadas (WILHELM, 2004).

Segundo alguns levantamentos, há uma estimativa anual nos Estados Unidos, de perdas aproximadas de setenta bilhões de dólares em requisições fraudulentas (FEDERAL BUREAU OF INVESTIGATION - FBI, 2006). No mundo, estimam-se, por exemplo, perdas na ordem de cento e cinqüenta bilhões de dólares por ano, em fraudes na área de telecomunicações (MENA *apud* WILHELM, 2004), duzentos e sessenta e cinco bilhões de dólares no domínio de cartões de crédito, entre outras (WILHELM, 2004).

No Relatório Anual sobre Ocorrências de Fraudes, publicado pela ACFE em 2002 nos Estados Unidos, está contida a seguinte frase “nós temos que aceitar que nenhum negócio e nenhuma pessoa estão imunes à fraude” (ACFE *apud* SILVERSTONE; DAVIA, 2005).

Esta afirmação é confirmada pelo *survey* sobre fraudes, elaborado com a participação de novecentos executivos *seniors* de empresas de todo o mundo, e apresentado no *Global Fraud Report* (ECONOMIST INTELLIGENCE UNIT AND KROLL, 2007), segundo o qual, em relação a eventos de fraudes, as organizações pesquisadas se definiram como altamente vulneráveis (vinte por cento), vulneráveis (cinqüenta e sete por cento), pouco vulneráveis (treze por cento), sem vulnerabilidades significativas (três por cento) e não responderam (sete

por cento). A causa mais freqüente para vulnerabilidade, segundo a mesma pesquisa, foi apontada como sendo o aumento na complexidade do ambiente de informações (trinta e um por cento).

Na tentativa de melhorar esta situação, segundo o *Global Fraud Report*, a maioria das empresas que forneceram informações (mais de setenta por cento) tem concentrado seus investimentos nas áreas de *information technology* (TI) e de finanças, visando obter mecanismos de prevenção e detecção de fraudes, pois estas áreas concentram os maiores aumentos deste tipo de evento (ECONOMIST INTELLIGENCE UNIT AND KROLL, 2007).

Em relação à área de TI, uma boa parte dos investimentos tem se concentrado no desenvolvimento e na implantação de ferramentas contendo técnicas para detecção automática de fraudes (ABBOTT *et al.*, 1998). Neste segmento, tem havido uma maior disponibilização de ferramentas, que apesar de embutirem técnicas estatísticas, de inteligência artificial e de *data mining*, estão orientadas para usuários não especialistas nestas tecnologias, mas que sejam especialistas em detecção de fraudes nos seus respectivos domínios (ABBOTT *et al.*, 1998).

O problema, no entanto, está na carência de conhecimentos que demonstrem a efetividade das técnicas empregadas para detecção automática de fraudes e os seus respectivos domínios. Esta é a conclusão obtida a partir de *surveys* elaborados sobre técnicas para detecção automática de fraudes (ABBOTT *et al.*, 1998; PHUA *et al.*, 2005; BOLTON; HAND, 2002; PHUA, 2003; YUFENG *et al.*, 2004). Segundo estes *surveys*, há uma enorme lacuna em relação à existência de informações que justifiquem a aplicação de uma determinada técnica para detecção automática para problemas de fraude frente a um domínio específico, quando se avaliam os resultados alcançados.

Esta lacuna, de acordo com o parágrafo anterior, refere-se a não existência de uma definição mais precisa sobre uma estrutura de classes para o domínio de problemas de fraude, o que prejudicaria uma avaliação mais adequada sobre quais as técnicas para detecção automática deveriam ser utilizadas frente a um processo de detecção de fraude (*fraud detection*), implicando (PROVOST *apud* BOLTON; HAND, 2002):

> 1º) No uso de uma mesma técnica de modo generalizado para diferentes domínios de fraudes, sem considerar que apesar de parecerem

> consideravelmente similares, os mesmos possuem características distintas tanto em relação à natureza do próprio domínio quanto à natureza dos dados relacionados ao mesmo;
>
> 2º) Na falta de observação sobre as diversas características para as diferentes técnicas para detecção de fraudes, tais como, padrões de temporalidade, modo de estimar probabilidades, quando da seleção das mesmas visando à aplicação junto a um domínio;
>
> 3º) Na falta de comparação entre técnicas diferentes aplicadas para um mesmo domínio de fraude, considerando os contextos estabelecidos. Uma vez que, o sucesso em um contexto não implica necessariamente no sucesso em outro completamente diferente, mesmo ocorrendo no mesmo domínio.

Em geral, deve-se considerar que, durante os processos de detecção de fraudes, a escolha das técnicas pode depender tanto de questões relacionadas com exigências operacionais, de restrições a recursos e de compromissos das gerências para a redução da fraude, quanto de questões referentes a características das técnicas para detecção automática a serem empregadas e dos dados a serem analisados (PHUA *et al.*, 2005).

Deste modo, o desenvolvimento de pesquisas objetivando a construção de taxonomias para domínios específicos (problemas de detecção de fraudes) com plena caracterização, a fim de associá-los com outra taxonomia, voltada para técnicas para detecção automática de fraude, é uma necessidade já estabelecida neste campo do conhecimento (PROVOST *apud* BOLTON; HAND, 2002).

## 1.2 OBJETIVO

Considerando o exposto, o objetivo desde trabalho é construir um base de conhecimentos composta pelas ontologias para técnicas para detecção automática de fraudes e para problemas de fraudes, e pelo *merge* entre as duas. A finalidade é atender uma demanda determinada por Provost, quando o mesmo se referiu à necessidade de se associar de forma efetiva, o domínio de problemas de fraude às técnicas para detecção a serem utilizadas durante o processo de *fraud detection* (PROVOST *apud* BOLTON; HAND, 2002).

Em relação às ontologias baseadas em taxonomia para técnicas para detecção automática de fraudes, existem poucas referências. Entre as existentes, tem-se, por exemplo, a *Universal*

*Knowledge Grid* (UKG), que propõe uma arquitetura em *grid* (*grid computing* ou computação em grade é um modelo capaz de alcançar uma alta taxa de processamento ao dividir uma tarefa entre diversas máquinas), e que tem como um dos principais serviços, uma base de conhecimentos contendo uma taxonomia para mineração de dados. Baseada em ontologia, esta arquitetura se destina à construção em larga escala de sistemas de conhecimentos distribuídos (LI *et al.*, 2006).

Apesar de mineração de dados, ser uma das técnicas utilizadas para detecção automática de fraudes há outras, baseadas em inteligência artificial ou em estatística, cujas propriedades também serão observadas na ontologia sobre este domínio nesta dissertação. Também deve ser observado, que a ontologia sobre técnicas para detecção automática de fraudes será complementada pelas taxonomias dos elementos que compõem a abordagem para avaliar escolhas de técnicas junto a problemas de fraude específicos, o que a distingui de outras ontologias sobre domínio similares.

Em relação a problemas de fraude, existem algumas já desenvolvidas. Por exemplo, o projeto *Financial Fraud Prevention – Oriented Information Resources using Ontology Technology* (FFPOIROT) foi desenvolvido pela Comunidade Econômica Européia, com a finalidade de gerar ontologias sobre fraudes para domínios específicos, utilizando metodologias próprias para aquisição de conhecimento e para desenvolvimento de ontologias: *Application Knowledge Engineering Methodology* (AKEM) e *Developing Ontology Guided Methodology* (DOGMA), respectivamente (ZHAO; LEARY, 2005).

Porém, nesta dissertação está proposto uma ontologia para o domínio problemas de fraude, concentrada no processo de detecção de fraude, sendo abordados, primordialmente, os aspectos referentes aos requisitos do ambiente relacionados ao problema a ser analisado (volume, complexidade e completeza dos dados, por exemplo). São tratados, também pela dissertação, às dimensões de avaliação do processo de detecção (precisão da análise, facilidade e flexibilidade no uso de técnicas, por exemplo). Deste modo, a ontologia sobre problemas de fraude que comporá a base de conhecimentos da dissertação, torna-se específica, ao permitir o relacionamento entre o problema de fraude com o processo de detecção a ser utilizado.

Em resumo, os objetivos específicos da dissertação são:

a) Construir uma ontologia para técnicas para detecção automática de fraudes, fornecendo os conceitos, relações e restrições que definem a mesma;

b) Construir uma ontologia para problemas de fraudes, fornecendo os conceitos, relações e restrições que definem a mesma;

c) Efetuar o *merge* entre as duas ontologias anteriores, a fim de gerar uma base de conhecimentos capaz de responder aos questionamentos referentes *às quais técnicas para detecção automática de fraudes podem ser usadas para determinado domínio de problema*.

## 1.3 BENEFÍCIOS ESPERADOS

Com a implementação das ontologias, espera-se prover uma infraestrutura que permita suprir os conhecimentos necessários para que se possa identificar de modo efetivo, quais técnicas para detecção automática de fraudes possam ser encaminhadas para determinado domínio.

Sendo possível obter:

a) Resultados mais precisos, uma vez que, a escolha mais adequada de uma técnica para detecção automática de fraudes significa escolher aquela mais eficaz para um domínio específico, considerando o volume de dados e a complexidade do problema existente;

b) Maior garantia na redução de custos, pois a adequação do processo de escolha implica, também, numa maior eficiência do processo de detecção de fraudes a partir do uso de técnicas para detecção automática;

c) Por fim, num processo onde a escolha das técnicas se torna mais adequada e a geração dos resultados mais precisa, haverá o aumento, por conseguinte, do nível de conhecimentos dos especialistas envolvidos quanto ao aprimoramento do próprio processo de detecção de fraude para um domínio específico, além de permitir o compratilhamento destes conhecimentos.

## 1.4 ESTRUTURA DO TRABALHO

Esta dissertação possui cinco capítulos, sendo os mesmos:

O Capítulo 2 aborda os aspectos conceituais para a construção das ontologias, contendo informações referentes aos domínios para problemas de fraude e para técnicas para detecção automática de fraudes.

O Capítulo 3 apresenta a base teórica referente às ontologias, abordando aplicações, linguagens de representação e ferramentas de edição, bem como processos de apoio à construção das mesmas.

O Capítulo 4 concentra todas as informações sobre as três ontologias construídas, baseadas na metodologia de Uschold & King´s *method*.

No capítulo 5, tem-se a conclusão do trabalho, abordando suas contribuições e limitações, e oferecendo uma perspectiva de trabalhos que podem ser desenvolvidos futuramente para melhorar a abordagem proposta.

# CAPÍTULO 2

## 2 ASPECTOS CONCEITUAIS PARA AS ONTOLOGIAS

Em 1494, Luca Pacioli, pai do Método das Partidas Dobradas, o qual é utilizado até os dias atuais nos sistemas contábeis das organizações, em seu livro *Summa de Arithmetica, Geometria, Proportioni et Proportionalita*, declara "quem conduz seus negócios sem saber tudo sobre ele, vê seu dinheiro voar como moscas" (PACIOLI *apud* ALLEN, 1999). Nas sociedades do século XXI, em particular, os setores de negócios, ainda estão vendo seu dinheiro voar como moscas por não ter a capacidade para combater a ocorrência de fraudes (SILVERSTONE; DAVIA, 2005).

A fraude é tão antiga quanto à humanidade e pode tomar uma variedade ilimitada de formas (BOLTON; HAND, 2002).

De acordo com Thomas Hoving, ex-diretor do Metropolitan Museum of Art de Nova York (nos Estados Unidos), da pré-história até os dias atuais, quarenta por cento de todos os objetos de arte são falsos ou frutos de eventos fraudulentos. Segundo Hoving, foram identificados objetos de artes fenícios fraudulentos datando desde o sexto até o segundo milênio a.C., além de haver fortes indícios de que na Roma Antiga existiam verdadeiras fábricas de objetos de arte falsos (HOVING *apud* ALLEN, 1999).

No século XIX, por exemplo, tem-se o clássico exemplo da fraude ética cometida pelo renomado cientista francês Louis Pasteur, quando o mesmo omitiu o nome de um colega químico ao publicar estudos sobre uma vacina para a doença conhecida como antraz, causada pela bactéria *Bacillus Anthracis* (GEISON *apud* ALLEN, 1999).

Durante o século XX, os eventos fraudulentos se transformaram de pequenas traições, pequenos golpes monetários, charlatanices médicas, para atos mais sofisticados e complexos, sejam em dimensões sejam em tecnologia. Têm-se eventos ocorrendo nos domínios bancário, governamental, de telecomunicação, informática, cartão de crédito, seguro, entre outros (ALLEN, 1999).

### 2.1 CARACTERIZAÇÕES PARA FRAUDES

Há aspectos básicos que são suficientes para caracterizar uma ocorrência como fraudulenta (ACFE *apud* SILVERSTONE; DAVIA, 2005):

a) É clandestina;

b) Gera uma violação na relação de confiança entre o fraudador (*perpetrator* ou *fraudster*) e a vítima;

c) É cometida para beneficiar financeiramente, direta ou indiretamente, o fraudador;

d) Gera custos para as organizações com a perda de bens e recursos.

Com base no exposto, têm-se várias definições para o termo fraude. A *European Healthcare Fraud and Corruption Network* (EHFCN), dedicada ao combate a fraudes e a corrupção, define o termo como sendo: "o uso ou a apresentação de declarações ou de documentos falsos (incorretos, incompletos), a utilização dos mesmos para fins diferentes dos especificados, ou a não divulgação de informações que deveriam ser legalmente divulgadas, o que gera apropriações indevidas ou retenções ilegais de fundos e de propriedades de outros." (EHFCN *apud* ALEXOPOULOS *et al.*, 2006).

Por sua vez, o estatuto *Fraud and False Statements* na legislação federal dos Estados Unidos, traz a seguinte definição para fraude: "qualquer representação falsa de uma matéria de fato, produzida através de palavras ou de condutas, de alegações falsas ou inverídicas, ou do encobrimento daquilo que deveria ter sido divulgado, visando iludir ou com a pretensão de iludir o outro, gerando no mesmo alguma espécie de dano." (BAKER; RANGOS *apud* ALLEN, 1999).

Outra concepção, usualmente aceita, é a que define fraude como "o ganho intencional ou involuntário de qualquer coisa com valor, de uma pessoa, de pessoas, ou de organizações, através de uma pessoa, de pessoas, ou de organizações, usando meios enganosos, declarações falsas ou incompletas ou não compreensíveis, e omissões." (PHUA *et al.*, 2005).

Todas estas definições citadas anteriormente remetem a aspectos legais e introduzem cinco elementos fundamentais de um processo de fraudulento (ALLEN, 1999):

a) O fraudador (*perpetrator* ou *fraudster*);

b) A vítima ou alvo (*target*);

c) A fraude ou o problema de fraude (*fraud problem*);

d) A presença, a ausência, ou o grau de intenção (*intent*);

e) A coisa de valor envolvida na fraude (*thing*).

Com base nestes elementos é que têm sido construídas, em geral, as taxonomias para fraude, concentrando-se na especificação detalhada dos tipos identificados (ALLEN, 1999).

## 2.2 CICLO DE VIDA DA GESTÃO DE FRAUDES

A partir dos elementos que compõem um processo fraudulento (item 2.1), estabeleceu-se um esquema para gestão sobre a ocorrência de fraudes, chamada de Ciclo de Vida da Gestão de Fraudes ou *Fraud Management Lifecycle*, tendo o objetivo de reduzir as perdas e os custos sociais associados às mesmas (WILHELM, 2004).

O *Fraud Management Lifecycle* é um processo composto de oito estágios, montados sobre uma estrutura em rede, de forma não linear e não seqüencial, onde cada estágio opera de modo interdependente e inter-relacionado, sem seguir um ciclo tradicional (WILHELM, 2004).

Neste esquema, os oito estágios se assemelham a nós (dentro dos quais são desempenhadas atividades, operações e funções), sendo os mesmos (WILHELM, 2004):

1) Intimidação (*fraud deterrence*): caracteriza-se por ações e atividades de desencorajamento, a fim de tentar parar ou prevenir a fraude antes que ela aconteça;

2) Prevenção (*fraud prevention*): caracteriza-se por ações e atividades para prevenir a ocorrência de fraudes;

3) Detecção (*fraud detection*): caracteriza-se por ações e atividades, tal como, monitoramento estatístico para identificar e localizar atos fraudulentos, durante e depois de terem acontecidos;

4) Mitigação (*fraud mitigation*): caracteriza-se por ações e atividades cuja finalidade é conter a ocorrência de perdas decorrentes de atos fraudulentos, evitando que elas continuem a ocorrer ou de continuar ocorrendo;

5) Análise (*fraud analysis*): caracteriza-se pela identificação e pelo estudo das perdas ocorridas (apesar das atividades de intimidação, prevenção e detecção), buscando obter quais são os fatores determinantes;

6) Política para Fraudes (*fraud policy*): caracteriza-se por atividades de criação, avaliação, comunicação e apoio à implementação de ações para redução de incidentes de fraudes;

7) Investigação (*fraud investigation*): caracteriza-se por ações voltadas à obtenção de evidências suficientes para conter atividades fraudulentas, recuperar ou obter a restituição de recursos desviados, e levantar provas para subsidiar processos judiciais contra fraudadores;

8) Instauração de Processo (*fraud prosecution*): caracteriza-se pelo encaminhamento de processos judiciais, visando, principalmente, a punição dos envolvidos e a recuperação dos recursos desviados.

Dentre todos os estágios apresentados, o foco de atuação das organizações tem se concentrado sobre aqueles de prevenção e de detecção de fraudes, por serem considerados os críticos dentro de um processo de gestão (PARODI, 2005).

Sob este aspecto, propõe-se que, para uma efetiva estratégia de prevenção e detecção, sejam estabelecidas, com o máximo de clareza possível, definições sobre o emprego de recursos humanos, de infraestrutura e tecnológicos, além de serem observadas as seguintes questões (SILVERSTONE; DAVIA, 2005):

a) Entender porque a fraude é cometida;

b) Assegurar que fatores que podem motivar empregados no cometimento de fraudes sejam minimizados;

c) Entender quais são as oportunidades para fraudes no negócio;

d) Localizar as exposições e áreas de risco e reduzir as oportunidades de fraudes;

e) Conhecer os sintomas de fraude;

f) Comunicar o comportamento esperado dos empregados;

g) Responder, apropriadamente, à identificação dos problemas e à punição dos fraudadores.

Deste modo, é necessário o completo entendimento sobre o contexto de um domínio de fraude, para que o investimento realizado pelas organizações sobre um ambiente tecnológico para detecção seja o mais apropriado possível (ECONOMIST INTELLIGENCE UNIT AND KROLL, 2007).

## 2.3 TIPOLOGIAS DE FRAUDES

As fraudes podem ser divididas em três grupos: fraudes que foram detectadas e estão detalhadas para domínio público, fraudes que foram detectadas e não estão detalhadas para domínio público, e fraudes que não foram detectadas (SILVERSTONE; DAVIA, 2005).

Esta classificação é importante, pois traz como implicação que qualquer esforço feito para construir uma tipologia para fraudes, além de se tratar de uma perspectiva em particular, terá como escopo aproximado, um terço do universo pretendido (ACFE *apud* SILVERSTONE; DAVIA, 2005).

Na pesquisa para a dissertação, encontramos duas taxonomias baseadas nos cinco elementos fundamentais de um processo fraudulento, citados no item 2.1.

A primeira delas é uma taxonomia proposta por Allen (1999) que tem como raiz (*root*) o elemento vítima ou *target*, e possui a seguinte constituição (Figura 1):

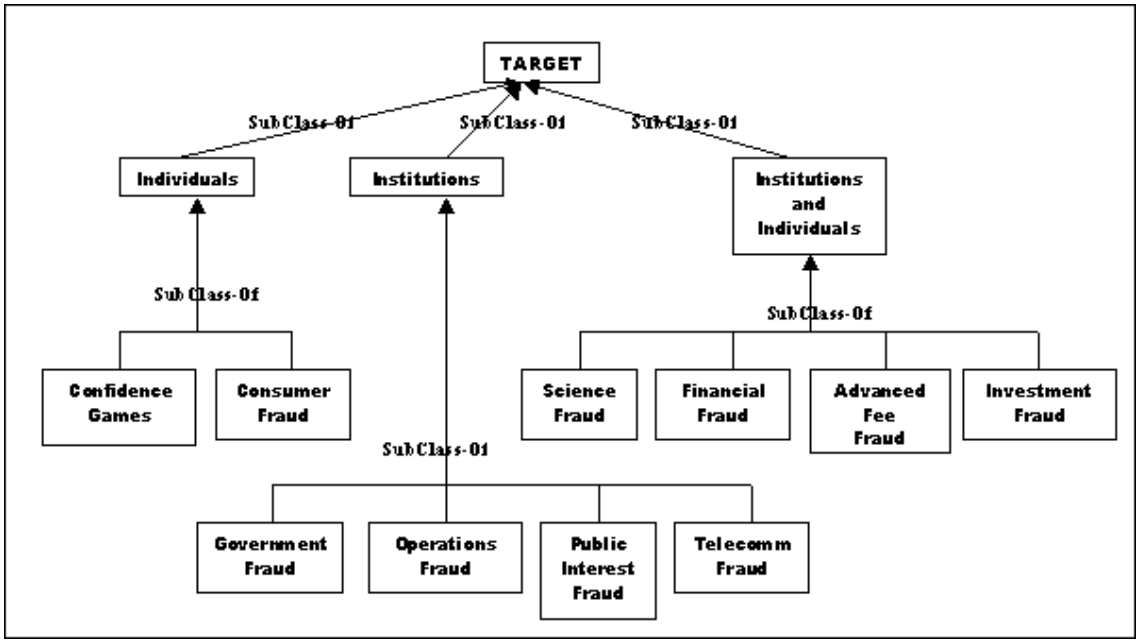


Figura 1 - Taxonomia de Fraude baseada no elemento *target*
Nota: elaborado por Allen (1999).

Nela, as classes *Individuals* e *Institutions*, correspondem a pessoas e a organizações respectivamente, e as subclasses têm as seguintes descrições (ALLEN, 1999):

a) Jogos de Confiança (*Confidence Games*): trata do estabelecimento de confiança pelo fraudador, a fim de subtrair algo de valor de uma vítima. Alguns domínios correlacionados são: falsificação de identidade (*impersonation* ou *identity fraud*), jogo ilegal (*gambling*), esquemas de fraudes envolvendo cultos religiosos onde os praticantes são vítimas de perdas financeiras (*cult fraud*);

b) Fraude ao Consumidor (*Consumer Fraud*): ocorre quando o fraudador comete um ato fraudulento durante a compra de um bem ou de um serviço. Entre os domínios relacionados, temos: fraude em cartão de crédito (*credit card fraud*), fraude na emissão de diplomas e certificados para a área de educação (*education fraud*), simulação fraudulenta de defeitos em veículos automotores (*motor vehicles fraud*);

c) Fraude Governamental (*Government Fraud*): ocorre quando o agente governamental ou é o fraudador ou é o alvo. Existem dois tipos de domínios para este problema de fraude: aqueles relacionados com os chamados programas de direito (*entitlement programs*), dentro dos quais temos fraudes

em educação envolvendo estudantes fantasmas e desvio de verbas (*education fraud*), fraudes em programas sociais e de assistência social (*welfare fraud*), fraudes em programas de saúde e de seguro saúde (*medicare and medicaid fraud*), fraudes na previdência social pública (*social security fraud*); e aqueles relacionados com os chamados programas de serviços (*service programs*), que engloba fraudes na área alfandegária (*customs fraud*), na área de defesa (*defense fraud)*, na área postal (*postal fraud*), entre outras;

d) Fraude em Operações (*Operations Fraud*): este tipo de fraude se dá no universo das operações de uma organização: fraude em operações interna, quando há o envolvimento de empregados e bens; fraude em operações externa, quando envolve licitação, contratação e competição. Neste contexto, alguns dos domínios que se aliam a este problema são: fraudes em processos de licitação (*bidding fraud*), fraudes através, por exemplo, do subfaturamento de contratos (*contract fraud*), e fraudes provocados através de um empregado, por exemplo, quando o mesmo falsifica credenciais (*employee or internal fraud*);

e) Fraude Pública (*Public Fraud, Public Interest Fraud*): ocorre quando o agente que presta algum tipo de serviço público (pertencente ao poder público ou não) ou é o fraudador ou é o alvo. Como exemplos de domínios relacionados, temos: desvios em fundos de caridade (*charity fraud*), desvios de valores devidos em taxas e impostos (*tax fraud*), fraudes no resultado de uma eleição (*election fraud*);

f) Fraude em Telecomunicações (*Telecommunication* ou *Telecomm Fraud*): problema de fraude, basicamente, concentrado em três domínios: fraudes em serviços de telefonia fixa (*hard-line telephone service fraud*), fraudes em serviços de telefonia móvel ou comunicação móvel (*wireless phone service*) e fraudes relacionadas com a *internet* (*internet service fraud*);

g) Fraude Científica (*Science Fraud*): problema de fraude onde o fraudador ou o alvo tem origem na área acadêmica. Em geral, assim como a *Telecomm Fraud*, este tipo também está concentrado em três domínios: fraudes relacionadas com a geração de requisições, relatórios e conteúdos falsos (*criminal science fraud*); geração de perdas financeiras para uma instituição

(governo ou não) a partir de ações de negligência, participação em eventos fictícios ou de pouca significância, e não cumprimento de contratos e prazos (*civil science fraud*); e falta de conduta experimental (projeto experimental intencionalmente falho, uso de experimentos sem conclusão, etc), de conduta para publicação (falsificação de dados e de métodos, informação sobre experimentos fantasmas, etc.), falsificação de currículo vitae, entre outros (*Ethical Science Fraud*);

h) Fraude Financeira (*Financial Fraud*): fraude concentrada nas áreas bancária, financeira e de seguro. Este tipo tem como domínios relacionados: fraudes envolvendo operações e empréstimos bancários (*Banking and Lending Fraud*), fraudes envolvendo operações financeiras não bancárias (exceto ativos) (*Financial Statement Fraud*), fraudes referentes ao mercado de seguro privado (*Insurance Fraud*);

i) Fraude em Adiantamento de Taxa (*Advanced Fee Fraud*): é descrito como uma fraude onde o fraudador convence a vítima a fazer um pagamento adiantado por uma entrega futura de um bem ou de um serviço, que ou não será entregue ou não será entregue nas condições contratadas. Não há domínios específicos para este tipo de problema de fraude;

j) Fraude em Investimentos (*Investment Fraud*): normalmente ocorre em três situações: o fraudador vende um bem que não possui ou que não tem a posse, o fraudador deturpa as características, o valor ou retorno potencial de um recurso a ser negociado, ou o fraudador gerencia mal um recurso para privar o seu proprietário de usufruir do seu potencial de uso ou de investimento. Assim, alguns domínios para este problema de fraude são: fraudes envolvendo objetos de arte (*Art Fraud*), fraudes relacionadas com o mercado de gemas, pedras ou minerais (*Gem and Mineral Fraud*), fraudes em operações com papéis de riscos referentes a débitos de organizações privadas e de governos (*Bond Fraud*).

A outra taxonomia identificada durante as pesquisas para esta dissertação possui como raiz o elemento fraudador, tendo a seguinte constituição (PHUA *et al.*, 2005):

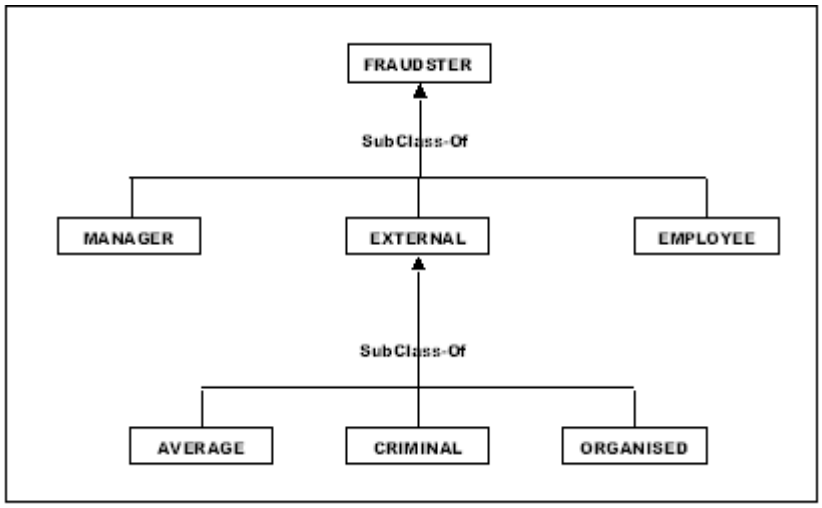


Figura 2 - Taxonomia de Fraude baseada no elemento fraudador
Nota: elaborado por Phua *et al.* (2005).

Esta taxonomia se concentra no ambiente corporativo, buscando estabelecer quais tipos de fraudadores são mais identificáveis pelas ações de detecção de fraudes (PHUA *et al.*, 2005):

a) Gerência (*Manager*): o fraudador é ocupante de uma função de gestão dentro da organização. Não há associação a um domínio específico.

b) Empregado (*Employee*): o fraudador é um empregado dentro da organização. Não há associação a um domínio específico.

c) Externo (*External*): o fraudador é alguém fora dos quadros da organização.

c.1) *Average*: o fraudador tem um comportamento, ocasionalmente, desonesto, isto é, ele pratica a fraude quando surge uma oportunidade, quando tem uma tentação repentina ou quando sofre dificuldades financeiras. Um exemplo de domínio relacionado a um problema de fraude provocado por este tipo de fraudador seria uma fraude em seguros (*Insurance Fraud*). Neste caso, os sistemas de detecção, normalmente, utilizam abordagens mais padronizadas, considerando que o *modus operandi* dos responsáveis pelas fraudes são menos variados;

c.2) *Criminal* e *Organised*: o fraudador atua de forma individual (*Criminal*) ou em grupo (*Organised*), de modo sistemático e em tempo integral. Como exemplos domínios de problemas de fraude praticados por estes tipos de fraudador, têm-se, fraudes em operações de crédito (*Credit Fraud*) e fraudes

em telecomunicações (*Telecomm Fraud*). Neste caso, os sistemas de detecção automática utilizam abordagens mais sofisticadas, considerando que o *modus operandi* dos fraudadores envolve o desenvolvimento de padrões próximos àqueles considerados legais, utilizando processos de dissimulação de identificação, além de possuírem algoritmos que produzem ações de contraponto aos próprios processos de identificação dos sistemas de detecção automática de fraudes das organizações atingidas.

A primeira, das duas taxonomias apresentadas anteriormente, aquela baseada no elemento vítima, foi escolhida para apoiar esta dissertação, por permitir a construção de uma ontologia onde seria possível associar a tipologia de fraude a uma estrutura taxonômica contendo conceitos relacionados à caracterização do ambiente, no qual serão aplicadas as técnicas para detecção automática de fraudes.

## 2.4 TÉCNICAS PARA DETECÇÃO AUTOMÁTICA DE FRAUDES

Com o aumento das relações nas sociedades modernas, a necessidade de se identificar ocorrências de fraudes de modo mais rápido e com maior nível de certeza, tem sido considerado algo de fundamental importância, seja pelo volume seja pela complexidade dos dados envolvidos (SILVERSTONE; DAVIA, 2005). Assim, o emprego de tecnologias que envolvam não somente o processo de detecção, mas também a detecção automática de fraudes, como por exemplo, ferramentas de mineração de dados e de estatística, têm se tornando cada vez mais comum (FAWCETT; PROVOST, 1997).

Particularmente, em relação ao universo de dados (volume e complexidade), as especificidades dos domínios envolvidos determinam características únicas, que em última instância, deveriam refletir na escolha da técnica ou das técnicas para detecção automática a serem utilizadas.

Portanto, tomando-se quatro importantes domínios de problemas de fraudes e considerando o padrão dos dados quanto ao fator volume em cada um dos domínios (ABBOTT *et al.*, 1998; PHUA *et al.*, 2005; BOLTON; HAND, 2002; PHUA, 2003):

a) Fraudes corporativas (*Internal Fraud*):

a.1) Nível Gerencial (*Manager*): conjuntos pequenos de dados, em geral, com menos de quinhentas ocorrências;

a.2) Nível Empregados (*Employee*): conjuntos médios a grandes de dados, em geral, podendo chegar a mais de cinco milhões de ocorrências;

b) Fraudes em Seguros (*Insurance Fraud*): conjuntos médios de dados, em geral, com volumes em torno de quarenta a cinqüenta mil ocorrências;

c) Fraudes em cartões de crédito (*Credit Card Fraud*): conjuntos grandes de dados, em geral, com mais de doze milhões de ocorrências/ano;

d) Fraudes em telecomunicações (*Telecomm Fraud*): conjuntos imensos de dados, em geral, com mais de cem milhões de ocorrências.

Concluiríamos que será de pouca efetividade a aplicação de uma técnica para detecção automática, com baixo desempenho sobre grandes volumes de dados, caso os mesmos sejam referentes a um problema de fraude no domínio de telecomunicações, por exemplo.

Além do volume e complexidade dos dados, outro aspecto, por exemplo, a ser observado durante a escolha da técnica para detecção automática em relação ao domínio de fraude, seria aquele ligado ao nível de perícia dos analistas envolvidos (responsáveis pela detecção). Uma eventual dificuldade dos analistas em relação à técnica empregada pode determinar a não revelação de informações fundamentais sobre as amostras analisadas, tais como, atributos importantes, relacionamentos prováveis e padrões não conhecidos (PHUA, 2003).

Assim, a escolha de técnicas para detecção automática de fraudes, que sejam de difícil aprendizagem ou que demandem a contratação de especialistas, pode inviabilizar um processo de detecção (WILLIAM *apud* PHUA *et al.*, 2005).

Conforme descrito no item 2.2, o estágio de detecção de fraudes ou *fraud detection* se caracteriza, principalmente, pela identificação e localização de fraudes o mais rápido possível, a partir do momento em que as mesmas estejam sendo ou tenham sido praticadas (BOLTON; HAND, 2002). Ele se desenvolve por meio de ações e atividades, tais como, aqueles de natureza dissuasória, onde a detecção efetuada ocorre através de sondagens e testes sobre o ambiente que está sendo fraudado (WILHELM, 2004).

O estágio de *fraud detection* atua basicamente sobre três perspectivas (WILHELM, 2004): teste de fraudes (*fraud testing*), tentativas de fraudes (*fraud attempts*) e fraudes bem sucedidas (*fraud successful*). Esta separação é determinada pelo fato de que, nem todas as tentativas de

fraudes serem bem sucedidas, e nem todas as tentativas identificadas terem a intenção de ser bem sucedidas, podendo ser apenas ações para detecção de vulnerabilidades visando atos futuros (WILHELM, 2004).

Assim, apenas a identificação clara dos componentes indicativos de testes, de tentativas e de sucesso referentes a uma fraude, indicará se os métodos e as técnicas empregadas para a *fraud detection* possuem a eficiência esperada (WILHELM, 2004; ABBOTT *et al.*, 1998).

Para tentar garantir ao máximo estas eficiência e eficácia, é que cada vez mais se tem procurado reduzir as atividades manuais referentes aos procedimentos de detecção, optando-se pelo investimento em tecnologias que tragam um maior nível de automação (PHUA *et al.*, 2005). De fato, como é impossível estar absolutamente certo sobre a legalidade da intenção de uma aplicação ou de uma transação, procura-se contornar este problema, utilizando-se algoritmos matemáticos implementados através de técnicas para detecção automática de fraudes, a fim de que se tenha maior rapidez no tratamento desta realidade, com menor custo e com maior precisão (PHUA *et al.*, 2005).

A partir de pesquisas em diversos países, vêm sendo desenvolvidos *engines* analíticos baseados em lógica *fuzzy*, algoritmo genético, máquinas de aprendizagem, redes neurais, entre outros, aplicados a soluções especializadas para detecção automática de fraudes em diversas áreas de negócio, como, cartão de crédito, comércio eletrônico, seguro, varejo, telecomunicação, etc. (PHUA *et al.*, 2005; DHAR; STEIN, 1997; ABBOTT *et al.*, 1998).

Geralmente, as técnicas para detecção automática de fraudes possuem diferentes objetivos e características. Isto significa que estas, para serem eficientes e eficazes, devem ser aplicadas às classes de problemas cujas características sejam as mais próximas possíveis dos objetivos e das características das técnicas, para que as mesmas possam oferecer os seus melhores desempenhos (DHAR, STEIN, 1997).

As técnicas, em geral, buscam responder a três perguntas básicas (DHAR; STEIN, 1997):

a) *Como podem ser aplicadas*?

b) *Qqual o universo a ser trabalhado*?

c) *O que se espera com o uso das mesmas*?

Para responder a tais perguntas, deve-se explorar pelo menos uma das seguintes questões:

a) *Qual o método que se pretende utilizar para efetuar a comparação entre os dados observados de uma amostra a ser analisada e os respectivos valores esperados*?

b) *Quais os status para as dimensões de avaliação e para os fatores ambientais que causam impactos sobre o uso de técnicas para detecção automática junto a um problema de fraude específico*?

Nesta dissertação nos concentraremos sobre a segunda questão, a qual se refere à abordagem a ser utilizada, visando avaliar o impacto da escolha de uma determinada técnica para detecção junto a um problema de fraude específico.

### 2.4.1 Abordagens para avaliação

Apesar das pesquisas atuais, o uso de técnicas para detecção automática de fraudes ainda é muito concentrado em poucos tipos. Este fato é uma das principais dificuldades para se ter uma avaliação mais efetiva sobre as mesmas, principalmente, se for considerada a dinâmica do universo de fraudes, onde novos domínios são estabelecidos e novos tipos de ocorrências são identificados todo o tempo (PHUA *et al.*, 2005).

Para se iniciar uma avaliação de técnicas a serem empregadas, é necessário que aspectos ambientais do contexto de um problema de fraude sejam observados (FAWCETT; PROVOST, 1997).

Duas abordagens, neste sentido, são conhecidas. A primeira foi proposta por Phua (2003) e possui cinco dimensões:

a) Interpretabilidade (*Interpretability*): refere-se ao nível de conhecimento necessário para que um especialista de domínio ou uma pessoa não-técnica possa entender os modelos de predições através de visualização ou de regras;

b) Efetividade (*Effectiveness*): refere-se à predição total e exata, e o desempenho de cada técnica;

c) Robustez (*Robustness*): refere-se à habilidade de fazer predições corretas a partir de dados ruidosos (com sujeiras ou inconsistentes) e com valores perdidos;

d) Escalabilidade (*Scalability*): refere-se à capacidade de construir um modelo eficiente para análise, capaz de atuar sobre amostras de dados de vários tamanhos;

e) Velocidade (*Speed*): refere-se o quanto uma técnica é eficaz em termos de rapidez para a obtenção de padrões, a partir de um dado modelo.

A segunda abordagem tem como fonte uma proposição conceitual, chamada de *Intelligence Density*, estabelecida por Dhar e Stein (1997), com a finalidade de medir a produtividade e a inteligência de uma organização.

Segundo os formuladores da proposição, *Intelligence Density* seria o equivalente pós-industrial (era da informação) ao conceito de produtividade no período industrial. Assim, como uma organização possui alta produtividade, caso produza em grande quantidade no menor tempo e com os menores custos possíveis, pode-se dizer que a mesma tem alta *Intelligence Density*, se os seus sistemas de informações forem capazes de dar, aos tomadores de decisão, os conhecimentos necessários para as decisões nos menores tempos possíveis (DHAR; STEIN, 1997).

Por exemplo, se um tomador de decisão decide examinar duas fontes de informações, com as mesmas qualidades e com as mesmas conclusões, fonte A por 3 minutos e fonte B por 30 minutos, pode-se concluir que a fonte de informação A tem 10 vezes mais *Intelligence Density* do que a fonte de informação B (DHAR; STEIN, 1997).

Para obter esta medida, os autores propõem que se combinem certas dimensões (dimensões de avaliação) utilizadas para avaliar as fontes de informações, com um conjunto de fatores ambientais capazes de intervir no processo de geração de informações.

Deste modo, de acordo com o conceito proposto, teríamos as seguintes dimensões para avaliação de uma fonte de informação e os seguintes fatores ambientais, que uma vez combinados, gerariam uma medida para a produtividade de uma determinada fonte (*Intelligence Density*) em relação a um processo decisório (DHAR; STEIN, 1997):

I. Dimensões (*Dimension*):

a) Engenharia (*Engineering*)

– Facilidade de uso (*Ease of Use*): refere-se ao nível de complexidade de uma técnica, para que a mesma possa ser usada diariamente pelos especialistas em negócio;

– Encapsulamento (*Embeddability*): refere-se ao nível de facilidade para que uma técnica possa ser incorporada à infraestrutura de uma organização;

– Compacidade (*Compactness*): refere-se a implementação em código (quantidade de linhas e complexidade lógica) de uma técnica;

– Escalabilidade (*Scalability*): refere-se à capacidade de uma determinada técnica, em suportar a adição de mais variáveis ao problema ou o incremento de faixa de valores que as variáveis possam vir a assumir;

– Flexibilidade (*Flexibility*): refere-se ao nível de facilidade que uma técnica possa oferecer, para que as relações entre variáveis ou entre variáveis e os respectivos domínios possam ser alteradas, ou os objetivos estabelecidos possam ser modificados;

b) Logística (*Logistical*)

– Facilidade Computacional (*Computing Ease*): refere-se ao grau de implementação para uma técnica, sem haver necessidade de se requerer *hardware* e *software* especializados;

– Tempo de Desenvolvimento (*Development Time*): refere-se ao tempo que uma organização gastaria para desenvolver uma solução baseada numa técnica específica;

– Independência de Especialistas (*Independence from Experts*): refere-se ao grau de independência para projetar, construir e testar uma solução baseada numa técnica, sem uso de especialistas na mesma;

c) Qualidade (*Quality*):

- Precisão (*Accuracy*): refere-se ao grau de precisão oferecido por uma técnica para as saídas geradas, em termos de correção ou de melhor decisão ou de opção;
- Clareza na Explicação ou Explicabilidade (*Explainability*): refere-se ao nível de descrição que uma técnica oferece, para o modo como uma conclusão foi alcançada;
- Tempo de Resposta (*Response Time*): refere-se ao tempo gasto por uma técnica para completar uma análise num nível desejado de precisão;

d) Recursos (*Resource*):

- Curva de Aprendizagem (*Learning Curve*): refere-se ao grau que uma organização deve alcançar para se tornar suficientemente competente na solução de problemas, a partir do uso de uma determinada técnica;
- Tolerância a Dados Incompletos (*Tolerance for Sparse Data*): refere-se ao grau de comprometimento no nível de qualidade oferecido por uma técnica, devido a falta de dados ou pela existência de dados incompletos;
- Tolerância a Dados Inconsistentes (*Tolerance for Noise in Data*): refere-se ao grau de comprometimento no nível de qualidade oferecido por uma técnica, notadamente no que se refere à precisão, devido à inconsistência no conteúdo dos dados;

e) Tolerância a Complexidade (Tolerance for Complexity): refere-se ao grau de comprometimento no nível de qualidade oferecido por uma técnica, devido ao número de interações entre os vários componentes do problema modelado (por exemplo, muitas interações não-lineares entre variáveis) ou a complexidade do conhecimento para modelar um problema.

II. Fatores Ambientais (*Environment*):

a) Quantidade de dados (*Amount of Data*): relacionado com a quantidade de bytes da amostra de dados presente na fonte de informação;

b) Entendimento do negócio (*Business Understand*): relacionado com o grau de complexidade do domínio ao qual se relaciona a amostra de dados presente na fonte de informação;

c) Complexidade do problema de negócio (*Complexity of Business Problem*): relacionado com o grau de complexidade do problema referente ao domínio sob o qual se encontra a amostra de dados envolvida na fonte de informação;

d) Complexidade dos dados (*Complexity of Data*): relacionado com o grau de complexidade da amostra de dados envolvida na fonte de informação;

e) Complexidade da infraestrutura (*Complexity of Infrastructure*): relacionado com o grau de complexidade da infraestrutura requerida para o processamento da amostra de dados presente na fonte de informação;

f) Entendimento dos dados (*Data Understand*): relacionado com o grau de entendimento requerido para a amostra de dados presente na fonte de informação;

g) Infraestrutura disponível (*Infrastructure Available*): relacionado com o grau de complexidade da infraestrutura disponível para o processamento da amostra de dados presente na fonte de informação;

h) Dados limpos (*Scrubbed Data*): relacionado com o grau de eficiência da representação e do conteúdo da amostra de dados (uso de matrizes esparsas, redução de redundância, redução de inconsistências) presente na fonte de informação;

i) Dados atualizados (*Update of Data*): relacionado com o nível de atualização da amostra de dados envolvida na fonte de informação.

Na montagem da base de conhecimentos para esta dissertação, foi usada a segunda abordagem, baseada no conceito de *Intelligence Density*. Desta forma, foram incorporadas às taxonomias apresentadas os conceitos referentes às dimensões de avaliação e aos fatores ambientais.

## 2.5 TÉCNICAS PARA DETECÇÃO AUTOMÁTICA DE FRAUDES

Uma das dificuldades para se implementar um processo de *fraud detection* se deve normalmente, a existência de muitos registros legítimos para cada fraudulento. Uma técnica para detecção capaz de identificar com noventa e nove por cento de acerto, quais os registros legítimos como legítimos e quais aqueles que são realmente fraudulentos, pode ser considerada altamente eficaz (BOLTON; HAND, 2002).

Entretanto, uma busca por transações ou ocorrências fraudulentas com o máximo grau de certeza exige, por exemplo, algoritmos rápidos e eficientes, associados ao ambiente de processamento nas condições mais favoráveis possíveis (BOLTON; HAND, 2002).

Levantamentos indicam, em média, o seguinte valor potencial correspondente a fraudes detectadas: se de cem milhões transações, zero vírgula um por cento são fraudulentas, caso uma organização tenha perdas de dez dólares para cada ocorrência ou transação fraudulenta, esta organização irá contabilizar um prejuízo de um milhão de dólares (FEDERAL BUREAU OF INVESTIGATION – FBI, 2006).

Isto conduz a uma constatação: uma fraude pode ser reduzida ao nível que se queira desde que sejam observados os esforços e custos a serem empregados (BOLTON; HAND, 2002). Na prática, acordos são firmados para estabelecer níveis desejados para os custos de detecção e para os de esforços, o que neste caso envolve, qual ou quais as técnicas para detecção de fraude deverão ser utilizadas, sobre quais fatores ambientais, e qual os valores a serem atribuídos às respectivas dimensões de avaliação (alto ou médio ou baixo, por exemplo) (BOLTON; HAND, 2002; ABBOTT *et al.*, 1998).

Com base em *surveys* e em outros trabalhos analisados, a partir do documento “Técnicas para Detecção Automática de Fraudes: Uma Revisão Sistemática” (Anexo I) e da literatura especializada (Anexo III) utilizados nesta dissertação, identificamos sete técnicas com alta freqüência de uso, cujas descrições seguem abaixo (ABBOTT *et al.*, 1998; PHUA *et al.*, 2005; BOLTON; HAND, 2002; PHUA, 2003; DHAR; STEIN, 1997; GROTH, 2000; KIMBALL, 1996; GOLDSCHMIDT; PASSOS, 2005; BARBIERI, 2001; YUFENG *et al.*, 2004):

a) Suporte à Decisão Orientada a Dados (*Data Driven Decision Support* ou DDDS): este é o designativo genérico para englobar as tecnologias de *Data*

*Warehousing* e de aplicações OLAP. Com a tecnologia de *Data Warehousing* é possível armazenar em um único local, séries de dados oriundas de fontes distintas, integrá-las, e então manipulá-las através do uso de aplicações baseadas na tecnologia OLAP, com o objetivo de se obter conhecimentos para tomada de decisões. Geralmente, esta técnica tem sua aplicação voltada para a tarefa de análise de dados no contexto do estágio de detecção de fraude;

b) Raciocínio Baseado em Casos (*Case-Based Reasoning* ou CBR): é uma técnica que se aproveita dos conhecimentos obtidos previamente, para resolver um determinado problema. Cada tentativa de solução passada é armazenada como um registro, chamado de caso, formando uma base de casos (*case base*). A base de casos se torna então um modelo. Para solucionar um problema, a técnica prevê que antes deva ser procurada uma base de casos o um modelo, com atributos similares, sendo que a solução gerada é uma síntese composta daqueles casos anteriores, devidamente ajustados, e dos novos. Com isto, o crescimento da base de casos implica num aumento da precisão da aplicação desta técnica. Em relação à detecção de fraudes, as tarefas com maior incidência de uso de CBR são aquelas voltadas para análise de dados e para monitoramentos de possíveis comportamentos fraudulentos;

c) Lógica Fuzzy (*Fuzzy Logic* ou FL): é uma técnica que simula um método de raciocínio, permitindo as descrições de regras. O poder desta técnica vem da habilidade que ela demonstra em descrever um fenômeno ou um processo particular, linguisticamente, representando-o através de um pequeno número de regras muito flexíveis, incluídas numa base de conhecimentos. O conceito *fuzzy* pode aplicado a uma situação vaga, difusa, onde não podemos responder simplesmente "Sim" ou "Não"; deste modo, Lógica Fuzzy é aquela que admite valores lógicos intermediários entre a falsidade e a verdade, como o talvez (ZADEH, 2002). *Fuzzy Logic* é constantemente usada na tarefa de efetuar monitoramentos de comportamentos a fim de detectar fraudes;

d) Algoritmo Genético (*Genetic Algorithm* ou GA): é uma técnica que envolve o uso de três componentes: um conjunto de variáveis, um conjunto de restrições (*constraints*) e um conjunto de objetivos. As variáveis são utilizadas

para descrever os vários aspectos do problema, as *constraints* são utilizadas para restringir os valores permitidos para cada uma das variáveis de um problema, e os objetivos definem os resultados esperados considerando o universo de dados disponíveis. A técnica de Algoritmo Genético tem sua aplicação no estágio de detecção de fraudes, principalmente, no que se refere às tarefas de geração de regras e de conhecimentos sobre determinado do domínio;

e) Sistema Baseado em Regras (*Rule-Based System* ou RBS): esta técnica cresceu em aplicabilidade fora do campo dos teoremas lógicos, como uma maneira de se estabelecer o que é verdadeiro ou falso em relação a uma determinada afirmação. Tipicamente, uma RBS armazena fatos referentes à solução heurística de problemas, em uma base especial chamada de base de regras (*rule base*). Os fatos são armazenados sob a forma de regras do tipo *IF-THEN*, e os mesmos são necessários para resolver problemas quando são confrontados com os dados. Uma RBS, na detecção automática de fraudes, é empregada em tarefas, tais como, analisar dados e efetuar monitoramentos de possíveis comportamentos fraudulentos;

f) Redes Neurais (*Neural Networks* ou NN): esta técnica é utilizada para gerar modelos (generalizações) a partir de um conjunto de dados, podendo estar os mesmos incompletos ou com problemas de consistência (ruidosos). A técnica procura “aprender” (determinar padrões) diretamente da massa de dados, através de exames repetidos dos mesmos, da busca de relacionamento entre eles, e da construção automática de modelos. Esta técnica tem seu uso voltado, prioritariamente, nas tarefas de detecção de *clusters*, geração de classificações e indução de regras, a fim de se obter padrões para fraude;

g) Aprendizagem de Máquina/Algoritmo de Particionamento Recursivo (*Machine Learning/Recursive Partitioning Algorithm* ou ML): a técnica de *Machine Learning* cria regras e árvores de regras, através do uso de pesquisa heurística sobre um conjunto de dados, a fim de obter relacionamentos e padrões estatísticos. Com isto, ela procura definir *cluster* de registros em categorias específicas. Dentre os algoritmos baseados nesta técnica, um dos mais recorrentes no universo de detecção de fraudes, é o de Particionamento

Recursivo ou *Recursive Partitioning Algorithm*, o qual “aprende” (determinar padrões) a partir dos dados, como na técnica de *Neural Networks*, mas também tenta encontrar relacionamentos a serem explicitados em regras, como na técnica RBS. No estágio de detecção de fraudes, esta técnica é utilizada, por exemplo, na tarefa de efetuar monitoramentos de possíveis comportamentos fraudulentos (através da construção de *clusters*).

## 2.6 CONCLUSÃO

Os aspectos conceituais para construção da base de conhecimentos nesta dissertação envolvem definições sobre o que seja um problema de fraude, suas características e tipos, seu ciclo de vida, e quais as técnicas envolvidas no estágio de detecção. Em relação a este, a atenção é concentrada sobre o emprego de técnica para detecção automática para fraudes, considerando a realidade atual, onde a complexidade e o volume de dados envolvidos nestas operações, em geral, são significativos.

Conforme comentado, a uma necessidade de se observar o domínio de fraudes, qual o contexto empregado para detecção das mesmas, a fim de que se possa empregar a técnica mais adequada. Assim, ter clareza na compreensão dos domínios e no contexto de análise é fundamental no processo de escolha da técnica, sempre observando, “*como podem ser aplicadas*?”, “*como elas trabalham*?”, “*o que se espera com o uso das mesmas*?”.

As dimensões de avaliação estabelecidas pelos domínios da fraude e os fatores ambientais associados às mesmas, devem compor a base de conhecimentos que deverá determinar a relação entre os problemas e as técnicas para detecção automática de fraudes. Assim, as taxonomias de problemas de fraudes e de técnicas para detecção automática, serão associadas àquelas referentes às dimensões de avaliação de fonte de informações e aos fatores ambientais.

# CAPÍTULO 3

## 3 ONTOLOGIAS

Historicamente, o termo ontologia se originou na Idade Antiga, por meio da filosofia e da metafísica, sendo definido como a “Ciência do Ser” (onto = ser, logia = ciência), a fim de responder perguntas do tipo *o que é o ser*? *O que caracteriza o ser*? *Quais estruturas são usadas por nossa mente para capturar a realidade*? Uma ontologia, para filósofos e metafísicos, seria expressa por intermédio do estabelecimento de um sistema de categorias (substância, qualidade, quantidade, relação, ação, paixão, lugar e tempo) para classificar qualquer coisa que pudesse ser dita (predicado) sobre qualquer coisa no mundo (sujeito) (ARISTOTELES *apud* CORCHO *et al.*, 2006).

Emmanuel Kant, na Idade Moderna, organizou um esquema ontológico em quatro classes, cada uma com sua tríade (quantidade – unidade/pluralidade/totalidade, qualidade – realidade/negação/limitação, relação – herança/casualidade/comunidade, e modalidade – possibilidade/existência/necessidade) para categorizar a essência das coisas e como elas são percebidas e entendidas (KANT *apud* CORCHO *et al.*, 2006).

A comunidade da ciência da Computação, por sua vez, vem usando o termo ontologia dentro do contexto de uma informação compartilhada referente a descrições formais de domínios (LACY, 2005), sendo domínio definido como um assunto específico de uma área (área do conhecimento) que é o foco de interesse de uma comunidade em particular (GRUBER, 1993b).

A partir deste conceito, podemos intuir que desenvolver uma ontologia pode ter similaridades com o processo de desenvolvimento de *software*, com seus conjuntos e suas estruturas de dados, tendo como propósito disponibilizar informações para serem usadas por programas. Estas informações que são compartilhadas e formalmente descritas podem ser, por exemplo, métodos para solução de problemas (PSM), aplicações independentes de domínio, agentes de software que usam ontologias e bases de conhecimentos que usam ontologias como dados (NOY; MCGUINESS, 2001).

Existem algumas razões para que ontologias sejam desenvolvidas, dentre elas temos (NOY; MCGUINESS, 2001):

a) Compartilhar entendimento comum de uma estrutura de informação entre pessoas e agentes de software;

b) Possibilitar o re-uso de conhecimentos de um domínio;

c) Efetuar suposições explícitas de um domínio;

d) Separar conhecimento de domínio de conhecimento operacional;

e) Analisar o conhecimento de um domínio.

Das razões apresentadas acima, algumas foram mais determinantes para a escolha desta tecnologia como elemento base desta dissertação.

A primeira delas foi *possibilitar o reuso de conhecimentos de um domínio*.

O projeto de dissertação trabalha com o desenvolvimento de duas ontologias, técnicas para detecção automática de fraudes e problemas de fraudes, e o *merging* entre elas.

Em relação a problemas de fraude, identificamos apenas duas ontologias, sendo uma proposta pelo projeto FFPOIROT (ZHAO; LEARY, 2005) e outra desenvolvida pela EHFCN, chamada de *Generic Fraud Ontology in e-Government* (ALEXOPOULOS *et al.*, 2006), as quais não tratam de aspectos de detecção de fraudes. Por isso, foi necessário o desenvolvimento de uma ontologia de problemas de fraudes, baseada na taxonomia proposta por Allen (1999), a qual tem como elemento-raiz a vítima (*target*) da fraude, complementada com conhecimentos relacionados com aspectos de detecção de fraudes.

Para técnicas para detecção automática de fraudes, identificamos apenas uma ontologia, conhecida como UKG, que incorpora taxonomia para conceitos de mineração de dados (LI *et al.*, 2006), e que é voltada para o desenvolvimento de sistemas para descoberta de conhecimentos sobre *grid*. Apesar de ser uma possível referência para o uso de conceitos da tecnologia de mineração de dados, optamos por uma maior concentração nos aspectos relativos à definição de uso para uma tecnologia, em vez de tratar um maior aprofundamento na sua descrição. Deste modo, mantêm-se a necessidade de se implementar completamente,

uma ontologia para técnicas para detecção automática de fraudes baseada na proposta conceitual apresentada por Dhar e Stein (1997) para *Intelligence Density* (ver seção 2.4.1).

Como conclusão, a dissertação disponibiliza duas novas bases de conhecimentos, que poderão ser reusadas de modo conjugado ou não.

Outra razão para a escolha o uso de ontologias nesta dissertação, se refere ao processo de *analisar o conhecimento de um domínio*, o que significa identificar, descrever e relacionar os termos que irão compor uma base de conhecimentos. A análise formal de termos é extremamente valiosa quando se deseja reusar ontologias existentes e estendê-las (NOY; MCGUINESS, 2001). Deste modo, quando optamos por usar esta tecnologia, imaginamos disponibilizar algo que possa ser entendido, manipulado e estendido a fim de gerar um novo conhecimento.

## 3.1 CONCEITOS DE ONTOLOGIA NO AMBIENTE COMPUTACIONAL

Segundo Uschold e Gruninger, no contexto da ciência da Computação, ontologia é um termo usado para fazer referências ao entendimento compartilhado de algum domínio de interesse, por meio da classificação de um mundo vinculado ao mesmo, sendo este mundo concebido como um conjunto de conceitos (entidades, classes, propriedades, atributos, processos) e de inter-relações entre estes conceitos (USCHOLD; GRUNINGER, 1996).

Um conceito largamente aceito em Computação é o que se segue, “uma ontologia é uma especificação formal de uma conceitualização” (GRUBER, 1993a).

Segundo Gruber, “especificação formal” diz respeito àquela descrita formalmente, o que no universo da Computação pode ser entendida como uma coleção de termos e seus relacionamentos, expressa em uma linguagem legível para máquinas (computadores) e contida em um arquivo de documentos (GRUBER, 1993a).

Quanto a “uma conceitualização”, deve-se entender como algo próximo a um modelo abstrato, numa visão simplificada do mundo que se pretende representar (GRUBER, 1993a).

Se fizermos um paralelo entre os conceitos de ontologia propostos pelos filósofos (item 3), e àqueles trazidos pela ciência da Computação, considerando a premissa através da qual a realidade computacional só é percebida se for estruturada tanto para pessoas como para

máquinas (computadores), podemos inferir pelo menos três diferenças importantes entre as definições apresentadas (CORCHO *et al.*, 2006):

1) Para a ciência da Computação, uma ontologia só é uma ontologia se ela for estruturada, permitindo que tanto pessoas como computadores possam "raciocinar" sobre a mesma;

2) Se um computador não "entender" uma ontologia, ela não pode ser considerada como tal, pois a mesma tem que ser estruturada e codificada em linguagem interpretável de máquina (STUDER *et al.*, 1998; GÓMEZ-PÉREZ *et al.* 2003 *apud* CORCHO *et al.*, 2006);

3) Para a ciência da Computação, a essência do termo ontologia reside nos aspectos de reusabilidade e compartilhamento, enquanto que, nas definições propostas pelos filósofos, estes aspectos estão ausentes.

Para uma ontologia ser descrita formalmente, é necessário que ela seja projetada. Deste modo, Gruber propõe um conjunto de critérios preliminares para descrição formal de ontologias, visando, principalmente, o compartilhamento de conhecimentos (GRUBER, 1993a):

a) Clareza (*clarity*): uma ontologia deve comunicar a intenção do significado dos seus termos, através da descrição formal dos conceitos e dos relacionamentos referentes a um domínio. Assim, uma ontologia deve ser descrita com:

- Objetividade: a definição dos termos devem ser independentes dos contextos social e computacional;
- Formalismo: os termos devem ser expressos através de axiomas lógicos ou por uma linguagem que os expressem logicamente;
- Completude: as definições dos termos devem ser completas (um predicado deve ser definido através de condições necessárias e suficientes), além de serem documentados em linguagem natural;

b) Coerência (*coherence*): uma ontologia deve ser coerente, ou seja, as inferências devem ser consistentes com as definições (os axiomas devem ser

logicamente consistentes) e com os conceitos que são definidos de modo não formal (exemplos e documentações em linguagem natural);

c) Extensibilidade (*extendibility*): uma ontologia deve ser capaz de definir novos termos para usos específicos, baseados no vocabulário existente e sem necessidade de rever as definições existentes;

d) Mínima codificação específica (*minimal encoding bias*): a conceitualização de uma ontologia deve ser especificada no nível do conhecimento, sem depender de codificações específicas ou particulares, a fim de facilitar o processo de compartilhamento do conhecimento entre sistemas;

e) Mínimo compromisso ontológico (*minimal ontological commitment*): considerando que *ontological commitment* é baseado no uso consistente de um vocabulário, minimizá-lo significar dizer que se devem definir somente os termos que são essenciais à comunicação do conhecimento, de modo consistente com o que se quer representar.

Construída com base nos critérios propostos por Gruber (1993b), uma ontologia deverá apresentar como características:

a) Entendimento comum sobre um domínio;

b) Semântica explícita (semântica é uma forma de descrição formal de termos e seus relacionamentos que suportam o entendimento de máquinas (computadores)). Explicitar semântica significa permitir que os *softwares* (*web applications*, *intelligent agents*, por exemplo) “entendam” as informações, façam suposições explícitas sobre um domínio, reduzam interpretações ambíguas e permitam interoperabilidade;

c) Expressividade. A expressividade de uma ontologia está relacionada com o grau de interpretação da mesma, por um *software*, a partir de sua representação;

d) Informação compartilhável. Ontologias suportam o compartilhamento, o uso e o reuso de informações, obtidos por meio de uma semântica explícita que expressa declarações. Um pré-requisito para compartilhamento é a utilização de uma mesma linguagem e o acesso às informações.

## 3.2 ELEMENTOS DE ESPECIFICAÇÃO DE UMA ONTOLOGIA

Considerando que uma ontologia é uma especificação formal de conceitos sobre um domínio, alguns componentes são esperados na composição desta especificação (NOY; MCGUINESS, 2001):

> 1) Classes ou conceitos (*classes* ou *concepts*): descrevem conceitos de um domínio. São usualmente organizadas em taxonomias através das quais mecanismos de herança podem ser aplicados (CORCHO *et al.*, 2006);
>
> 2) Relações ou propriedades (*relations* ou *slots* ou *roles* ou *properties*): são associados a cada classe, descrevendo suas características e seus atributos (NOY; MCGUINESS, 2001);
>
> 3) Facetas ou restrições (*facets* ou *role restrictions* ou *restrictions*): representam restrições sobre as relações (NOY; MCGUINESS, 2001).

Complementando, uma ontologia juntamente com um conjunto de membros (instâncias ou *instances*) de uma classe, compõe o que se denomina de uma base de conhecimentos (*base knowledge*) (CORCHO *et al.*, 2006; NOY; MCGUINESS, 2001).

## 3.3 CLASSIFICAÇÃO PARA ONTOLOGIA

Quanto aos tipos de ontologias, utilizamos como referência uma das formas de classificação proposta por Gómez-Pérez (2004), por ser bastante abrangente e uma das mais citadas em artigos relacionados ao tema:

> a) Ontologias de Representação do Conhecimento (*Knowledge Representation* (KR) *Ontologies*): provêem representações primitivas (classes, subclasses, atributos, instâncias, e relações) usadas para formalizar conhecimentos sob o paradigma KR, principalmente por linguagens baseadas em *frames*, permitindo a construção de outras ontologias através de convenções também baseadas em *frames* (VAN HEIJST *et al. apud* GÓMEZ-PÉREZ *et al.*, 2004). Exemplos: *Frame Ontology* (GRUBER *apud* GÓMEZ-PÉREZ *et al.*, 2004) e OKBC *Ontology*;

b) Ontologias Gerais (*General* ou *Common Ontologies*): usadas para representar os conhecimentos de senso comum, referenciados por vários domínios. Estas ontologias incluem vocabulários do tipo: *thing*, *event*, time, *space*, etc. (VAN HEIJST *apud* GÓMEZ-PÉREZ *et al.*, 2004; MIZOGUCHI *et al. apud* GÓMEZ-PÉREZ *et al.*, 2004). Exemplos: *Meteorology Ontology* (BORST *apud* GÓMEZ-PÉREZ *et al.*, 2004);

c) Ontologias de Alto Nível (*Top-level* ou *Upper-level* ou *Base Ontologies*): descrevem muitos conceitos e provêem noções gerais sobre quais os termos-raiz de uma determinada ontologia podem ser ligados. Como existem várias *Upper-level Ontologies* utilizando critérios distintos, o IEEE Standard Upper Ontology (SUO) Working Group vem tentando especificar uma *Upper-level Ontology*. Exemplos: SENSUS (possui termos como: *quality*, *object*, *process*), Cyc (possui termos como: *collection*, *individual*, *tangible*);

d) Ontologias de Domínio (*Domain Ontologies*): são reusáveis para um dado domínio (médico, farmacêutico, automobilístico, etc). Elas provêem vocabulários sobre conceitos referentes a um domínio e suas relações, sobre as atividades constantes num domínio, e sobre as teorias e princípios elementares que governam um domínio. Os conceitos de uma *Domain Ontology* são normalmente especializações daqueles já definidos pelas *Upper-level Ontologies* (VAN HEIJST *apud* GÓMEZ-PÉREZ *et al.*, 2004; MIZOGUCHI *et al. apud* GÓMEZ-PÉREZ *et al.*, 2004);

e) Ontologias de Tarefa (*Task Ontologies*): descrevem vocabulários relacionados com uma atividade ou tarefa genérica (*diagnosing*, *scheduling*, *selling*, etc.), através da especialização dos termos contidos numa *Upper-level Ontologies*. Estes termos podem pertencer ou não a um dado domínio (MIZOGUCHI *et al.*, 1995; GUARINO *apud* GÓMEZ-PÉREZ *et al.*, 2004);

f) Ontologias de Domínio-Tarefa (*Domain-Task Ontologies*): são aquelas reusáveis dentro de um domínio, mas não entre domínios. São independentes de aplicação;

g) Ontologias de Método (*Method Ontologies*): descrevem conceitos relevantes e relações, aplicados ao processo de raciocínios específicos de uma

tarefa em particular (TIJERINO; MIZOGUCHI *apud* GÓMEZ-PÉREZ *et al.*, 2004);

h) Ontologias de Aplicação (*Application Ontologies*): são dependentes de aplicações. Contêm definições necessárias ao modelo de conhecimento requerido para uma particular aplicação, sendo estendidas e especializadas a partir de vocabulários de Ontologias de Domínio e de Tarefa (VAN HEIJST *et al. apud* GÓMEZ-PÉREZ *et al.*, 2004).

As ontologias desenvolvidas para esta dissertação são classificáveis como sendo do tipo, Ontologias de Domínio.

## 3.4 METODOLOGIAS PARA CONSTRUÇÃO DE ONTOLOGIA

Segundo Hoog, “é extremamente difícil julgar o valor de uma metodologia de um modo objetivo. Naturalmente, a experimentação é a maneira apropriada para fazê-lo, apesar de ser pouco prático, considerando que há muitas condições que não podem ser controladas” (HOOG *apud* GÓMEZ-PÉREZ *et al.*, 2004).

Durante muitos anos, várias metodologias para construção de ontologias foram desenvolvidas para propósitos específicos. Em 1995, Uschold e King apresentou um trabalho contendo um método para construção de ontologias, baseado na experiência de ambos na área empresarial, e neste mesmo ano, Grüninger e Fox, apresentou uma metodologia para construção de ontologias contida no projeto *Toronto Virtual Enterprise* (TOVE). A partir de então, outras metodologias, de propósito mais geral, passaram a ser elaboradas, tendo como referências ambos os trabalhos.

Entretanto, até o momento, não há uma proposta unificadora para os padrões metodológicos existentes, nem tão pouco há uma abordagem metodológica capaz de cobrir todas as atividades referentes à construção de uma ontologia (FERNÁNDEZ-LÓPEZ; GÓMEZ-PÉREZ, 1999; CORCHO *et al.*, 2006).

Sendo assim, com a carência de padrões que permita a escolha de uma metodologia, esta dissertação parte de um *framework* com avaliações de metodologias existentes, aquelas com maior número de citações, elaborado por Gómez-Pérez *et al.* (2004), para propor uma alternativa a esta seleção. A proposta apresentada foi desenvolvida através da associação dos

critérios de avaliação contidos no *framework* (elaborado por Gómez-Pérez, Fernández-López e Corcho), com valores e pesos, os quais estão relacionados com o contexto estabelecido para construção de uma referida ontologia.

### 3.4.1 *Framework* para escolha de metodologia

No início da década de noventa do século passado, tivemos os primeiros movimentos para o desenvolvimento de metodologias para construção de ontologias, com a publicação do Cyc method, em 1990 (GÓMEZ-PÉREZ *et al.*, 2004), do Uschold & King´s *method*, em 1995 (USCHOLD; KING, 1995), do TOVE *Project Ontology* (GRÜNINGER; FOX, 1995), entre outros.

A maioria das metodologias elaboradas até aqui tiveram o seu foco concentrado sobre as atividades de desenvolvimento (conceitualização e implementação), e sem dar atenção para outras atividades como, por exemplo, *merging*, aprendizagem e avaliação de ontologias (GRÜNINGER; FOX, 1995).

Isto é reflexo da ausência de padrões unificadores, o que de certa forma reflete ainda uma imaturidade da Engenharia Ontológica ou *Ontology Engineering*, principalmente se comparada com a Engenharia de Software e com a Engenharia de Conhecimento (FERNÁNDEZ-LÓPEZ; GÓMEZ-PÉREZ, 1999).

Entre as mais citadas metodologias, tem-se, *The* Cyc *method*, Uschold & King's *method*, Grüninger & Fox´s *methodology*, *The* KACTUS *approach* ou *The* Amaya *method*, METHONTOLOGY, SENSUS-*based method*, On-To-Knowledge, SABiO (GÓMEZ-PÉREZ *et al.*, 2004; CORCHO *et al.*, 2006; FALBO *et al.*, 1998; FALBO; MENEZES, 2004).

A proposta para seleção de metodologia para construção de ontologias, apresentada nesta dissertação, tem como objetivo, alinhar a avaliação das metodologias quanto ao uso, quanto às atividades e quanto às etapas do processo de construção, com a importância dos critérios de avaliação utilizados, com a aplicação de valores e pesos, considerando-se o contexto de construção de uma referida ontologia.

Assim, parte-se do *framework* original, estendendo o mesmo com a atribuição de valores e pesos para os critérios de avaliação utilizados, de acordo com o contexto de construção, gerando tabelas complementares, a fim de se obter um valor final para cada metodologia

avaliada, e assim, sugerir a mais apropriada para o desenvolvimento de uma ontologia referente a um domínio específico.

### 3.4.2 *Framework* original

Em 2004, Gómez-Pérez *et al.*, apresentou um *framework* contendo a avaliação das principais metodologias existentes para construção de ontologias, utilizando um conjunto de quatro grupos de critérios, estruturados com base nos princípios da Engenharia de Software (GÓMEZ-PÉREZ *et al.*, 2004; USCHOLD; KING, 1995):

1º grupo: *Estratégia de Construção*, critérios de avaliação:

a) *Ciclo de vida – desenvolvimento*: refere-se aos estágios de desenvolvimento de uma ontologia, às atividades desenvolvidas e à relação entre os estágios, subdividindo-se em:

- *Desenvolvimento incremental*: segundo este critério, a ontologia evolui em camadas, permitindo a inclusão de novas definições somente quando uma nova versão é planejada;
- *Desenvolvimento por protótipos*: segundo este critério, a ontologia cresce conforme a necessidade, permitindo adições, remoções e modificações de definições a qualquer momento;

b) *Estratégia de acordo com a aplicação*: refere-se ao grau de dependência da ontologia com a possível aplicação que irá utilizar a mesma, subdividindo-se em:

- *Aplicação dependente*: segundo este critério, as ontologias são construídas com base nas aplicações que irão utilizá-las;
- *Aplicação semi-dependente*: segundo este critério, possíveis cenários de uso da ontologia são usados para desenvolver a especificação da ontologia;
- *Aplicação independente*: segundo este critério, as ontologias são construídas sem nenhuma relação com uma aplicação específica;

c) *Uso de núcleo ontológico*: refere-se ao uso ou não de uma base ontológica pré-existente, como ponto inicial para o desenvolvimento de uma Ontologia de Domínio (*Domain Ontology*);

d) *Estratégia para identificação de conceitos*; existem três critérios definidos:

- *Bottom-up*: a partir dos mais concretos para os mais abstratos;
- *Top-down*: a partir dos mais abstratos para os mais concretos;
- *Middle-out*: a partir dos mais relevantes para os mais abstratos e/ou para os mais concretos;

2º grupo: *Suporte Tecnológico*: refere-se à existência de ferramentas específicas para apoio à aplicação da metodologia;

3º grupo: *Processo de Desenvolvimento Ontológico*; possui como critérios de avaliação:

a) *Gerenciamento ontológico*: refere-se às atividades de gestão do processo de desenvolvimento de uma ontologia:

- *Programação* (*Scheduling*): segundo este critério, a metodologia deve prever a identificação das tarefas a serem desempenhadas, o modo como serão estruturadas e o tempo e os recursos necessários para serem completadas;
- *Controle*: segundo este critério, a metodologia deve prever a administração estrita das tarefas a serem executadas;
- *Qualidade*: segundo este critério, a metodologia deve prever a garantia de qualidade para as tarefas a serem realizadas;

b) *Desenvolvimento ontológico*: refere-se às atividades de desenvolvimento propriamente ditas:

*Atividades de pré-desenvolvimento*:

- *Estudo de ambiente*: segundo este critério, a metodologia deve prever uma avaliação de metas e cenários a serem atingidos com a implantação da ontologia;

- *Estudo de viabilidade*: segundo este critério, a metodologia deve prever uma avaliação sobre a disponibilidade de informações consistentes e confiáveis sobre um determinado domínio, sobre a existência e a disponibilidade de especialistas etc.;

*Atividades de desenvolvimento*:

- *Especificação*: segundo este critério, a metodologia deve prever aquisição de conhecimentos através: da identificação dos conceitos chaves e dos seus relacionamentos dentro do domínio de interesse (escopo ou *scoping*), da produção de textos precisos e sem ambigüidades para definir conceitos e seus relacionamentos, e da identificação de termos referentes aos conceitos e seus relacionamentos;
- *Conceitualização/Formalização*: segundo este critério, a metodologia deve prever a organização dos conhecimentos adquiridos na especificação, estruturá-los e transformá-los num modelo declarativo baseado, por exemplo, em expressões de lógica de primeira ordem;
- *Implementação*: segundo este critério, a metodologia deve prever a construção e a formalização de modelos estruturados para a ontologia, em um formato entendível para uma máquina (computador), possivelmente através do uso de uma linguagem;

*Atividades de pós-desenvolvimento*:

- *Manutenção*: segundo este critério, a metodologia deve prever atualizações e correções da ontologia desenvolvida;
- *Utilização*: segundo este critério, a metodologia deve prever quais os retornos obtidos pelos usuários que utilizam as aplicações apoiadas nas ontologias desenvolvidas;

c) *Suporte ontológico*: refere-se às atividades que apóiam todo o processo de desenvolvimento de ontologias:

- *Aquisição de conhecimento*: segundo este critério, a metodologia deve prever a atividade de aquisição do conhecimento a partir de especialistas ou através de outras fontes significativas no domínio especificado;
- *Avaliação*: segundo este critério, a metodologia deve prever o julgamento técnico da ontologia, o software associado a mesma e a documentação produzida em relação aos *frames* de referência (isto é, especificações de requisitos e questões de competência, quando for o caso);
- *Integração*: segundo este critério, a metodologia deve prever a integração entre a ontologia que está sendo desenvolvida e outras já existentes;
- *Gestão de configuração*: segundo este critério, a metodologia deve prever o registro de todas as versões da documentação e do código da ontologia para controle das mudanças efetuadas;
- *Documentação*: segundo este critério, a metodologia deve prever pelo menos a documentação de todas as hipóteses, os principais conceitos e as primitivas que expressam as definições da ontologia;
- *Merging* e *Alinhamento*. *Merging*: consiste na obtenção de uma nova ontologia a partir de várias outras ontologias (GANGEMI *et al., 1999*; NOY; MUSEN, 2001; STUMME; MAEDCHE *apud* GÓMEZ-PÉREZ *et al.*, 2004). *Alinhamento*: consiste no estabelecimento de diferentes tipos de *mappings* ou mapeamentos (ligações), os quais são associações entre termos e expressões definidos na ontologia "fonte" e entre termos e expressões de uma ontologia "alvo", a partir de regras re-escritas, sem gerar uma nova ontologia (NOY; MUSEN *apud* CORCHO *et al.*, 2006);

4º grupo: *Aplicação de Ontologias*. Possui como critérios de avaliação:

a) *Projetos onde a metodologia tem sido utilizada*;

b) *Aceitação por outras organizações* (em relação as que conceberam a metodologia);

c) *Ontologias criadas*;

d) *Domínios ontológicos atendidos*;

e) *Aplicações ou áreas de negócio que usaram as ontologias desenvolvidas* (áreas de concentração).

### 3.4.3 *Framework* estendido

O *framework* de Gómez-Pérez *et al.* (2004) estabelece a construção de tabelas individuais para cada um dos grupos de critérios, construindo células a partir do cruzamento entre os critérios específicos e as metodologias analisadas. Dentro de cada célula são atribuídos conteúdos estabelecidos conforme a natureza de cada grupo, utilizando-se fontes de informações públicas.

Para os 1º, 2º e 4º grupos os valores atribuídos expressam o conteúdo associado ao critério ou uma expressão de negação (negando a existência do objeto relativo ao conteúdo referente ao critério). Por exemplo, em relação ao 2º grupo (*Suporte Tecnológico*) o valor atribuído pode ser ou o nome de uma ou várias ferramentas que apóiam o uso da metodologia ou a expressão "nenhuma ferramenta específica" (GÓMEZ-PÉREZ *et al.*, 2004).

Para o 3º grupo, é proposto um esquema de conteúdos distinto, contendo as expressões (GÓMEZ-PÉREZ *et al.*, 2004):

a) "*Described*", significando que a metodologia prevê a descrição de como cada tarefa é desempenhada, quando é feita, quem a executa, etc.;

b) "*Proposed*", significando que a metodologia prevê apenas a identificação do processo, sem entrar em detalhes;

c) "*NP*", significando que a documentação pública não faz menção do item relacionado ao critério analisado.

Entretanto, o *framework* original oferece apenas a possibilidade de avaliar metodologias distintas, o que não é suficiente para encaminhar um procedimento de escolha, uma vez que não oferece mecanismo de comparação entre as mesmas.

Sendo assim, é que propomos que o mesmo seja estendido, seguindo a orientação de Fernández-López e Gómez-Pérez (1999), alinhando o conhecimento resultante da avaliação obtida, com as necessidades referentes ao contexto para a construção de uma ontologia.

Para tanto, complementamos o *framework* original, executando o seguinte roteiro:

1º) Construir as seguintes tabelas de referência, tabelas 1 e 2, com valores expressos em números para substituir àqueles propostos pelo *framework* original, considerando para o caso da tabela 1 (contendo valores atribuídos aos critérios de avaliação pertencentes aos grupos 1º, 2º e 4º) a conversão para o valor de referência 3, quando existir expressão de conteúdo na célula das tabelas, e conversão para o valor de referência 1, quando existir uma expressão de negação. E para a tabela 2 (contendo valores atribuídos aos critérios de avaliação pertencentes ao grupo 3º), a conversão para os valores de referência 3 ou 2 ou 1, quando o conteúdo das células forem *Described* ou *Proposed* ou *NP*, respectivamente.

Tabela 1 - Tabela de referência para conversão em valores (1º, 2º e 4º grupos)

| 1º, 2º e 4º grupos de critérios | Valores propostos pelo *framework* | |
|---|---|---|
| | Expressão de conteúdo | Expressão de negação ou "Não especificado" |
| Valores de referência (expressos em números) | 3 | 1 |

Nota: elaboração própria

Tabela 2 - Tabela de referência para conversão de valores (3º grupo)

| 3º grupo de critérios | Valores propostos pelo *framework* | | |
|---|---|---|---|
| | Described | Proposed | NP |
| Valores de referência (expressos em números) | 3 | 2 | 1 |

Nota: elaboração própria

2º) Construir uma tabela de significância (Tabela 3), atribuindo pesos aos critérios de avaliação propostos pelo *framework* original, considerando o contexto referente à ontologia que será construída. Atribui-se o peso 0 para os critérios com pouca significância em dado contexto, e o peso 2 para aqueles com mais significância.

Tabela 3 - Tabela de significância

| Significância para o domínio | Peso |
|---|---|
| Pouco significante | 0 |
| Significante | 2 |

Nota: elaboração própria

3º) Multiplicar cada valor atribuído pelos pesos, a fim de se obter um valor único para cada critério para cada metodologia avaliada;

4º) Somar os valores das células de cada metodologia e transportar os valores totais para uma tabela Resultante (Tabela 6), na qual será possível visualizar as somas totais obtidas para a relação VALOR X PESO, para cada metodologia por critério e por grupo;

5º) Analisar, ao final, a tabela Resultante, onde os maiores valores indicarão as possíveis metodologias mais adequadas para demanda estabelecida.

### 3.4.4 Aplicação do *framework* estendido

A dissertação está apoiada na necessidade de construção de duas ontologias: uma relacionada a problemas de fraude e outra relacionada a técnicas para detecção automática de fraudes.

Assim sendo, analisando-se o contexto dos domínios das ontologias propostas e do propósito inicial para a construção das mesmas, é que se aplicou o *framework* detalhado anteriormente do modo que se segue:

1º) Acrescentou-se a metodologia SABiO (FALBO *et al.*, 1998; FALBO; MENEZES, 2004), por se tratar de uma metodologia brasileira com citações no Brasil e no exterior, ao conjunto de metodologias contidas no *framework* original, quais sejam, Cyc *method*, Uschold & King´s *method*, Grüninger & Fox´s *methodology*, KACTUS *method*, METHONTOLOGY, SENSUS *method* e On-To-Knowledge *methodology*;

2º) Atribui-se aos critérios de avaliação para os grupos 2º (*Suporte Tecnológico*) e 4º (*Aplicação de Ontologias*) pesos referentes a pouca significância, considerando o contexto da dissertação. Foi atribuído ao critério estabelecido para o 2º grupo (*Suporte Tecnológico*) peso 0 ou pouco significante, porque se decidiu pelo emprego de

ferramentas com aplicabilidade geral e não proprietárias no contexto da dissertação, como é o caso da ferramenta *Protégé*.

O mesmo se deu para os critérios de avaliação que compõem o 4º grupo (*Aplicação de Ontologias*), atribuindo-se peso 0 a todos eles, considerando que para os domínios de problemas de fraudes e de técnicas para detecção automática de fraudes foram identificadas poucas iniciativas na área de ontologia, sendo que em alguns casos se utilizou metodologias e ferramentas proprietárias, como as metodologias AKEM e DOGMA usadas no projeto FFPOIROT.

Deste modo, como todos os critérios de avaliação para o 2º e 4º grupos retornariam valor final 0, não houve necessidade de se reproduzir as tabelas convertidas para tais grupos;

3º) Construiu-se as tabelas para os grupos de critérios 1º (*Estratégia de Construção*) e 3º (*Processo de Desenvolvimento Ontológico*) (tabelas 4 e 5 respectivamente);

4º) Estabeleceu-se como significantes os seguintes critérios pertencentes ao 1º grupo (*Estratégia de Construção*):

a) Critério para a abordagem *Middle-out* foi considerado como significante. Porque para ambos os domínios, problemas de fraude e técnicas para detecção automática de fraudes, o processo de identificação de palavras-chave e relacionamentos não obedece a uma hierarquia, uma vez que o conhecimento no geral está disposto de modo horizontal, tornando a presença de conceitos concretos e abstratos para estes domínios algo pouco relevante;

b) Critério para *Aplicação independente* foi considerado como significante, pois a proposta é desenvolver uma base de conhecimentos sem associar, previamente, a uma aplicação ou a um cenário específico;

5º) Estabeleceu-se como significantes os seguintes critérios pertencentes ao 3º grupo (*Processo de Desenvolvimento Ontológico*):

a) Critérios para *Especificação e Conceitualização/Formalização* foram considerados como significantes, por serem fundamentais no processo de construção ontológica, em qualquer contexto;

b) Critérios para *Aquisição do conhecimento*, *Avaliação*, *Integração* e *Documentação* foram considerados como significantes, porque no contexto desta dissertação, tem-se como objetivo gerar uma base de conhecimentos que possa vir a ser mantida e expandida futuramente;

c) Critério para *Merging/Alinhamento* foi considerado como significante, pois a base de conhecimentos desenvolvida para dissertação é complementada por um *merging* entre as duas ontologias referentes aos domínios de problemas de fraude e de técnicas para detecção automática de fraudes, obtendo-se uma terceira;

6º) Executou-se as operações a fim de se obter a tabela resultante (Tabela 6), e assim se levantou a indicação de qual a metodologia seria mais adequada ao projeto desta dissertação, baseado no maior valor encontrado.

Para aplicar o *framework* estendido, visando determinar qual metodologia se apresenta como a mais apropriada, primeiro, gerou-se as tabelas para os grupos de critérios 1º (*Estratégia de Construção*) e 3º (*Processo de Desenvolvimento Ontológico*) (Tabelas 4 e 5 respectivamente). O quadro 1 contêm a legenda para as tabelas 4,5 e 6 e possui como fonte o trabalho de Gómez-Pérez *et al.* (2004):

| **Acrônimo** | **Descrição** | **Acrônimo** | **Descrição** |
|---|---|---|---|
| Cyc | Cyc method | Tabela 5 – a.3 | (Critério) Qualidade |
| U&K´s | Uschold & King's method | Tabela 5 – b.1 | (Critério) Atividades de pré-desenvolvimento |
| G&F´s | Grüninger & Fox´s methodology | Tabela 5 – b.1.1 | (Critério) Estudo de ambiente |
| KAC. | KACTUS approach ou Amaya method | Tabela 5 – b.1.2 | (Critério) Estudo de viabilidade |
| MET. | METHONTOLOGY | Tabela 5 – b.2 | (Critério) Atividades de desenvolvimento |
| SENS. | SENSUS-based method | Tabela 5 – b.2.1 | (Critério) Especificação |
| O-T-K. | On-To-Knowledge | Tabela 5 – b.2.2 | (Critério) Conceituação/Formalização |
| SABiO | SABiO | Tabela 5 – b.2.3 | (Critério) Implementação |
| Tabela 4 – a.1 | (Critério) Desenvolvimento incremental | Tabela 5 – b.3 | (Critério) Atividades de pós-desenvolvimento |
| Tabela 4 – a.2 | (Critério) Desenvolvimento protótipo | Tabela 5 – b.3.1 | (Critério) Manutenção |
| Tabela 4 – b.1 | (Critério) Aplicação dependente | Tabela 5 – b.3.2 | (Critério) Utilização |
| Tabela 4 – b.2 | (Critério) Aplicação semi-dependente | Tabela 5 – c.1 | (Critério) Aquisição do conhecimento (*knowledge* |

| | | | |
|---|---|---|---|
| | | | *acquisition*) |
| Tabela 4 – b.3 | (Critério) Aplicação independente | Tabela 5 – c.2 | (Critério) Avaliação |
| Tabela 4 – d.1 | (Critério) *Bottom-up* | Tabela 5 – c.3 | (Critério) Integração |
| Tabela 4 – d.2 | (Critério) *Top-down* | Tabela 5 – c.4 | (Critério) Gestão de configuração |
| Tabela 4 – d.3 | (Critério) *Midde-out* | Tabela 5 – c.5 | (Critério) Documentação |
| Tabela 5 – a.1 | (Critério) Programação (*schedul*ing) | Tabela 5 – c.6 | (Critério) *Merging* eAlinhamento |
| Tabela 5 – a.2 | (Critério) Controle | | |

Quadro 1 - Legenda para as tabelas 4, 5 e 6

Tabela 4 - 1º Grupo (Estratégia de Construção)

| | | Metodologias | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Cyc | | U&K´s | | G&F`s | | KAC. | | MET. | | SENS. | | O-T- K. | | SABiO | |
| Critérios | | V | P | V | P | V | P | V | P | V | P | V | P | V | P | V | P |
| a) Ciclo de vida | a.1 | 1 | 0 | 3 | 0 | 3 | 0 | 1 | 0 | 3 | 0 | 1 | 0 | 3 | 0 | 3 | 0 |
| | a.2 | 3 | 0 | 3 | 0 | 3 | 0 | 3 | 0 | 3 | 0 | 1 | 0 | 3 | 0 | 3 | 0 |
| b) Estratégia de acordo com a aplicação | b.1 | 1 | 0 | 1 | 0 | 1 | 0 | 3 | 0 | 1 | 0 | 1 | 0 | 3 | 0 | 1 | 0 |
| | b.2 | 1 | 0 | 1 | 0 | 3 | 0 | 1 | 0 | 1 | 0 | 3 | 0 | 1 | 0 | 1 | 0 |
| | b.3 | 3 | 2 | 3 | 2 | 1 | 2 | 1 | 2 | 3 | 2 | 1 | 2 | 1 | 2 | 3 | 2 |
| c) Núcleo ontológico | | 3 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 3 | 0 | 3 | 0 | 1 | 0 |
| d) Estratégia para identificação de conceitos | d.1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 3 | 0 | 1 | 0 |
| | d.2 | 1 | 0 | 1 | 0 | 1 | 0 | 3 | 0 | 1 | 0 | 1 | 0 | 3 | 0 | 1 | 0 |
| | d.3 | 1 | 2 | 3 | 2 | 3 | 2 | 1 | 2 | 3 | 2 | 1 | 2 | 3 | 2 | 3 | 2 |
| Totalização | | 8 | | 12 | | 8 | | 4 | | 12 | | 4 | | 8 | | 12 | |

Nota: elaboração própria

Tabela 5 - 3º Grupo (Processo de Desenvolvimento Ontológico) (continua)

| | | | Metodologias | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Cyc | | U&K´s | | G&F`s | | KAC. | | MET. | | SENS. | | O-T- K. | | SABiO | |
| Critérios | | | V | P | V | P | V | P | V | P | V | P | V | P | V | P | V | P |
| a)Gerenciamen. Ontológico | | a.1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 2 | 0 | 1 | 0 | 3 | 0 | 1 | 0 |
| | | a.2 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 2 | 0 | 1 | 0 | 3 | 0 | 1 | 0 |
| | | a.3 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 2 | 0 | 1 | 0 | 3 | 0 | 1 | 0 |
| | b.1 | b.1.1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 2 | 0 | 1 | 0 |
| | | b.1.2 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 3 | 0 | 1 | 0 |
| b)Desenvolvim. Ontológico | b.2 | b.2.1 | 1 | 2 | 3 | 2 | 3 | 2 | 2 | 2 | 3 | 2 | 2 | 2 | 3 | 2 | 3 | 2 |
| | | b.2.2 | 1 | 2 | 3 | 2 | 3 | 2 | 2 | 2 | 3 | 2 | 1 | 2 | 3 | 2 | 3 | 2 |
| | | b.2.3 | 2 | 2 | 3 | 2 | 3 | 2 | 2 | 2 | 3 | 2 | 3 | 2 | 3 | 2 | 3 | 2 |
| | b.3 | b.3.1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 2 | 0 | 1 | 0 | 2 | 0 | 1 | 0 |
| | | b.3.2 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 2 | 0 | 1 | 0 |
| c)Suporte Ontológico | | c.1 | 2 | 2 | 3 | 2 | 2 | 2 | 1 | 2 | 2 | 2 | 1 | 2 | 3 | 2 | 2 | 2 |
| | | c.2 | 1 | 2 | 3 | 2 | 3 | 2 | 1 | 2 | 3 | 2 | 1 | 2 | 2 | 2 | 3 | 2 |

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| c.3 | 2 | 3 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 1 | 2 | 2 | 2 | 2 | 2 |
| c.4 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 3 | 0 | 1 | 0 | 2 | 0 | 1 | 0 |
| c.5 | 3 | 2 | 3 | 2 | 3 | 2 | 2 | 2 | 3 | 2 | 2 | 2 | 3 | 2 | 3 | 2 |
| c.6 | 1 | 2 | 1 | 2 | 1 | 2 | 1 | 2 | 1 | 2 | 1 | 2 | 1 | 2 | 1 | 2 |
| Totalização | 28 | | 42 | | 40 | | 26 | | 40 | | 26 | | 32 | | 40 | |

Nota: elaboração própria

Em seguida, transportaram-se as totalizações obtidas nas tabelas 4 e 5 para cada grupo de critérios por metodologias na tabela Resultante (Tabela 6), respectivamente:

Tabela 6 - Resultante

| | Metodologias | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Critérios (Grupos) | Cyc | U&K´s | G&F`s | KAC. | MET. | SENS. | O-T- K. | SABiO |
| 1º | 8 | 12 | 8 | 4 | 12 | 4 | 8 | 12 |
| 3º | 28 | 42 | 40 | 26 | 40 | 26 | 32 | 40 |
| Total Geral | 36 | 54 | 48 | 30 | 52 | 30 | 40 | 52 |

Nota: elaboração própria

Finalmente, após efetuar o somatório dos valores lançados na tabela Resultante (Tabela 6), identificou-se a metodologia ou as metodologias que obtiveram o maior valor, obtendo-se a indicação pretendida.

No caso da dissertação, a aplicação do *framework* estendido apontou a metodologia Uschold & King´s *method* como aquele de maior somatório de valores, cinqüenta e quatro (54), indicando-a, portanto, como a mais alinhada ou mais apropriada para apoiar a construção das ontologias previstas.

A indicação da Uschold & King´s *method*, entretanto, trouxe uma lacuna no que tange as atividades de *merging/alinhamento*, que é considerado como critério com significância, no contexto considerado. Neste caso, um método ou uma metodologia complementar deve ser buscado para que esta atividade possa ser executada.

Com isto, nota-se que nem sempre a metodologia indicada contempla todas as atividades necessárias ao desenvolvimento das ontologias requeridas, algo que foi primeiramente observado por Gómez-Pérez *et al.* (2004).

### 3.4.5 Método para *Merge*

Quando uma metodologia não consegue atender a todas as atividades necessárias à construção de uma ontologia, é preciso buscar um método que complemente a opção definida.

Existem pelo menos três métodos ou metodologias bastante utilizadas para atividades de *merging*:

a) *Onions*, desenvolvida pelo Conceptual Modeling Group of the CNR, em Roma/Itália, que permite a criação de uma biblioteca de ontologias a partir de diferentes fontes (GÓMEZ-PÉREZ *et al.*, 2004);

b) FCA-*Merge*, desenvolvida pelo Institute AIFB da University of Karlsuhe na Alemanha, voltada para ontologias consideradas *lightweight* ou pouco complexas (STUMME; MAEDCHE *apud* GÓMEZ-PÉREZ *et al.*, 2004);

c) *Prompt*, desenvolvido pelo Stanford Medical Informatics, grupo da Stanford University, onde também foi projetada e desenvolvida a ferramenta *Protégé*, que tem disponibilidade de um *plug-in* (Prompt), baseado neste método, compatível até as versões inferiores a 4.0 (GÓMEZ-PÉREZ *et al.*, 2004; NOY; MUSEN, 2000; PROTÉGÉ, 2008).

Como a dissertação tem implementado suas ontologias no *Protégé* versão 4.0 beta, a operação de *merging* é executada através de um *plug-in* já incorporado a referida versão, cujo processo de execução é similar àquele presente no método *Prompt* (PROTÉGÉ, 2005).

## 3.5 LINGUAGENS DE REPRESENTAÇÃO E FERRAMENTAS DE APOIO

Segundo o conceito de ontologia defendido por Gruber (1993a), “uma ontologia é uma especificação formal de uma conceitualização”, é necessário que uma ontologia seja expressa de tal forma que possa ser legível para máquinas (computadores). Deste modo, o uso de linguagens de representação e ferramentas de apoio é algo importante.

### 3.5.1 Linguagens de representação

As linguagens de representação de ontologias foram criadas no início dos anos noventa do século XX (CORCHO *et al.*, 2006). Estas linguagens são divididas em dois tipos (GÓMEZ-PÉREZ *et al.*, 2004):

a) *Traditional Ontology Languages* (Quadro 2). É considerada, normalmente, uma evolução daquelas utilizadas para representação do conhecimento (*Knowledge Representation* – KR), sendo baseadas, na sua maioria, em lógica de primeira ordem, em *frames* combinados com lógica de primeira ordem e em *Description Logics* (DL´s);

*b)* *Web-based Ontology Languages* ou *Ontology Markup Languages* (Quadro 3). São aquelas cuja sintaxe é baseada em linguagens de marcação (*markup*), principalmente, XML. A criação deste tipo de linguagem tem relação direta com a necessidade de explorar as características da *Web* e das suas novas tendências, como a *Web Semântica*.

| Linguagem | Descrição |
|---|---|
| KIF | Knowledge Interchange Format, basea-se em lógica de primeira ordem; propõe-se resolver o problema da heterogeneidade da representação do conhecimento (KR) e permitir a troca entre diversos sistemas de informações (GENESERETH; FIKES *apud* CORCHO *et al*., 2006). |
| Ontolingua | Criada pelo Knowledge System Laboratory (KSL)/Stanford University, basea-se na combinação de frames com lógica de primeira ordem, visando apoiar a geração de ontologias situadas no Ontolingua Server (FARQUHAR *et al*. 1997; GRUBER apud CORCHO *et al*., 2006). |
| FLogic | Frame Logic, basea-se na combinação de frames com lógica de primeira ordem; desenvolvida originalmente como uma abordagem orientada a objeto para lógica de primeira ordem e usada para manipular banco de dados orientados a objeto; mais tarde foi adaptada para implementação de ontologias (KIFER *et al.* apud CORCHO *et al*., 2006). |
| Loom | Basea-se em DL´s, não foi construída como uma linguagem para ontologias e sim, como um ambiente para construção de sistemas especialistas de uso geral; descreve modelos de domínios através de objetos e relações (TBox) e fatos sobre indivíduos (ABox) (MACGREGOR apud CORCHO *et al*., 2006). |

Quadro 2 - *Traditional Ontology Languages* – alguns exemplos

| Linguagem | Descrição |
|---|---|
| SHOE | Simple HTML Ontology Extension - SHOE, desenvolvida pela University of Maryland, foi criada como extensão da HTML, apesar das tags utilizadas na representação de ontologias não serem definidas em HTML. Seu propósito é incorporar conhecimento semântico legível para máquinas em documentos Web (LUKE; HEFLIN apud CORCHO *et al*., 2006). |
| OIL | Ontology Interchange Language/Ontology Inference Layer - OIL, pertence ao contexto do projeto On-To-Knowledge e é considerada uma linguagem Web-based KR, combinando sintaxe XML com modelagem de primitivas a partir do paradigma KR (HORROCKS *et al.* apud CORCHO *et al*., 2006). |
| DAML+OIL | DARPA Agent Markup Language – DAML+OIL foi criada a partir junção dos trabalhos desenvolvidos pelos projetos On-To-Knowledge (europeu) e DARPA (americano), com o propósito de ser uma linguagem que permitisse a implementação de semântica de marcação (semantic markup) para recursos da Web (HORROCKS; VAN HARMELEN *apud* CORCHO *et al*., 2006). |
| RDF/RDF Schema | Resource Description Framework/Schema – RDF/RDFS vem sendo desenvolvida pelo World Wide Web Consortium W3C dentro do esforço para a criação de um metadados com o objetivo de descrever os recursos da Web. O modelo de dados usado por RDF/RDFS é equivalente àquele das redes semânticas formais, consistindo de três tipos de objetos: recursos (*resources*), propriedades (*properties*) e declarações (*statements*) (LASSILA; SWICK, 1999; BRICKLEY; GUHA *apud* CORCHO *et al*., 2006). |
| OWL | Ontology Web Language - OWL é o resultado do trabalho do W3C Web Ontology (WebOnt) Working Group. Derivada da DAML+OIL, é construída sobre RDF/RDFS. As ontologias em OWL usam sintaxe em XML e herdam notação de triplas oriundas de RDF (LACY, 2005). |

Quadro 3 - *Ontology Markup Languages* – alguns exemplos

### 3.5.2 Seleção da linguagem de representação

Para definir a linguagem de representação a ser utilizada na construção das ontologias, consideramos os seguintes aspectos:

1º) A premissa que as ontologias a serem construídas para a dissertação são do tipo ***lightweight***, *ou seja*, compostas basicamente de conceitos ou classes, atributos ou propriedades, e relações (CORCHO *et al.*, 2006), em contraponto as ontologias ***heavyweight***, onde são adicionadas mais restrições (*constraints*) semânticas de um determinado domínio (GÓMEZ-PÉREZ *et al.*, 2004);

2º) A aplicação do *framework* descrito inicialmente por Corcho e Gómez-Pérez para seleção de linguagem de representação, o qual permite avaliar a expressividade e a "capacidade de raciocínio" de uma linguagem ontológica, a partir da análise de duas dimensões fundamentais: a representação do conhecimento (KR) e os mecanismos de raciocínio (CORCHO; GÓMEZ-PÉREZ *apud* GÓMEZ-PÉREZ *et al.*, 2004);

3º) A determinação em desenvolver ontologias **semi-formais** (expressadas em linguagem definida artificial e formalmente (exemplos, Ontolingua, OWL)), em contra-ponto àquelas **altamente informais** (expressadas em linguagem natural), **semi-informais** (expressadas em linguagem natural, mas de modo restrito e estruturado), e **rigorosamente formais** (expressadas de forma meticulosa, a partir de termos definidos com semântica formal, teoremas e provas de propriedades tais como integridade e integralidade), conforme definição proposta por Gómez-Pérez *et al.* (2004);

4º) A opção por uma linguagem baseada em uma tecnologia, que permita o acesso fácil ao conhecimento disponibilizado, facilitando o compartilhamento, o entendimento, a manipulação, a extensão, e a geração de um novo conhecimento;

Assim, escolheu-se como linguagem de representação, OWL, uma vez que:

a) Em relação à aplicação do *framework* citado anteriormente (CORCHO; GÓMEZ-PÉREZ *apud* GÓMEZ-PÉREZ *et al.*, 2004), OWL se apresenta de modo plenamente satisfatório em relação a:

a.1) Representação do conhecimento (KR):

a.1.1) Conceitos ou Classes: implementa atributos de instâncias, atributos de classes, restrições (*constraints*), cardinalidade (mínimo/máximo);

a.1.2) Conceitos de Taxonomia: implementa *subclass-of*, disjunção-decomposição, exaustiva-decomposição, partição;

a.1.3) Relações ou Propriedades: implementa relações binárias, relações *n-ary*, relações hierárquicas;

a.1.4) Implementa instâncias;

a.2) Mecanismos de raciocínio:

a.2.1) Implementa classificações automáticas para conceitos (classes);

a.2.2) Implementa herança múltipla;

a.2.3) Implementa *constraint checking*, ou seja, detecta inconsistências na ontologia;

b) OWL é uma linguagem do tipo *Ontology Markup Languages*, portanto baseada em XML, permitindo que as ontologias implementadas com a mesma tenham alta portabilidade entre aplicações e sejam facilmente lidas e gerenciadas a partir de bibliotecas padrões, além de permitir que muitas ferramentas de apoio possam editá-las, manipulá-las e documentá-las (CORCHO *et al.*, 2006).

### 3.5.3 Ferramentas de apoio

As ferramentas de apoio são usadas para criar, manter, documentar, importar e exportar ontologias, além de permitir a visualização gráfica das taxonomias contextualizadas, mantendo bibliotecas ontológicas e operando mecanismos de inferência, entre outras funcionalidades.

Existem várias ferramentas de apoio, dentre elas temos:

a) OntoEdit: criada pelo Institute AIFB/University of Karlsruhe/Alemanha, está na sua versão 0.6, disponível nos formatos *freeware* e comercial. É uma aplicação em Java, que armazena ontologias em arquivos, suporta F-Logic, RDF-Schema e OIL, e tem suas funções estendidas através de *plug-ins* (SURE *et al. apud* CORCHO *et al.*, 2006);

b) OilEd: desenvolvido em 2001 pelo IMG da University of Manchester/Inglaterra, está disponível no formato *freeware*. É uma aplicação Java, que armazena ontologias em arquivos, suporta OIL e exporta ontologias em OIL-RDF e DAML-RDF (BECHHOFER *et al. apud* CORCHO *et al.*, 2006);

c) Ontolingua Server: é uma *suite* de ferramentas para construção colaborativa de ontologias, desenvolvida, implementada e mantida pelo Knowledge Systems Laboratory/Stanford University, objetivando, principalmente, o desenvolvimento de ontologias de modo distribuído e colaborativo. Tem sua interface no formato *web* e suporta edição de ontologias na linguagem Ontolingua (FARQUHAR *et al.*, 1996);

d) WebODE: é uma *suite* avançada para construção de ontologias sobre uma arquitetura *n-tier*, criada e mantida pelo Ontology Group/Universidad Politécnica de Madrid. Ela se encontra na versão 2.0.9, possui suporte metodológico para METHONTOLOGY, armazena ontologias em bases de dados relacionais, e importa e exporta ontologias para linguagens como: XML, RDF(S), DAML+OIL, OWL (ARPÍREZ *et al.*, 2003);

e) *Protégé*: é uma ferramenta desenvolvida em Java pelo SMI (Stanford Medical Informatics/Stanford University) e se encontra na versão 4.0. Está disponível no formato *freeware* e nos modos *stand-alone* e *web*, suporta a linguagem OWL e os seus dialetos, apóia a metodologia METHONTOLOGY, e tem suas funcionalidades estendidas por uma grande quantidade de *plug-ins*.

Nesta dissertação, optamos pelo *Protégé* versão 4.0 beta, pois além das características descritas acima que a torna alinhada com os objetivos desta dissertação, ela possui como característica principal, a sua alta flexibilidade.

## 3.6 ONTOLOGIAS APLICADAS A FRAUDES

O projeto da dissertação trabalha com o desenvolvimento de uma base de conhecimentos composta de duas ontologias, técnicas para detecção automática de fraudes e problemas de fraudes, e uma terceira resultante do processo de *merging* entre as duas primeiras.

Nas pesquisas efetuadas, identificamos duas ontologias voltadas para o problema de fraude, uma proposta pelo projeto FFPOIROT da Comunidade Econômica Européia (ZHAO; LEARY, 2005) e outra desenvolvida pela EHFCN, *Generic Fraud Ontology in e-Government* (ALEXOPOULOS *et al.*, 2006), além da ontologia UKG, voltada para o domínio de técnicas para detecção automática de fraudes.

### 3.6.1 Projeto FFPOIROT: Topical Ontology of Fraud

Em 2005, foi disponibilizada a versão 3.0 da *Topical Ontology of Fraud* (TOF), contida no projeto FFPOIROT, e desenvolvida pelo *Information Society Technologies Europoean* (IST EC) da Comunidade Econômica Européia (ZHAO; LEARY, 2005).

Para o desenvolvimento da TOF, um conceito fundamental foi estabelecido inicialmente: *Topical Ontology* ou Ontologia Tópica.

O conceito de *Topical Ontology* ou Ontologia Tópica foi introduzido, segundo o projeto FFPOIROT, para fazer distinção de outros tipos de ontologias, tais como, *Application Ontology*, *Domain Ontology* Ou *Base Ontology* (ZHAO; LEARY, 2005). De acordo com este projeto, *Topical Ontology* é um tipo de ontologia que trata sobre um conjunto de temas que representam a estrutura de conhecimento de um dado domínio (escopo, hipóteses, princípios, regras e padrões referentes aos temas), consistindo de objetos e relações abstratos e do modo como eles estão reunidos para estruturar o conhecimento (DELGADO *apud* ZHAO; MEERSMAN, 2006).

O conteúdo de uma *Topical Ontology* não é necessariamente composto por conceitos universais, e sim, de um corte específico no espaço do conhecimento que tem uma relevância em particular, para um ângulo próprio de observação e granularidade (NILES; PEASE, 2001).

Fraude é um conceito para o qual a abordagem referente à *Topical Ontology* pode ser aplicada, de acordo com projeto FFPOIROT, pois o mesmo não se reduz a um contexto que possa ser definido como uma Ontologia de Aplicação (*Application Ontology*), pois não possui detalhes operacionais relevantes que possam ser capturados e formalizados em relação a uma tarefa em particular. Não pode ser conceitualizado como uma Ontologia de Domínio (*Domain Ontology*), pois não se aplica a um conhecimento especializado específico, e por ser suficientemente descritivo, e também, está aquém do nível de generalização proposto por uma Ontologia de Base ou *Base Ontology* (ZHAO; MEERSMAN, 2006).

Deste modo, uma *Topical Ontology of Fraud* (TOF) tem como objetivo, capturar, essencialmente, entidades e relacionamentos conceituais que compõem a estrutura de conhecimentos referentes a fraudes, assumidos e compartilhados por profissionais que trabalham com o assunto, tais como, examinadores de fraudes, investigadores, detetives, auditores e analistas de inteligência (ZHAO; LEARY, 2005).

A TOF se propõe a servir como base para: aplicações de gestão de conhecimento e sistemas de informação voltados à detecção e ao monitoramento de ocorrências de fraudes (ZHAO; LEARY, 2005).

A arquitetura do TOF é composta por módulos, os quais são pacotes que representam a conceitualização de vários pontos de vistas. Nestes pacotes encontramos classes e propriedades referentes: à tipologia de fraudes (*Fraud Type*), à qualidade (*Participant Profile* e *Motivation*), às atividades do ciclo de vida da gestão antifraude (*Prevention*, *Detection*, *Investigation*, *Resolution*), aos atores que praticam atos fraudulentos (*Actors*), e à configuração de fraudes, *Fraud Configuration* (ZHAO; LEARY, 2005; ZHAO; MEERSMAN, 2006). Ver figura 3.

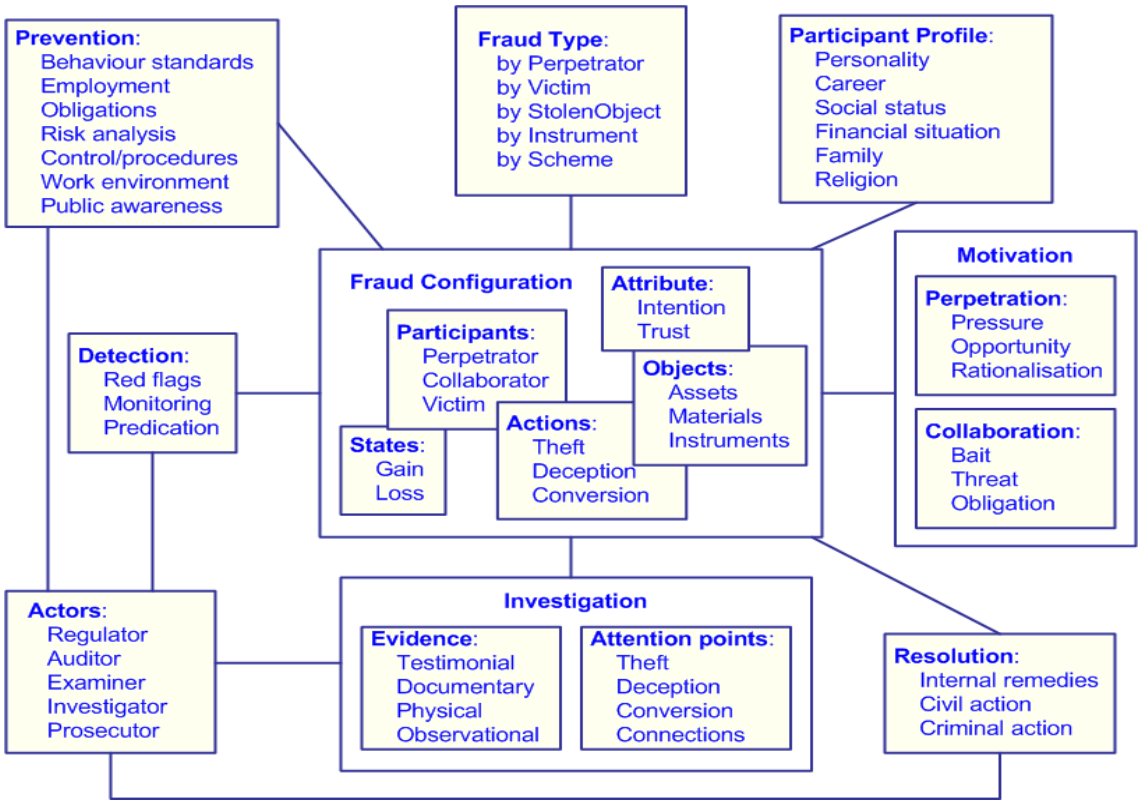


Figura 3 - TOF´s *Fraud Architecture*

Nota: elaborado por Zhao e Leary (2005)

Cada um destes pacotes é detalhado a partir das classes e relações identificadas, conforme descrição no anexo II.

Assim, a TOF funciona como uma camada ontológica intermediária ou de ligação na base de conhecimentos (ZHAO; LEARY, 2005), conforme arquitetura demonstrada na figura 4. Nela, problemas de fraude como uma *Topical Ontology* se apresenta como uma ontologia de referência para aquelas voltadas ao desenvolvimento de aplicações específicas (*Application Ontology*), tendo como base ontologias de Domínio (*Domain Ontology*) e de Alto Nível (*Base Ontology*).

Para o projeto FFPOIROT, problemas de fraude são vistos como algo tópico, que deve ser usado para definir outros domínios (JEN; JASON, 1998).

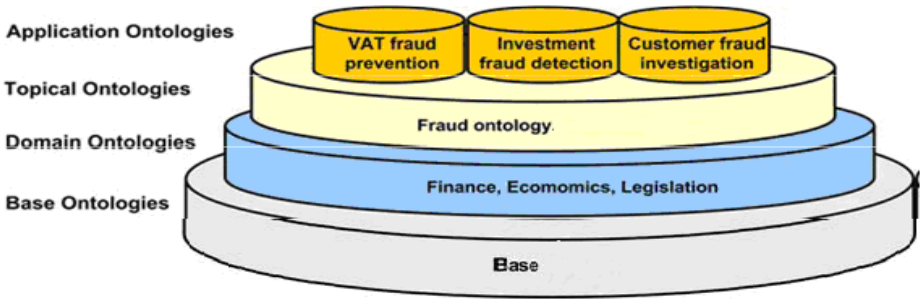


Figura 4 - Posicionamento da TOF
Nota: elaborado por Zhao e Leary (2005)

### 3.6.2 Generic Fraud Ontology in e-Government

*Generic Fraud Ontology in e-Government* é, na verdade, um *framework*, desenvolvido pela EHFCN, que se propõe construir uma ontologia genérica para fraude, estruturada sobre uma arquitetura em camadas, com a finalidade de servir como base ontológica comum, sobre a qual, várias ontologias específicas sobre fraudes poderão ser construídas (ALEXOPOULOS *et al.*, 2006).

A metodologia utilizada como apoio, está fundamentada sobre duas premissas. A primeira declara que os aspectos pertinentes ao conceito de fraude estão concentrados no contexto do processo de detecção de eventos fraudulentos e que este processo é considerado similar ao processo de gestão de risco operacional presente nas organizações (ALEXOPOULOS *et al.*, 2006).

A segunda premissa estabelece os conceitos que definem as camadas que constituem a arquitetura (Figura 5), gerada pelo *framework* (ALEXOPOULOS *et al.*, 2006):

a) *Generic Upper Ontology*: camada onde são capturados os conhecimentos genéricos e independentes de um domínio, ajudando a minimizar as redundâncias e duplicidades dentro de toda a ontologia;

b) *Domain Specific Fraud Ontologies* e *Generic Fraud Ontology*: nestas camadas, são expressos os conceitos e as regras referentes a fraudes, definidos pelos especialistas, juntamente com os possíveis tipos de fraudes. Deste modo, enquanto as *Domain Specific Fraud Ontologies* expressam as características gerais de um domínio específico (por exemplo, previdência social), a *Generic Fraud Ontology* fornece um conhecimento mais abstrato e mais genérico relacionado à fraude no domínio especificado;

c) *Case Specific Domain Ontologies*: fornece os conhecimentos básicos sobre os quais as regras definidas irão se basear.


Figura 5 - *Fraud Ontology Layered Architecture*

Nota: elaborado por Alexopoulos (2006)

Para a EHFCN, a metodologia proposta é apresentada no contexto dos serviços de *e-government*, mas pode ser estendida a outros contextos, e procura definir um processo compreendido em três passos (ALEXOPOULOS *et al.*, 2006):

1º) Estabelecimento do contexto de fraudes, envolve a definição de tipos de fraudes que a organização deseja minimizar e a identificação dos processos de negócios sobre os quais elas incidem;

2º) Identificação das fraudes dentro do contexto estabelecido, envolve a descrição dos casos de fraudes potenciais que podem ocorrer na organização e os métodos de detecção correspondentes;

3º) E, transformação das informações identificadas em um modelo ontológico, com base numa arquitetura de três camadas independentes mais interconectadas, cada uma com seu próprio conjunto de ontologias (ver Figura 5).

## 3.7 ONTOLOGIAS APLICADAS ÀS TÉCNICAS PARA DETECÇÃO AUTOMÁTICA

No processo de mapeamento de ontologias para o fornecimento de conhecimentos reutilizáveis nesta dissertação, apenas uma ontologia foi identificada, possuindo conceitos relacionados com a ontologia para técnicas para detecção automática de fraudes.

A *Universal Knowledge Grid* é uma ontologia baseada no modelo de arquitetura em *grid computing*, com o propósito de apoiar a construção de sistemas de conhecimento distribuídos em larga escala sobre *grid*, dando ênfase às aplicações de alto desempenho para descoberta de conhecimentos e serviços de integração, distribuídos geograficamente (LI *et al.*, 2006).

Tem como um dos principais serviços, uma base de conhecimentos contendo uma taxonomia para mineração de dados, conforme demonstrado pela figura 6 (LI *et al.*, 2006).

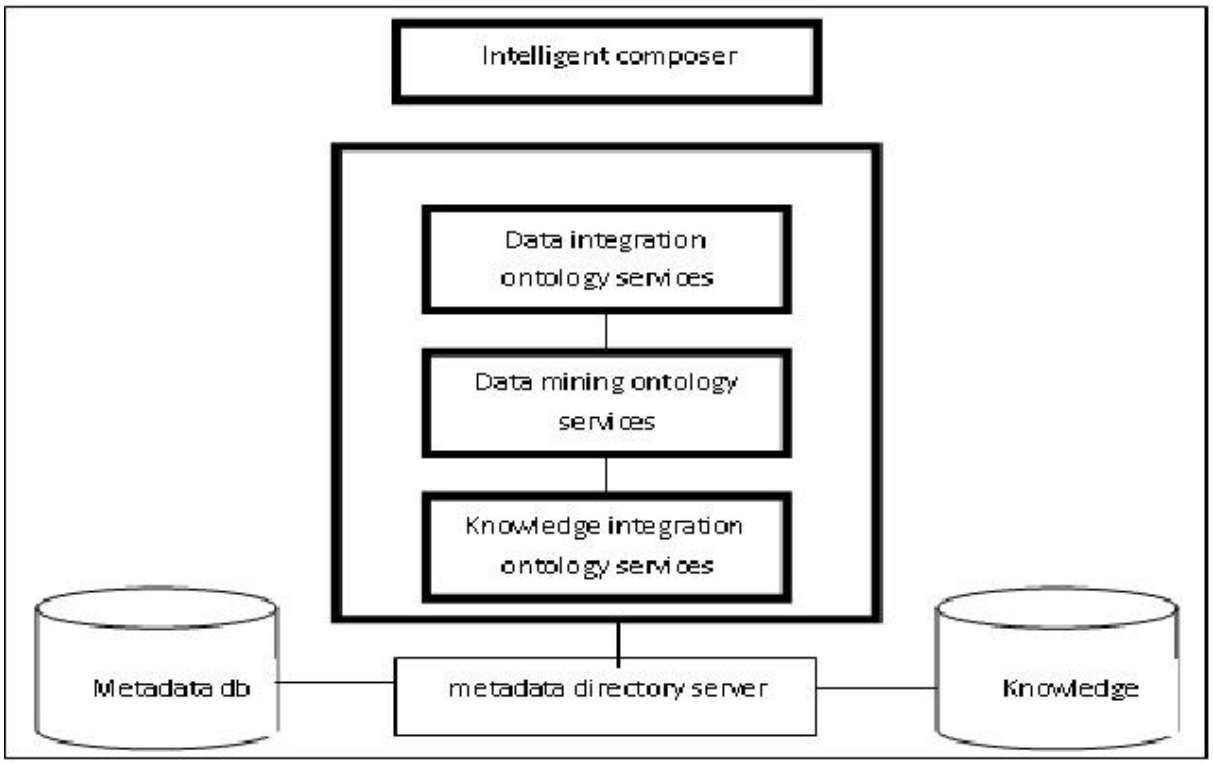


Figura 6 - *The* UKG *Architecture*

Nota: elaborado por Li *et al.* (2006)

A arquitetura da UKG é composta basicamente por três componentes, sendo o primeiro, *intelligent composer*, responsável por fornecer um conjunto de ferramentas gráficas. O segundo componente, *ontology server*, responsável pela gestão de três conjuntos de ontologias (*data integration ontology services*, *data mining ontology services* e *knowledge integration ontology services*) e pela oferta de serviços através modelos semânticos descritos nas respectivas ontologias. E o último componente, *metadata directory server*, é responsável por manter os metadados utilizados na descrição de todos os dados, ferramentas e conhecimentos usados pela UKG.

## 3.8 CONCLUSÃO

O uso de ontologias para atingir o objetivo desta dissertação, teve dois propósitos: *possibilitar o re-uso de conhecimentos de um domínio* e *analisar o conhecimento de um domínio*, permitindo que este conhecimento seja entendido, manipulado e estendido a fim de gerar um novo conhecimento.

Para tanto, estabeleceu-se o desenvolvimento de ontologias de Domínio, utilizando a metodologia do Uschold & King´s *method*, escolhida a partir de uma proposta de extensão

para o *framework* elaborado por Gómez-Pérez *et al.* (2004), voltado à avaliação de metodologias.

O processo de construção contou também, com o uso da linguagem OWL e da ferramenta de apoio, *Protégé* na sua versão 4.0 beta.

Observando o conceito de reusabilidade, foram feitas pesquisas e levantamentos para identificação de ontologias a serem usadas na dissertação. Duas foram identificadas referentes a problemas de fraude (aquela contida no projeto FFPOIROT da Comunidade Econômica Européia e a *Generic Fraud Ontology in e-Government* desenvolvida pela EHFCN), mas não se mostraram adequadas ao uso, seja pela complexidade seja pela falta de dificuldade de acesso ao seu conteúdo.

Em relação ao domínio de técnicas para detecção automática de fraudes, foi identificado um modelo ontológico, chamado UKG, que tem o foco principal, o apoio ao desenvolvimento de sistemas de conhecimento distribuídos, em larga escala, sobre arquitetura de *grid computing*. Uma das ontologias que compõe o seu conjunto possui uma taxonomia para conceitos de mineração de dados. Porém, o seu uso foi comprometido, neste momento, por não está orientada à prioridade estabelecida por esta dissertação e pela impossibilidade de acesso ao seu conteúdo.

# CAPÍTULO 4

## 4 ONTOLOGIAS DESENVOLVIDAS

Este capítulo contém as referências e o processo de construção das seguintes ontologias contidas na dissertação (ver apêndices A e B):

a) “Técnicas para Detecção Automática de Fraude” (*Automatic_Technique_Detection.owl*);

b) “Problemas de Fraude” (*Fraud_Problem.owl*);

c) “Detecção Automática de Fraudes” (*Fraud_Detection.owl*): obtida a partir da atividade de *merging* entre as duas ontologias anteriores e tendo como finalidade atender ao objetivo da dissertação.

Para construção das ontologias, foi aplicada a metodologia Uschold & King´s *method*, o qual prevê o desenvolvimento de modo incremental, oferecendo como esquema básico as seguintes atividades (USCHOLD; KING, 1995; USCHOLD; GRÜNINGER, 1996):

1º) Identificar finalidades (*identify purpose*): isto é, identificar com clareza, porque a ontologia está sendo construída, qual a proposta de uso, quem são os possíveis usuários para a mesma;

2º) Construir ontologia (*building the ontology*):

a. Capturar (*capture*). Envolve as tarefas de especificação e conceitualização para a construção de uma ontologia:

- Definir escopo (*scoping*), através da identificação dos conceitos-chave e relações-chave no domínio de interesse;
- Produzir definições não ambíguas para os conceitos-chave e relações-chave;
- Identificar os termos que se referenciam aos conceitos-chave e relações-chave identificados;
- Construir taxonomia conceitual, a fim de se definir a hierarquia entre os conceitos;

b. Codificar (*coding*): refere-se à representação explícita da especificação e da conceitualização "capturadas", através do uso de uma linguagem de representação e da criação de um código;

c. Integrar ontologias existentes (*integrating existing ontologies*): relaciona-se ao conceito de re-usabilidade de ontologias e às atividades de *merging*, que é a geração de uma nova ontologia a partir de várias ontologias de um mesmo domínio, e de *mapping* ou *alignment*, que significa a geração de mapeamentos ou ligações entre várias ontologias, preservando as ontologias originais e sem gerar uma nova ontologia (NOY; MUSEN, 2000);

3º) Avaliar (*evaluation*), "julgamento técnico das ontologias, a partir da verificação do ambiente de software associado e da documentação contendo os *frames* de referência. *Frames* de referência podem ser especificações de requisitos, de questões de competência, e de modelos do mundo real" (GÓMEZ-PEREZ *et al. apud* USCHOLD; GRÜNINGER, 1996);

4º) Documentar (*documentation*), "a documentação de uma ontologia deve conter, fundamentalmente, informações sobre os principais conceitos, como também, sobre as primitivas relacionadas aos mesmos" (SKUCE *apud* USCHOLD; KING, 1995).

Para construir as ontologias, foram observados os seguintes pontos:

1. Em relação à subatividade "Capturar" contida em "Construir ontologia", foram usadas como fontes de conhecimentos e referências:

a) A experiência do autor da dissertação, como especialista junto às áreas de inteligência (voltada para o combate a fraudes) e de risco operacional no Ministério da Previdência Social, durante o período de 1999 a 2005;

b) O documento "Técnicas para Detecção Automática de Fraudes: Uma Revisão Sistemática" (Anexo I), o qual corresponde a uma revisão sistemática desenvolvida pelo autor, como trabalho didático para a disciplina Engenharia de Software Experimental do Mestrado em Sistemas e Computação/UNIFACS, envolvendo artigos científicos sobre o tema, relacionados às técnicas para detecção automática aplicadas a domínios específicos de fraudes;

c) Consulta à literatura especializada (livros, artigos científicos, teses e dissertações) relacionada aos temas: fraude, técnicas para detecção e para detecção automática de fraudes (Anexo III).

f) A subatividade “Capturar” (conceitualizar ou categorizar, e modelar) foi desenvolvida com base na abordagem *Middle-out*; nesta abordagem a identificação dos conceitos e das relações, e dos respectivos termos a serem usados na ontologia, baseia-se no levantamento de itens considerados mais significativos, para depois completar com conceitos, relações e termos, com menor relevância, mas fundamentais à construção da taxonomia a ser utilizada (USCHOLD; KING, 1995).

g) As ontologias desenvolvidas foram formatadas como sendo do tipo *Lightweight* (ou seja, são compostas basicamente de conceitos ou classes, de atributos ou propriedades e de relações entre conceitos (CORCHO *et al.*, 2006)) e do tipo Semi-formal (expressadas através de linguagens definidas artificialmente e formalmente (GÓMEZ-PÉREZ *et al.*, 2004)).

h) A tabela contendo a lista de termos (ver Apêndice A), identificados a partir da definição de conceitos e relações, foi composta com os seguintes atributos, conforme estrutura sugerida por Gómez-Pérez *et al.* (2004):

(a) Nome: designativo do termo;

(b) Acrônimo: conjunto de letras (pronunciado como uma palavra normal) formado a partir das letras iniciais ou de sílabas de palavras sucessivas, ou das próprias palavras, constituindo uma denominação;

(c) Definição: descrição do significado do termo, observando-se questões de clareza e de ambigüidade;

(d) Tipo: identificação de qual tipo de representação do conhecimento está associada ao termo designado.

i) Para avaliação das ontologias foram usados:

- O algoritmo *Fact plus plus*, desenvolvido pela University of Manchester, implementado em C++ e incorporado através de um *plug-in* à versão 4.0 beta da ferramenta *Protégé*;

- A aplicação *Pellet 1.5*, desenvolvida em Java por Clark e Parsia, e incorporada através de um *plug-in* à versão 4.0 beta da ferramenta *Protégé*;
- Aplicação *DL Query*, desenvolvida pela University of Manchester, incorporada através de um *plug-in* à versão 4.0 beta da ferramenta *Protégé*; foi utilizada para avaliar as definições de classes quanto à sua hierarquia (relação entre classes e subclasses) e quanto à existência de ambigüidades e de não atribuições em relação aos conceitos-chave (PROTÉGÉ, 2005);
- A aplicação *Pellet* e aquela desenvolvida com base no algoritmo *Fact plus plus*, são aplicações do tipo *Reasoner* (isto é, voltadas para análise de seqüências lógicas em bases de conhecimento), acessadas a partir da opção *Classify* no *Protégé* versão 4.0 beta; são aplicadas para validar a correta estruturação da hierarquia de classes das ontologias, verificando se há classes ambíguas e classes não inferidas. Quando é identificado algum problema, o mesmo é indicado através do deslocamento de classes ambíguas ou não inferidas na estrutura, e atribuídas à superclasse *Nothing* (classe OWL pré-definida em modo vazio sem nenhum membro (LACY, 2005)).

## 4.1 TÉCNICAS PARA DETECÇÃO AUTOMÁTICA DE FRAUDES

Esta ontologia foi construída com duas finalidades. A primeira, com o objetivo de atender uma enorme carência quanto à disponibilidade de conhecimentos sobre o tema, técnicas para detecção automática de problemas de fraude. A segunda, para compor uma base de conhecimentos, cuja finalidade é sugerir qual a técnica para detecção automática mais adequada a um problema específico de fraude, considerando o contexto estabelecido para o processo de detecção.

Deste modo, a ontologia sobre Técnicas para Detecção Automática de Fraudes, poderá ser usada para responder as seguintes questões:

a) *Quais são as técnicas para detecção automática existentes para detecção de problemas de fraude*?

b) *Quais são as dimensões de avaliação a serem observadas para análise das técnicas para detecção automática a serem usadas*?

c) *Quais as relações entre dimensões de avaliação, fatores ambientais, e uma técnica para detecção automática específica*?

A resposta a estas questões atenderá tanto aos especialistas e fornecedores de tecnologias, quanto aos especialistas em fraudes, os quais poderão identificar técnicas para detecção automática a serem aplicadas sobre problemas específicos de fraudes.

### 4.1.1 Construir ontologia

A lista de termos (Apêndice A) que compõe esta ontologia foi construída com base na Revisão Estruturada contida no anexo I desta dissertação e na literatura especializada constante no anexo III.

Em relação à Revisão Estruturada, foram efetuados levantamentos e análises de vários domínios de fraudes, procurando identificar as técnicas para detecção automáticas usadas, a freqüência do uso das mesmas por domínio e quais os ambientes de uso destas técnicas. Quanto à literatura especializada, buscaram-se, principalmente, as questões referentes à estrutura conceitual das técnicas identificadas com maior freqüência na Revisão Estruturada, a fim de complementar as definições necessárias aos conceitos e às relações constantes na ontologia.

A lista gerada contém termos referentes a conceitos, relações e instâncias ou indivíduos:

1) Os conceitos (classes) e relações (propriedades) contêm as representações relacionadas ao domínio:
    a) Classe chamada de “Dimensão” (*Dimension*), contendo os termos referentes às dimensões para avaliação de técnicas em relação aos seus contextos de uso. Tem como subclasses:
        - *Engineering: Compactness, Ease_of_Use, Embeddability, Flexibility, Scalability*;
        - *Logistical: Computing_Ease, Development_Time, Independence_from_Experts;*
        - *Quality: Accuracy, Explainability, Response_Time;*

- *Resource: Learning_Curve, Tolerance_for_Complexity, Tolerance_for_Noise_in_Data, Tolerance_for_Sparse_Data;*

*b)* Classe chamada de "Ambiente" (*Environment*), contendo os termos referentes aos fatores ambientais que tem impacto sobre as dimensões. A taxonomia para esta e para a classe anterior foram definidas com base no conceito de *Intelligence Density* (DHAR; STEIN, 1997); ver item 2.4.1. Tem como subclasses: *Amount_of_Data, Business_Understand, Complexity_of_Business_Problem, Complexity_of_Data, Complexity_of_Infrastructure, Data_Understand, Infrastructure_Available, Scrubedd_Data, Update_of_Data;*

c) Classe chamada de "Técnica" (*Technique*), contendo os termos de técnicas para detecção automática de fraudes, sendo que as fontes principais para identificar dos mesmos foram os *surveys* analisados para esta dissertação (ABBOTT *et al.,* 1998; PHUA *et al.,* 2005; BOLTON; HAND, 2002; PHUA, 2003; YUFENG *et al.,* 2004). Ver item 2.5;

d) Classe chamada de "Relação de Dimensão" (*Dimension_Relation*), contendo indivíduos, os quais representam a relação entre dimensões e fatores ambientais que é utilizada para avaliação de uma técnica automática de detecção de fraudes, conforme abordagem proposta por Dhar e Stein (1997).

   Ontologicamente, este tipo de classe é chamado de *reified relation* (ALLEMBERG; HENDLER, 2008; W3C WORKING GROUP, 2006).

   Uma classe do tipo *reified relation* é criada, normalmente, em OWL e RDF, com o objetivo de representar uma situação onde a instância de uma propriedade está ligada a mais de dois indivíduos, configurando desta forma, uma relação do tipo *n-ary* (TUDORACHE, 2004; W3C WORKING GROUP, 2006).

   A classe *Dimension_Relation* é complementada com as propriedades que ligam, de forma binária, os argumentos referentes à classe *Dimension* e à classe *Environment*.

e) Propriedades:

- "*has_relation*", representando a relação entre a classe *Technique* e a classe *Dimension_Relation*;
- Representando a relação entre a classe *Dimension_Relation* e as sub-classes pertencentes às classes *Dimension* e *Environment*: *compactness_value, ease_of_use_value, embeddability_value, flexibility_value, scalability_value, computing_ease_value, development_time_value, independence_from_experts_value, accuracy_value, explainability_value, response_time_value, learning_curve_value, tolerance_for_complexity_value, tolerance_for_noise_in_data_value, tolerance_for_sparse_data_value, amount_of_data_value, complexity_of_business_problem_value, complexity_of_data_value, scrubbed_data_value, update_of_data_value, complexity_of_infrastructure_value, business_understand_value, data_understand_value, infrastructure_available_value.*

2) Indivíduos, referentes:

   a) Às instâncias possíveis para cada uma das subclasses da classe *Dimension*:

   - *Accuracy: Accuracy_High, Accuracy_Moderate e Accuracy_Low*;
   - *Compactness: Compactness_High, Compactness_Moderate, Compactness_Low;*
   - *Computing_Ease: Computing_Ease_High, Computing_Ease_Moderate, Computing_Ease_Low;*
   - *Development_Time: Development_Time_High, Development_Time_Moderate, Development_Time_Low;*
   - *Ease_of_Use: Ease_of_Use_High, Ease_of_Use_Moderate, Ease_of_Use_High;*
   - *Embeddability: Embeddability_High, Embeddability_Moderate, Embeddability_Low;*

- *Explainability: Explainability_High, Explainability_Moderate, Explainability_Low;*
- *Flexibility: Flexibility_High, Flexibility_Moderate, Flexibility_Low;*
- *Independence_from_Experts: Independence_from_Experts_High, Independence_from_Experts_Moderate, Independence_from_Experts_Low;*
- *Learning_Curve: Learning_Curve_High, Learning_Curve_Moderate, Learning_Curve_Low;*
- *Response_Time: Response_Time_High, Response_Time_Moderate, Response_Time_Low;*
- *Scalability: Scalability_High, Scalability_Moderate, Scalability_Low;*
- *Tolerance_for_Complexity: Tolerance_for_Complexity_High, Tolerance_for_Complexity_Moderate, Tolerance_for_Complexity_Low;*
- *Tolerance_for_Noise_in_Data: Tolerance_for_Noise_in_Data_High, Tolerance_for_Noise_in_Data_Moderate, Tolerance_for_Noise_in_Data_Low;*
- *Tolerance_for_Sparse_Data: Tolerance_for_Sparse_Data_High, Tolerance_for_Sparse_Data_Moderate, Tolerance_for_Sparse_Data_Low;*

b) Às instâncias possíveis para cada uma das subclasses da classe *Environment*:

- *Amount_of_Data: Amount_of_Data_Large, Amount_of_Data_Medium, Amount_of_Data_Small;*
- *Business_Understand: Business_Understand_High, Business_Understand_Moderate, Business_Understand_Low;*

- *Complexity_of_Business_Problem: Complexity_of_Business_Problem_High, Complexity_of_Business_Problem_Moderate, Complexity_of_Business_Problem_Low;*
- *Complexity_of_Data: Complexity_of_Data_High, Complexity_of_Data_Moderate, Complexity_of_Data_Low;*
- *Complexity_of_Infrastructure: Complexity_of_Infrastructure_High, Complexity_of_Infrastructure_Moderate, Complexity_of_Infrastructure_Low;*
- *Data_Understand: Data_Understand_High, Data_Understand_Moderate, Data_Understand_Low;*
- *Infrastructure_Available: Infrastructure_Available_Large, Infrastructure_Available_Medium, Infrastructure_Available_Small;*
- *Scrubbed_Data: Scrubbed_Data_No, Scrubbed_Data_Yes;*
- *Update_of_Data: Update_of_Data_Out_of_Data, Update_of_Data_Recent;*

c) Às instâncias possíveis para a classe *Technique*: *CBR* (Raciocínio Baseado em Casos ou *Case Based Reasoning*), *DDDS* (Dados Organizados para Tomada de Decisão ou *Data Driven Decision Support*), *FL* (Lógica *Fuzzy* ou *Fuzzy Logic*), *GA* (Algoritmo Genético ou *Genetic Algorithm*), *ML* (Aprendizagem de Máquina/Algoritmo de Particionamento Recursivo ou *Machine Learning/Recursive Partitioning Algorithm*), *NN* (Redes Neurais ou *Neural Networks*), *RBS* (Raciocínio Baseado em Regras ou *Rule Based System*).
Estas instâncias foram estabelecidas, considerando as técnicas automáticas para detecção de problemas de fraude que tiveram o maior número de registros, de acordo com os *surveys* (ABBOTT *et al.*, 1998; PHUA *et al.*, 2005; BOLTON; HAND, 2002; PHUA, 2003; YUFENG *et al.*, 2004) e as outras fontes utilizadas na pesquisa para esta dissertação;

d) Às instâncias possíveis para a classe *Dimension_Relation*, seguem da *Dimension_Relation_01* até *Dimension_Relation_29*.

As taxonomias presentes na ontologia são:

a) Taxonomia-base contendo os principais grupos de classes (Figura 7):

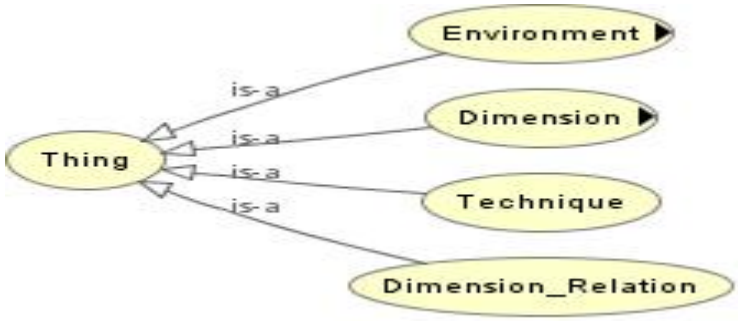


Figura 7 - Taxonomia-base para ontologia “Técnicas para Detecção Automática de Fraudes”.
Nota: elaboração própria

b) Taxonomia para a classe *Dimension* e suas subclasses (Figura 8), definindo as dimensões para avaliação de técnicas, conforme abordagem baseada no conceito de *Intelligence Density* elaborado por Dhar e Stein (1997); ver item 2.4.1:


Figura 8 - Taxonomia para a classe *Dimension* e suas subclasses
Nota: elaboração própria

c) Taxonomia para a classe *Environment* e suas subclasses, definindo os fatores ambientais que tem impacto sobre as dimensões para avaliação de técnicas, conforme abordagem baseada no conceito de *Intelligence Density* (DHAR; STEIN, 1997); ver item 2.4.1:

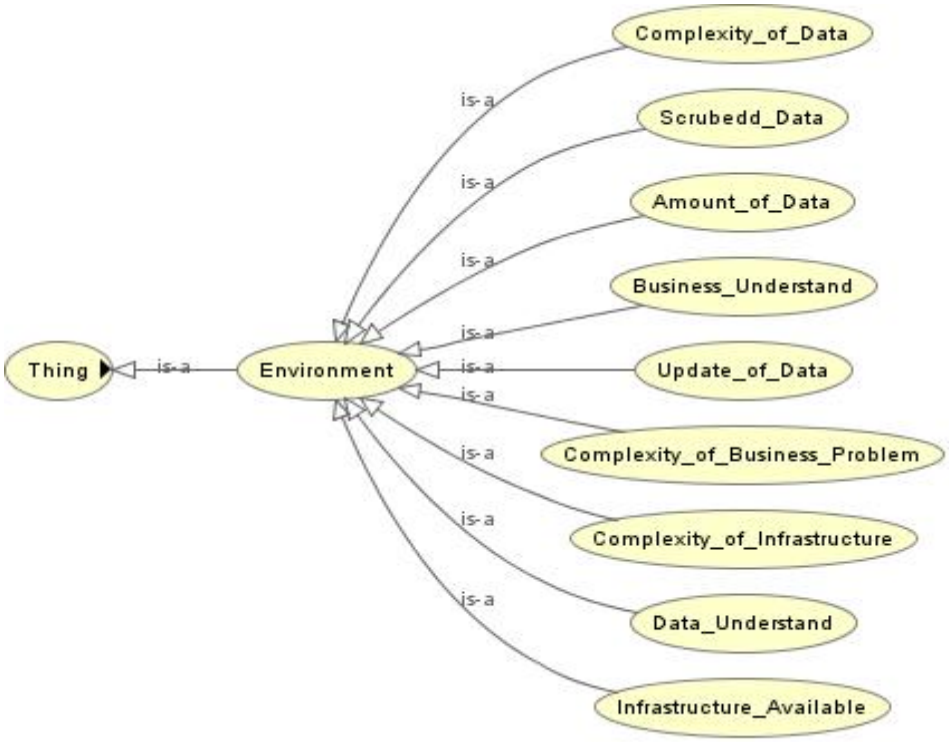


Figura 9 - Taxonomia para a classe *Environment* e suas subclasses

Nota: elaboração própria.

Os principais axiomas contidos nesta ontologia são:

a) Para que uma técnica baseada em Raciocínio Baseado em Regras (RBS) tenha tempo baixo de desenvolvimento para uma amostra de dados, isto só é possível, se esta amostra possuir quantidade pequena ou média de dados.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma a):

Restrição necessária (em linguagem natural):
*Uma relação Dimension_Relation_28 (*instância de *Dimension_Relation) possui a dimensão, tempo baixo de desenvolvimento (*instância de *Dimension), e os fatores ambientais, amostra com quantidade pequena de dados (instância de Environment) ou quantidade média de dados (*instância de *Environment).*

Axioma que define a restrição necessária:
*Dimension_Relation_28*
*and development_time_value value Development_Time_Low*
*and (amount_of_data_value value Amount_of_Data_Small*

*or amount_of_data_value value Amount_of_Data_Medium).*

2º. Estabelecer a relação entre a técnica RBS com a relação de dimensão Dimension_Relation_28:

Restrição necessária (em linguagem natural):
*Se uma técnica RBS (*instância de *Technique) tem tempo baixo de desenvolvimento para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_28 (*instância de *Dimension_Relation).*

Axioma que define a restrição necessária:
*RBS*
*and has_relation value Dimension_Relation_28.*

b) Uma técnica baseada em Raciocínio Baseado em Regras (RBS) gera resposta com explicabilidade alta para uma amostra de dados.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma b):

Restrição necessária (em linguagem natural):
*Uma relação Dimension_Relation_29 (*instância de *Dimension_Relation) possui a dimensão, explicabilidade alta (*instância de *Dimension).*

Axioma que define a restrição necessária:
*Dimension_Relation_29*
*and explainability_value value Explainability_High.*

2º. Estabelecer a relação entre a técnica RBS com a relação de dimensão Dimension_Relation_29:

Restrição necessária (em linguagem natural):
*Se uma técnica RBS (*instância de *Technique) gera resposta com explicabilidade alta para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_29 (*instância de *Dimension_Relation).*

Axioma que define a restrição necessária:

*RBS*
*and has_relation value Dimension_Relation_29.*

c) Uma técnica baseada em Dados Organizados para Tomada de Decisão (DDDS) possui facilidade computacional alta para uma amostra de dados.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma c):

Restrição necessária (em linguagem natural):
*Uma relação Dimension_Relation_06 (instância de Dimension_Relation) possui a dimensão, facilidade computacional alta (instância de Dimension).*

Axioma que define a restrição necessária:
*Dimension_Relation_06*
*and computing_ease_value value Computing_Ease_High.*

2º. Estabelecer a relação entre a técnica DDDS com a relação de dimensão Dimension_Relation_06:

Restrição necessária (em linguagem natural):
*Se uma técnica DDDS (instância de Technique) possui facilidade computacional alta para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_06 (Dimension_Relation).*

Axioma que define a restrição necessária:
*DDDS*
*and has_relation value Dimension_Relation_06.*

d) Para que uma técnica baseada em Dados Organizados para Tomada de Decisão (DDDS) apresente facilidade alta de uso para uma amostra de dados, foi definido que isto só é possível, se esta amostra for analisada sobre infra-estrutura com complexidade baixa ou moderada.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma d):

Restrição necessária (em linguagem natural):

*Uma relação Dimension_Relation_07 (instância de Dimension_Relation) possui a dimensão, facilidade alta de uso (instância de Dimension), e os fatores ambientais, infra-estrutura com complexidade baixa (instância de Environment) ou uma infra-estrutura com complexidade moderada (instância de Environment).*

Axioma que define a restrição necessária:
*Dimension_Relation_07*
*and ease_of_use value Ease_of_Use_High*
*and (complexity_of_infrastructure_value value Complexity_of_Infrastructure_Low*
*or complexity_of_infrastructure_value value Complexity_of_Infrastructure_Moderate).*

2º. Estabelecer a relação entre a técnica DDDS com a relação de dimensão Dimension_Relation_07:

Restrição necessária (em linguagem natural):
*Se uma técnica DDDS (instância de Technique) apresenta facilidade alta de uso para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_07 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:
*DDDS*
*and has_relation value Dimension_Relation_07.*

e) Uma técnica baseada em Dados Organizados para Tomada de Decisão (DDDS) possui flexibilidade alta para uma amostra de dados.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma e):

Restrição necessária (em linguagem natural):
*Uma relação Dimension_Relation_02 (instância de Dimension_Relation) possui a dimensão, flexibilidade alta (instância de Dimension).*

Axioma que define a restrição necessária:
*Dimension_Relation_02*

*and flexibility_value value Flexibility_High.*

2º. Estabelecer a relação entre a técnica DDDS com a relação de dimensão Dimension_Relation_02:

Restrição necessária (em linguagem natural):
*Se uma técnica DDDS (instância de Technique) possui flexibilidade alta para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_02 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:
*DDDS*
*and has_relation value Dimension_Relation_02.*

f) Uma técnica baseada em Dados Organizados para Tomada de Decisão (DDDS) gera resultados com tempo baixo de resposta para uma amostra de dados.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma f):

Restrição necessária (em linguagem natural):
*Uma relação Dimension_Relation_08 (instância de Dimension_Relation) possui a dimensão, resultados com tempo baixo de resposta (instância de Dimension).*

Axioma que define a restrição necessária:
*Dimension_Relation_08*
*and response_time_value value Response_Time_Low.*

2º. Estabelecer a relação entre a técnica DDDS com a relação de dimensão Dimension_Relation_08:

Restrição necessária (em linguagem natural):
*Se uma técnica DDDS (instância de Technique) gera respostas com tempo baixo de resposta para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_08 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:

*DDDS*
*and has_relation value Dimension_Relation_08.*

g) Para que uma técnica baseada em Raciocínio Baseado em Casos (CBR) gere resultados com precisão alta para uma amostra de dados, foi definido que isto só é possível, se esta amostra possuir quantidade grande de dados.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma g):

Restrição necessária (em linguagem natural):
*Uma relação Dimension_Relation_01 (instância de Dimension_Relation) possui a dimensão, resultados gerados com alta precisão (instância de Dimension), e o fator ambiental, amostra com quantidade grande de dados (instância de Environment).*

Axioma que define a restrição necessária:
*Dimension_Relation_01*
*and accuracy_value value Accuracy_High*
*and amount_of_data_value value Amount_of_Data_Large.*

2º. Estabelecer a relação entre a técnica DDDS com a relação de dimensão Dimension_Relation_01:

Restrição necessária (em linguagem natural):
*Se uma técnica CBR (instância de Technique) gera resultados com precisão alta para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_01 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:
*CBR*
*and has_relation value Dimension_Relation_01*.

h) Uma técnica baseada em Raciocínio Baseado em Casos (CBR) possui flexibilidade alta para uma amostra de dados.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma h):

Restrição necessária (em linguagem natural):
*Uma relação Dimension_Relation_02 (instância de Dimension_Relation) possui a dimensão, flexibilidade alta (instância de Dimension).*

Axioma que define a restrição necessária:
*Dimension_Relation_02*
*and flexibility_value value Flexibility_High.*

2º. Estabelecer a relação entre a técnica CBR com a relação de dimensão Dimension_Relation_02:

Restrição necessária (em linguagem natural):
*Se uma técnica CBR (instância de Technique) possui flexibilidade alta para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_02 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:
*CBR*
*and has_relation value Dimension_Relation_02.*

i) Para que uma técnica baseada em Raciocínio Baseado em Casos (CBR) possua independência alta de especialistas para uma amostra de dados, foi definido que isto só é possível, se esta amostra representar um problema de negócio com complexidade baixa ou moderada.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma i):

Restrição necessária (em linguagem natural):
*Uma relação Dimension_Relation_03 (instância de Dimension_Relation) possui a dimensão, independência alta de especialistas (instância de Dimension), e os fatores ambientais, problema de negócio com complexidade baixa (instância de Environment) ou problema de negócio com complexidade moderada (instância de Environment).*

Axioma que define a restrição necessária:
*Dimension_Relation_03*

*and independence_from_experts_value value Independence_from_Experts_High*
*and (complexity_of_business_problem_value value Complexity_of_Business_Problem_Low or complexity_of_business_problem_value value Complexity_of_Business_Problem_Moderate).*

2º. Estabelecer a relação entre a técnica CBR com a relação de dimensão Dimension_Relation_03:

Restrição necessária (em linguagem natural):
*Se uma técnica CBR (instância de Technique) possui independência alta de especialistas para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_03 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:
*CBR*
*and has_relation value Dimension_Relation_03.*

j) Para que uma técnica baseada em Raciocínio Baseado em Casos (CBR) gere resultados com tempo baixo de resposta para uma amostra de dados, foi definido que isto só é possível, se esta amostra for analisada sobre infra-estrutura disponível grande e representar problema de negócio com complexidade baixa ou moderada e possuir quantidade pequena ou média de dados.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma j):

Restrição necessária (em linguagem natural):
*Uma relação Dimension_Relation_04 (instância de Dimension_Relation) possui a dimensão, tempo baixo de resposta (instância de Dimension), e os fatores ambientais, infra-estrutura grande disponível (instância de Environment), problema de negócio com complexidade baixa (instância de Environment) ou problema de negócio com complexidade moderada (instância de Environment), e quantidade pequena de dados (instância de Environment) ou quantidade média de dados (instância de Environment).*

Axioma que define a restrição necessária:
*Dimension_Relation_04*
*and response_time_value value Response_Time_Low*
*and infrastructure_available_value value Infrastructure_Available_Large*
*and (complexity_of_business_problem_value value Complexity_of_Business_Problem_Low or complexity_of_business_problem_value value Complexity_of_Business_Problem_Moderate)*
*and (amount_of_data_value value Amount_of_Data_Small*

*or amount_of_data_value value Amount_of_Data_Medium).*

2º. Estabelecer a relação entre a técnica CBR com a relação de dimensão Dimension_Relation_04:

Restrição necessária (em linguagem natural):
*Se uma técnica CBR (instância de Technique) gera resultados com baixo tempo de resposta para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_04 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:
*CBR*
*and has_relation value Dimension_Relation_04.*

k) Uma técnica baseada em Raciocínio Baseado em Casos (CBR) permite escalabilidade alta para uma amostra de dados.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma k):

Restrição necessária (em linguagem natural):
*Uma relação Dimension_Relation_05 (instância de Dimension_Relation) possui a dimensão, escalabilidade alta (instância de Dimension).*

Axioma que define a restrição necessária:
*Dimension_Relation_05*
*and scalability_value value Scalability_High.*

2º. Estabelecer a relação entre a técnica CBR com a relação de dimensão Dimension_Relation_05:

Restrição necessária (em linguagem natural):
*Se uma técnica CBR (instância de Technique) permite escalabilidade alta para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_05 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:
*CBR*
*and has_relation value Dimension_Relation_05.*

l) Para que uma técnica baseada em Redes Neurais (NN) gere resultados com precisão alta para uma amostra de dados, foi definido que isto só é possível, se esta amostra representar um problema de negócio com complexidade baixa ou moderada.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma l):

Restrição necessária (em linguagem natural):
*Uma relação Dimension_Relation_22 (instância de Dimension_Relation) possui a dimensão, precisão alta (instância de Dimension), e os fatores ambientais, problema de negócio com complexidade baixa (intância de Environment) ou problema de negócio com complexidade moderada (intância de Environment).*

Axioma que define a restrição necessária:
*Dimension_Relation_22*
*and accuracy_value value Accuracy_High*
*and (complexity_of_business_problem_value value Complexity_of_Business_Problem_Low or complexity_of_business_problem_value value Complexity_of_Business_Problem_Moderate).*

2º. Estabelecer a relação entre a técnica NN com a relação de dimensão Dimension_Relation_22:

Restrição necessária (em linguagem natural):
*Se uma técnica NN (instância de Technique) gera resultados com precisão alta para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_22 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:
*NN*
*and has_relation value Dimension_Relation_22.*

m) Uma técnica baseada em Redes Neurais (NN) possui compacidade alta para uma amostra de dados.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma m):

Restrição necessária (em linguagem natural):
*Uma relação Dimension_Relation_10 (instância de Dimension_Relation) possui a dimensão, compacidade alta (instância de Dimension).*

Axioma que define a restrição necessária:
*Dimension_Relation_10*
*and compactness_value value Compactness_High.*

2º. Estabelecer a relação entre a técnica NN com a relação de dimensão Dimension_Relation_10:

Restrição necessária (em linguagem natural):
*Se uma técnica NN (instância de Technique) possui compacidade alta para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_10 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:
*NN*
*and has_relation value Dimension_Relation_10.*

n) Uma técnica baseada em Redes Neurais (NN) permite encapsulamento alto para uma amostra de dados.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma n):

Restrição necessária (em linguagem natural):
*Uma relação Dimension_Relation_18 (instância de Dimension_Relation) possui a dimensão, encapsulamento alto (instância de Dimension).*

Axioma que define a restrição necessária:
*Dimension_Relation_18*
*and embeddability_value value Embeddability_High.*

2º. Estabelecer a relação entre a técnica NN com a relação de dimensão Dimension_Relation_18:

Restrição necessária (em linguagem natural):
*Se uma técnica NN (instância de Technique) permite encapsulamento alto para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_18 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:
*NN*
*and has_relation value Dimension_Relation_18.*

o) Para que uma técnica baseada em Redes Neurais (NN) possua flexibilidade alta para uma amostra de dados, foi definido que isto só é possível, se esta amostra permitir um entendimento alto dos dados.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma o):

Restrição necessária (em linguagem natural):
*Uma relação Dimension_Relation_23 (instância de Dimension_Relation) possui a dimensão, flexibilidade alta (instância de Dimension), e o fator ambiental, entendimento alto dos dados (instância de Environment).*

Axioma que define a restrição necessária:
*Dimension_Relation_23*
*and flexibility_value value NN_Flexibility_High*
*and data_understand_value value Data_Understand_High.*

2º. Estabelecer a relação entre a técnica NN com a relação de dimensão Dimension_Relation_23:

Restrição necessária (em linguagem natural):
*Se uma técnica NN (instância de Technique) possua flexibilidade alta para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_23 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:
*NN*
*and has_relation value NN_Flexibility_High.*

p) Uma técnica baseada em Redes Neurais (NN) permite independência alta de especialistas para uma amostra de dados.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma p):

Restrição necessária (em linguagem natural):
*Uma relação Dimension_Relation_24 (instância de Dimension_Relation) possui a dimensão, independência alta de especialistas (instância de Dimension).*

Axioma que define a restrição necessária:
*Dimension_Relation_24*
*and independence_from_experts_value value Independence_from_Experts_High.*

2º. Estabelecer a relação entre a técnica NN com a relação de dimensão Dimension_Relation_24:

Restrição necessária (em linguagem natural):
*Se uma técnica NN (instância de Technique) permite independência alta de especialistas para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_24 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:
*NN*
*and has_relation value Dimension_Relation_24.*

q) Uma técnica baseada em Redes Neurais (NN) gera resultados com tempo baixo de resposta para uma amostra de dados.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma q):

Restrição necessária (em linguagem natural):
*Uma relação Dimension_Relation_08 (instância de Dimension_Relation) possui a dimensão, tempo baixo de resposta (instância de Dimension).*

Axioma que define a restrição necessária:
*Dimension_Relation_08*
*and response_time_value value Response_Time_Low.*

2º. Estabelecer a relação entre a técnica NN com a relação de dimensão Dimension_Relation_08:

Restrição necessária (em linguagem natural):
*Se uma técnica NN (instância de Technique) gera resultados com baixo tempo de resposta para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_08 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:
*NN*
*and has_relation value Dimension_Relation_08.*

r) Uma técnica baseada em Redes Neurais (NN) possui tolerância alta à complexidade para uma amostra de dados.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma r):

Restrição necessária (em linguagem natural):
*Uma relação Dimension_Relation_25 (instância de Dimension_Relation) possui a dimensão, tolerância alta à complexidade (instância de Dimension).*

Axioma que define a restrição necessária:
*Dimension_Relation_25*
*and tolerance_for_complexity_value value Tolerance_for_Complexity_High.*

2º. Estabelecer a relação entre a técnica NN com a relação de dimensão Dimension_Relation_25:

Restrição necessária (em linguagem natural):
*Se uma técnica NN (instância de Technique) possui tolerância alta à complexidade para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_25 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:
*NN*
*and has_relation value Dimension_Relation_25.*

s) Para que uma técnica baseada em Redes Neurais (NN) possua tolerância alta aos dados inconsistentes para uma amostra de dados, foi definido que isto só é possível, se esta amostra permitir um entendimento moderado ou alto dos dados.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma s):

Restrição necessária (em linguagem natural):
*Uma relação Dimension_Relation_26 (instância de Dimension_Relation) possui a dimensão, tolerância alta aos dados inconsistentes (instância de Dimension), e os fatores ambientais, entendimento moderado dos dados (instância de Environment) ou entendimento alto dos dados (instância de Environment).*

Axioma que define a restrição necessária:
*Dimension_Relation_26*
*and tolerance_for_noise_in_data_value value Tolerance_for_Noise_in_Data_High*
*and (data_understand_value value Data_Understand_Moderate or data_understand_value value Data_Understand_High).*

2º. Estabelecer a relação entre a técnica NN com a relação de dimensão Dimension_Relation_26:

Restrição necessária (em linguagem natural):

*Se uma técnica NN (instância de Technique) possui tolerância alta aos dados inconsistentes para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_26 (instância de Technique_Dimension_Relation).*

Axioma que define a restrição necessária:
*NN*
*and has_relation value Dimension_Relation_26.*

t) Para que uma técnica baseada em Redes Neurais (NN) possua tolerância alta aos dados incompletos para uma amostra de dados, foi definido que isto só é possível, se esta amostra permitir entendimento moderado ou alto dos dados.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma t):

Restrição necessária (em linguagem natural):
*Uma relação Dimension_Relation_27 (instância de Dimension_Relation) possui a dimensão, tolerância alta aos dados incompletos (instância de Dimension), e os fatores ambientais, entendimento moderado dos dados (instância de Environment) ou entendimento alto dos dados (instância de Environment).*

Axioma que define a restrição necessária:
*Dimension_Relation_27*
*and tolerance_for_sparse_data_value value Tolerance_for_Sparse_Data_High*
*and (data_understand_value value Data_Understand_Moderate or data_understand_value value Data_Understand_High).*

2º. Estabelecer a relação entre a técnica NN com a relação de dimensão Dimension_Relation_27:

Restrição necessária (em linguagem natural):

*Se uma técnica NN (instância de Technique) possui tolerância alta aos dados incompletos para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_27 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:
*NN*
*and has_relation value Dimension_Relation_27.*

u) Para que uma técnica baseada em Algoritmo Genético (GA) gere resultados com precisão alta para uma amostra de dados, foi definido que isto só é possível, se esta amostra representar problema de negócio com complexidade baixa.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma u):

Restrição necessária (em linguagem natural):
*Uma relação é Dimension_Relation_12 (instância de Relation) possui a dimensão, precisão alta (instância de Dimension), e o fator ambiental, problema de negócio com complexidade baixa (instância de Environment).*

Axioma que define a restrição necessária:
*Dimension_Relation_12*
*and accuracy_value value Accuracy_High*
*and complexity_of_business_problem_value value Complexity_of_Business_Problem_Low.*

2º. Estabelecer a relação entre a técnica GA com a relação de dimensão Dimension_Relation_12:

Restrição necessária (em linguagem natural):
*Se uma técnica GA (instância de Technique) gera resultados com precisão alta para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_12 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:
*GA*
*and has_relation value Dimension_Relation_12.*

v) Para que uma técnica baseada em Algoritmo Genético (GA) possua tempo baixo de desenvolvimento para uma amostra de dados, foi definido que isto só é possível, se esta amostra representar problema de negócio com complexidade baixa ou moderada.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma v):

Restrição necessária (em linguagem natural):
*Uma relação Dimension_Relation_13 (instância de Dimension_Relation) possui a dimensão, tempo baixo de desenvolvimento (instância de Dimension), e os fatores ambientais, problema de negócio com complexidade baixa (instância de Environment) ou problema de negócio com complexidade moderada (instância de Environment).*

Axioma que define a restrição necessária:
*Dimension_Relation_13*
*and development_time_value value Development_Time_Low*
*and (complexity_of_business_problem_value value Complexity_of_Business_Problem_Low or complexity_of_business_problem_value value Complexity_of_Business_Problem_Moderate).*

2º. Estabelecer a relação entre a técnica GA com a relação de dimensão Dimension_Relation_13:

Restrição necessária (em linguagem natural):
*Se uma técnica GA (instância de Technique) possui tempo baixo de desenvolvimento para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_13 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:
*GA*
*and has_relation value Dimension_Relation_13.*

w) Para que uma técnica baseada em Algoritmo Genético (GA) permita encapsulamento alto para uma amostra de dados, foi definido que isto só é possível, se esta amostra for analisada sobre infra-estrutura com complexidade baixa ou moderada e representar problema de negócio com complexidade baixa ou moderada.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma w):

Restrição necessária (em linguagem natural):
*Uma relação Dimension_Relation_14 (instância de Dimension_Relation) possui a dimensão, encapsulamento alto (instância de Dimension), e os fatores ambientais, infra-estrutura com complexidade baixa (instância de Environment) ou infra-estrutura com complexidade moderada (instância de Environment) e, problema de negócio com complexidade baixa (instância de Environment) ou problema de negócio com complexidade moderada (instância de Environment).*

Axioma que define a restrição necessária:
*Dimension_Relation_14*
*and embeddability _value value Embeddability_High*
*and (complexity_of_infrastructure_value value Complexity_of_Infrastructure_Low or complexity_of_infrastructure_value value Complexity_of_Infrastructure_Moderate)*
*and (complexity_of_business_problem_value value Complexity_of_Business_Problem_Low or complexity_of_business_problem_value value Complexity_of_Business_Problem_Moderate).*

2º. Estabelecer a relação entre a técnica GA com a relação de dimensão Dimension_Relation_14:

Restrição necessária (em linguagem natural):
*Se uma técnica GA (instância de Technique) permite encapsulamento alto para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_14 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:
*GA*
*and has_relation value Dimension_Relation_14.*

x) Para que uma técnica baseada em Algoritmo Genético (GA) possua flexibilidade alta para uma amostra de dados, foi definido que isto só é possível, se esta amostra for analisada sobre infra-estrutura com complexidade baixa ou moderada e possua dados com complexidade baixa ou moderada.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma x):

Restrição necessária (em linguagem natural):
*Uma relação Dimension_Relation_15 (instância de Dimension_Relation) possua a dimensão, é necessário que ela possua flexibilidade alta (instância de Dimension), e os fatores ambientais, infra-estrutura com complexidade baixa (instância de Environment) ou infra-estrutura com complexidade moderada (instância de Environment) e, dados com complexidade baixa (instância de Environment) ou dados com complexidade moderada (instância de Environment).*

Axioma que define a restrição necessária:
*Dimension_Relation_15*
*and flexibility_value value Flexibility_High*
*and (complexity_of_infrastructure_value value Complexity_of_Infrastructure_Low or complexity_of_infrastructure_value value Complexity_of_Infrastructure_Moderate)*
*and (complexity_of_data_value value Complexity_of_Data_Low or complexity_of_data_value value Complexity_of_Data_Moderate).*

2º. Estabelecer a relação entre a técnica GA com a relação de dimensão Dimension_Relation_15:

Restrição necessária (em linguagem natural):
*Se uma técnica GA (instância de Technique) possui flexibilidade alta para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_15 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:

*GA*

*and has_relation value Dimension_Relation_15.*

y) Para que uma técnica baseada em Algoritmo Genético (GA) gere resultado com baixo tempo de resposta para uma amostra de dados, foi definido que isto só é possível, se esta amostra representar problema de negócio com complexidade baixa.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma y):

Restrição necessária (em linguagem natural):

*Uma relação Dimension_Relation_16 (instância de Dimension_Relation) possui a dimensão, tempo baixo de resposta (instância de Dimension), e o fator ambiental, problema de negócio com complexidade baixa (instância de Environment).*

Axioma que define a restrição necessária:

*Dimension_Relation_16*

*and response_time_value value Response_Time_Low*

*and complexity_of_business_problem_value value Complexity_of_Business_Problem_Low.*

2º. Estabelecer a relação entre a técnica GA com a relação de dimensão Dimension_Relation_16:

Restrição necessária (em linguagem natural):

*Se uma técnica GA (instância de Technique) gera resultado com tempo baixo de resposta para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_16 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:

*GA*

*and has_relation value Dimension_Relation_16.*

z) Para que uma técnica baseada em Lógica Fuzzy (FL) gere resultados com precisão alta para uma amostra de dados, foi definido que isto só é possível, se esta amostra permitir entendimento alto do negócio.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma z):

Restrição necessária (em linguagem natural):
*Uma relação Dimension_Relation_09 (instância de Dimension_Relation) possui a dimensão, precisão alta (instância de Dimension), e o fator ambiental, entendimento alto do negócio (instância de Environment).*

Axioma que define a restrição necessária:
*Dimension_Relation_09*
*and accuracy_value value Accuracy_High*
*and business_understand_value value Business_Understand_High.*

2º. Estabelecer a relação entre a técnica FL com a relação de dimensão Dimension_Relation_09:

Restrição necessária (em linguagem natural):
*Se uma técnica FL (instância de Technique) gera resultados com precisão alta para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_09 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:
*FL*
*and has_relation value Dimension_Relation_09.*

aa) Uma técnica baseada em Lógica Fuzzy (FL) possui compacidade alta para uma amostra de dados.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma aa):

Restrição necessária (em linguagem natural):
*Uma relação Dimension_Relation_10 (instância de Dimension_Relation) possui a dimensão, compacidade alta (instância de Dimension).*

Axioma que define a restrição necessária:

*Dimension_Relation_10*

*and compactness_value value Compactness_High.*

2º. Estabelecer a relação entre a técnica FL com a relação de dimensão Dimension_Relation_10:

Restrição necessária (em linguagem natural):

*Se uma técnica FL (instância de Technique) possui compacidade alta para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_10 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:

*FL*

*and has_relation value Dimension_Relation_10.*

bb) Uma técnica baseada em Lógica Fuzzy (FL) possui flexibilidade alta para uma amostra de dados. Deste modo, temos:

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma bb):

Restrição necessária (em linguagem natural):

*Uma relação Dimension_Relation_02 (instância de Dimension_Relation) possui a dimensão, flexibilidade alta (instância de Dimension).*

Axioma que define a restrição necessária:

*Dimension_Relation_02*

*and flexibility_value value Flexibility_High.*

2º. Estabelecer a relação entre a técnica FL com a relação de dimensão Dimension_Relation_02

Restrição necessária (em linguagem natural):

*Se uma técnica FL (instância de Technique) possui flexibilidade alta para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_02 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:

*FL*

*and has_relation value Dimension_Relation_02.*

cc) Uma técnica baseada em Lógica Fuzzy (FL) gera resultado com tempo baixo de resposta para uma amostra de dados.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma cc):

Restrição necessária (em linguagem natural):
*Uma relação Dimension_Relation_08 (instância de Dimension_Relation) possui a dimensão, tempo baixo de resposta (instância de Dimension).*

Axioma que define a restrição necessária:
*Dimension_Relation_08*

*and response_time_value value Response_Time_Low.*

2º. Estabelecer a relação entre a técnica FL com a relação de dimensão Dimension_Relation_08:

Restrição necessária (em linguagem natural):
*Se uma técnica FL (instância de Technique) gera resultado com tempo baixo de resposta para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_08 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:
*FL*

*and has_relation value Dimension_Relation_08.*

dd) Para que uma técnica baseada em Lógica Fuzzy (FL) possua tolerância alta à complexidade para uma amostra de dados, foi definido que isto só é possível, se esta amostra possuir dados com complexidade baixa ou moderada e é analisada sobre infra-estrutura com complexidade baixa ou moderada.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma dd):

Restrição necessária (em linguagem natural):

*Uma relação Dimension_Relation_11 (instância de Dimension_Relation) possui a dimensão, tolerância alta à complexidade (instância de Dimension), e os fatores ambientais, dados com complexidade baixa (instância de Environment) ou dados com complexidade moderada (instância de Environment), e infra-estrutura com complexidade baixa (instância de Environment) ou infra-estrutura com complexidade moderada (instância de Environment).*

Axioma que define a restrição necessária:
*Dimension_Relation_11*
*and tolerance_for_complexity_value value Tolerance_for_Complexity_High*
*and (complexity_of_data_value value Complexity_of_Data_Low or complexity_of_data_value value Complexity_of_Data_Moderate)*
*and (complexity_of_infrastructure_value value Complexity_of_Infrastructure_Low or complexity_of_infrastructure_value value Complexity_of_Infrastructure_Moderate).*

2º. Estabelecer a relação entre a técnica FL com a relação de dimensão Dimension_Relation_11:

Restrição necessária (em linguagem natural):
*Se uma técnica FL (instância de Technique) possui tolerância alta à complexidade para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_11 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:
*FL*
*and has_relation value Dimension_Relation_11*.

ee) Para que uma técnica baseada em Aprendizagem de Máquina/Algoritmo de Particionamento Recursivo (ML) gere resultado com precisão alta para uma amostra de dados, foi definido que isto só é possível, se esta amostra for analisada sobre infra-estrutura com complexidade baixa ou moderada e possuir dados com complexidade baixa ou moderada.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma ee):

Restrição necessária (em linguagem natural):
*Uma relação Dimension_Relation_17 (instância de Dimension_Relation) possui a dimensão, precisão alta (instância de Dimension), e os fatores ambientais, infra-estrutura com complexidade baixa (instância de Environment) ou infra-estrutura com complexidade moderada (instância de Environment), e dados com complexidade baixa (instância de Environment) ou dados com complexidade moderada (instância de Environment).*

Axioma que define a restrição necessária:
*Dimension_Relation_17*
*and accuracy_value value Accuracy_High*
*and (complexity_of_infrastructure_value value Complexity_of_Infrastructure_Low or complexity_of_infrastructure_value value Complexity_of_Infrastructure_Moderate)*
*and (complexity_of_data_value value Complexity_of_Data_Low or complexity_of_data_value value Complexity_of_Data_Moderate).*

2º. Estabelecer a relação entre a técnica ML com a relação de dimensão Dimension_Relation_17:

Restrição necessária (em linguagem natural):
*Se uma técnica ML (instância de Technique) gera resultado com precisão alta para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_17 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:
*ML*
*and has_relation value Dimension_Relation_17.*

ff) Uma técnica baseada em Aprendizagem de Máquina/Algoritmo de Particionamento Recursivo (ML) permite encapsulamento alto para uma amostra de dados.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma ff):

Restrição necessária (em linguagem natural):

*Uma relação Dimension_Relation_18 (instância de Dimension_Relation) possui a dimensão, encapsulamenmto alto (instância de Dimension).*

Axioma que define a restrição necessária:
*Dimension_Relation_18*
*and embeddability_value value Embeddability_High.*

2º. Estabelecer a relação entre a técnica ML com a relação de dimensão Dimension_Relation_18:

Restrição necessária (em linguagem natural):
*Se uma técnica ML (instância de Technique) permite encapsulamento alto para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_18 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:
*ML*
*and has_relation value Dimension_Relation_18.*

gg) Para que uma técnica baseada em Aprendizagem de Máquina/Algoritmo de Particionamento Recursivo (ML) gera resultado com explicabilidade alta para uma amostra de dados, foi definido que isto só é possível, se esta amostra possuir dados com complexidade baixa.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma gg):

Restrição necessária (em linguagem natural):
*Uma relação Dimension_Relation_19 (instância de Dimension_Relation) possui a dimensão, explicabilidade alta (instância de Dimension), e o fator ambiental, dados com complexidade baixa (instância de Environment).*

Axioma que define a restrição necessária:
*Dimension_Relation_19*
*and explainability_value value Explainability_High*
*and complexity_of_data_value value Complexity_of_Data_Low.*

2º. Estabelecer a relação entre a técnica ML com a relação de dimensão Dimension_Relation_19:

Restrição necessária (em linguagem natural):
*Se uma técnica ML (instância de Technique) gera resultado com explicabilidade alta para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_19 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:
*ML*
*and has_relation value Dimension_Relation_19.*

hh) Para que uma técnica baseada em Aprendizagem de Máquina/Algoritmo de Particionamento Recursivo (ML) possua flexibilidade alta para uma amostra de dados, foi definido que isto só é possível, se esta amostra possuir quantidade grande de dados.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma hh):

Restrição necessária (em linguagem natural):
*Uma relação Dimension_Relation_20 (instância de Dimension_Relation) possui a dimensão, flexibilidade alta (instância de Dimension), e o fator ambiental, quantidade grande de dados (instância de Environment).*

Axioma que define a restrição necessária:
*Dimension_Relation_20*
*and flexibility_value value Flexibility_High*
*and amount_of_data_value value Amount_of_Data_Large.*

2º. Estabelecer a relação entre a técnica ML com a relação de dimensão Dimension_Relation_20:

Restrição necessária (em linguagem natural):
*Se uma técnica ML (instância de Technique) possui flexibilidade alta para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_20 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:
*ML*

*and has_relation value Dimension_Relation_20.*

ii) Uma técnica baseada em Aprendizagem de Máquina/Algoritmo de Particionamento Recursivo (ML) gera resultados com tempo baixo de resposta para uma amostra de dados.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma ii):

Restrição necessária (em linguagem natural):
*Uma relação Dimension_Relation_08 (instância de Dimension_Relation) possui a dimensão, tempo baixo de resposta (instância de Dimension).*

Axioma que define a restrição necessária:
*Dimension_Relation_08*
*and response_time_value value Response_Time_Low.*

2º. Estabelecer a relação entre a técnica ML com a relação de dimensão Dimension_Relation_08:

Restrição necessária (em linguagem natural):
*Se uma técnica ML (instância de Technique) gera resultados com tempo baixo de resposta para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_08 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:
*ML*
*and has_relation value Dimension_Relation_08.*

jj) Para que uma técnica baseada em Aprendizagem de Máquina/Algoritmo de Particionamento Recursivo (ML) possua escalabilidade alta para uma amostra de dados, foi definido que isto só é possível, se esta amostra possuir dados com complexidade baixa e quantidade pequena de dados.

Visando atender a este axioma, temos:

1º. Identificar qual relação possui os atributos que satisfazem o axioma jj):

Restrição necessária (em linguagem natural):

*Uma relação Dimension_Relation_21 (instância de Dimension_Relation) possui a dimensão, escalabilidade alta (instância de Dimension), e os fatores ambientais, dados com complexidade baixa (instância de Environment) e quantidade pequena de dados (instância de Environment).*

Axioma que define a restrição necessária:
*Dimension_Relation_21*
*and scalability_value value Scalability_High*
*and complexity_of_data value Complexity_of_Data_Low*
*and amount_of_data_value value Amount_of_Data_Small.*

2º. Estabelecer a relação entre a técnica ML com a relação de dimensão Dimension_Relation_21:

Restrição necessária (em linguagem natural):
*Se uma técnica ML (instância de Technique) possui escalabilidade alta para uma amostra de dados, então, é necessário que ela tenha relação com Dimension_Relation_21 (instância de Dimension_Relation).*

Axioma que define a restrição necessária:
*ML*
*and has_relation value Dimension_Relation_21*.

A compilação de todos os conhecimentos capturados é feita através da linguagem OWL, a partir da interface disponibilizada pela ferramenta *Protégé* versão 4.0 beta.

```
<!--
http://www.semanticweb/ontologies/2009/6/Automatic_Technique_Detection.owl#Dimension_
Relation_01 -->


   <owl:Thing rdf:about="#Dimension_Relation_01">
      <rdf:type rdf:resource="#Dimension_Relation"/>
      <rdfs:label xml:lang="pt"
         >RelacaoDeDimensao01</rdfs:label>
      <rdfs:comment xml:lang="en"
         >Relation about dimension to the high accuracy combined with large amount of
```

Figura 10 - Trecho de código OWL, contendo parte da definição do indivíduo *Dimension_Relation_01*
Nota: elaboração própria

Como exemplo, tem-se na figura 10, parte do código gerado em OWL para a definição do indivíduo *Dimension_Relation_01* pertencente à classe *Technique_Dimension_Relation.*

### 4.1.2 Avaliar

Para avaliação da ontologia, foram aplicados dois procedimentos.

O primeiro, através da opção *Classify* no *Protégé* versão 4.0 beta, aplicando-se o algoritmo *Fact plus plus* e a aplicação *Pellet* (ver item 4). Após a utilização das aplicações, constatou-se a inexistência de classes ambíguas ou sem inferência em relação à estrutura hierárquica de classes da ontologia.

Em relação à aplicação do algoritmo *Fact plus plus*, por exemplo, nota-se que a superclasse *Nothing* permeceu vazia, indicando a inexistência de classes ambíguas ou sem inferência em relação à estrutura hierárquica de classes da ontologia, como pode ser visualizado na figura 11.

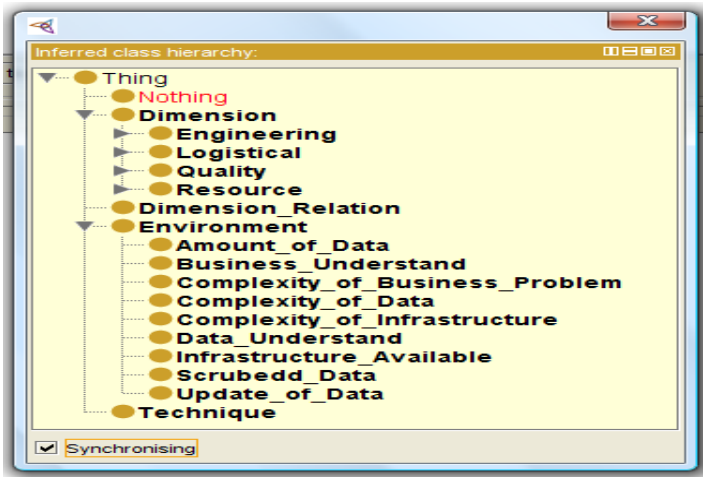


Figura 11 - Visão de inferência sobre hierarquia de classes, após aplicação do algoritmo *Fact plus plus*.
Nota: elaboração própria

O segundo procedimento foi executado através de *queries* a partir da aplicação *DL Query*, disponível nas opções iniciais da *Protégé* versão 4.0 beta, cujo objetivo é verificar se os axiomas inferidos (testar definições de classes quanto a sua hierarquia (relação entre classes, subclasses, propriedades e indivíduos) e quanto à existência de ambigüidades e de não atribuições) atendem às questões a serem respondidas por esta ontologia.

Assim, ao aplicar a *DL Query*, pudemos verificar que as seguintes questões foram respondidas:

a) "*Quais as técnicas para detecção automática existentes para detecção de problemas de fraude*?" (Figura 12).

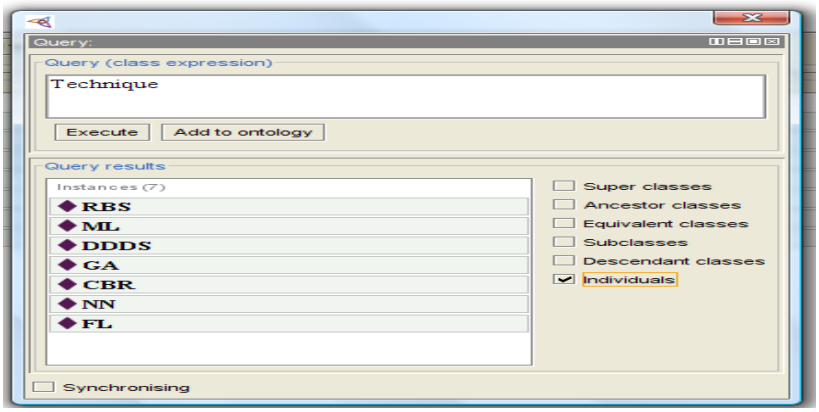


Figura 12 - Relação das técnicas para detecção automática de problemas de fraude
Nota: elaboração própria

b) "*Quais as dimensões de avaliação e fatores ambientais a serem observados, para análise das técnicas para detecção automática a serem usadas*?" (Figuras 13a e 13b).

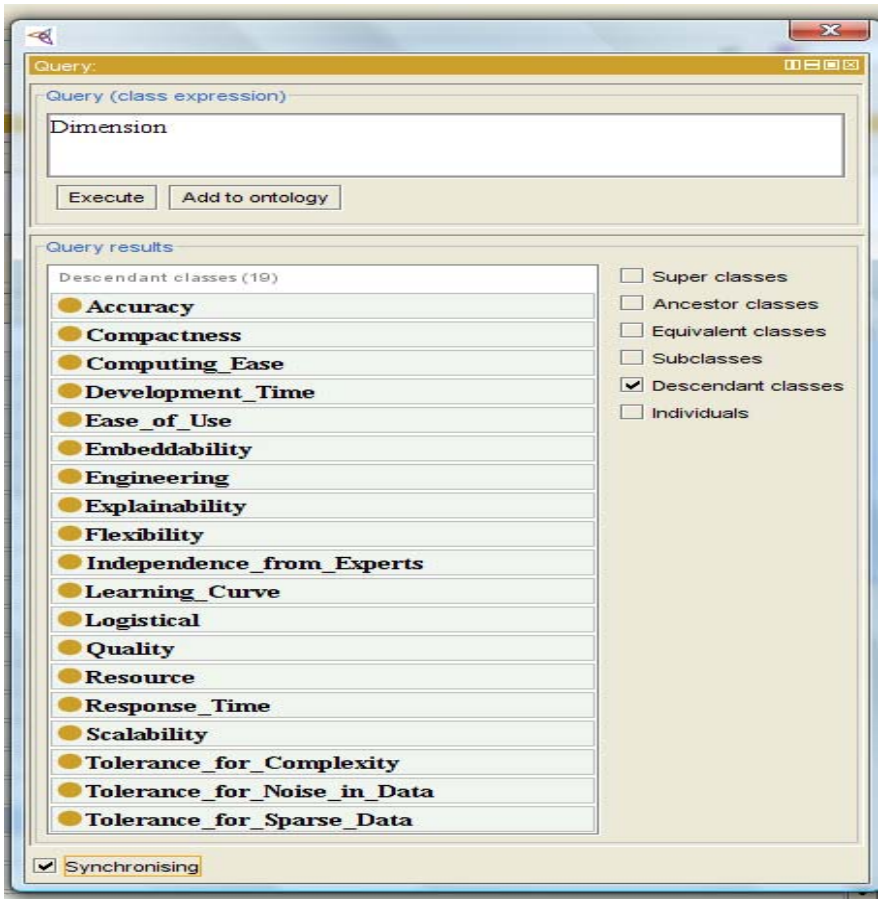


Figura 13a - Relação de classes referentes às dimensões de avaliação.
Nota: elaboração própria

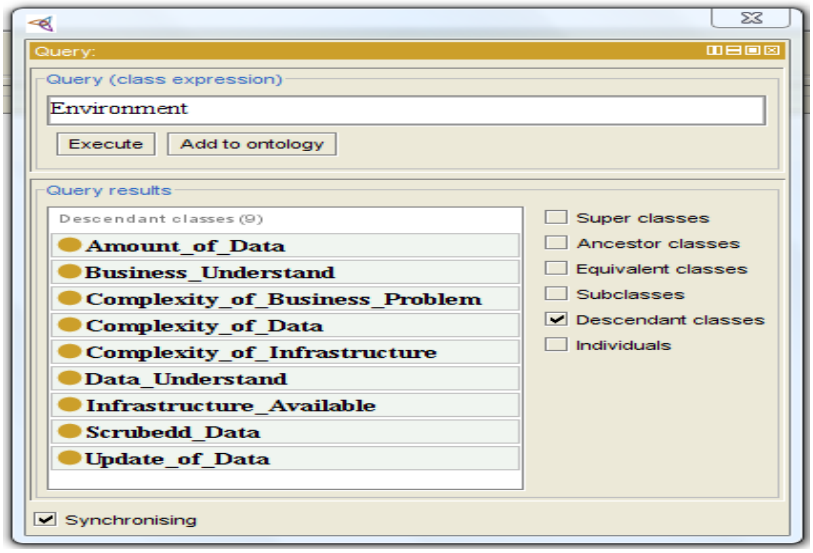


Figura 13b - Relação de classes referentes aos fatores ambientais
Nota: elaboração própria

c) “*Quais as relações entre dimensões de avaliação e fatores ambientais, e uma técnica para detecção automática específica*?” (Figuras 14a e 14b).

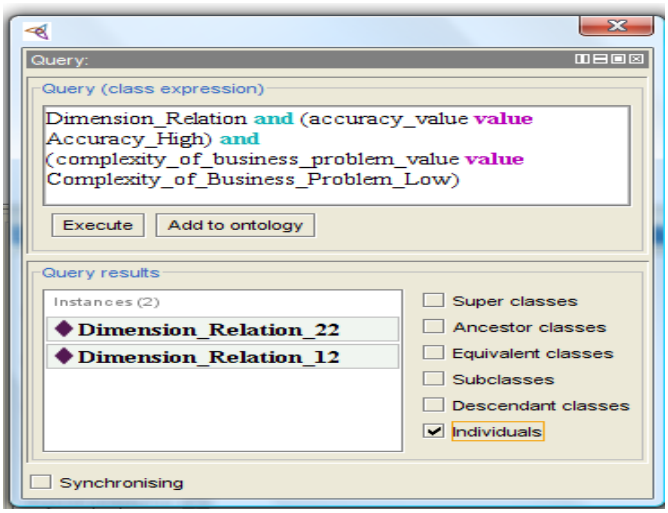


Figura 14a - Relação entre dimensões de avaliação e fatores ambientais
Notas: Como exemplo, tem-se, inicialmente, as relações que possuem como dimensão de avaliação, precisão alta, e como fator ambiental, complexidade baixa do problema representado pela amostra de dados: *Dimension_Relation_22* e *Dimension_Relation_12*.
Elaboração própria

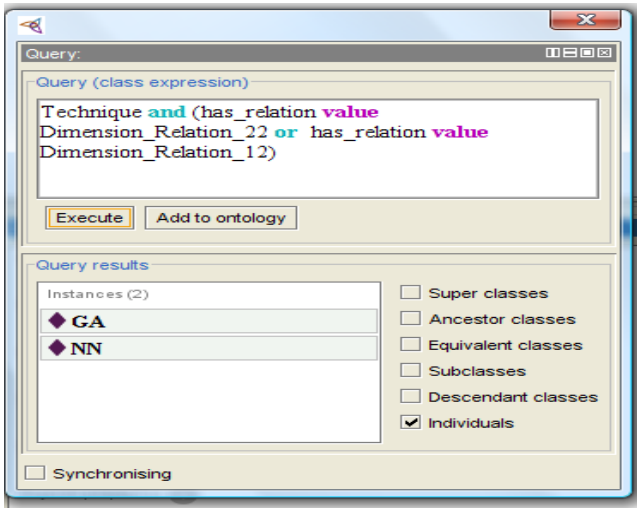


Figura 14b – Relação das técnicas que possuem relação com *Dimension_Relation_22* ou com *Dimension_Relation_12*
Notas: após identificadas as relações de dimensões, obtem-se as técnicas que possuem relação com as mesmas, no exemplo, são as técnicas que geram resultados com precisão alta e que, para isto, necessitam que a complexidade do problema de negócio representado seja baixa: *GA* e *NN*.
Elaboração própria

Como conclusão da avaliação, temos que em ambos os procedimentos, obteve-se resultados esperados, indicando que a ontologia não possui classes ambíguas ou sem referências e é capaz de responder às questões formuladas no item 4 desta dissertação.

### 4.1.3 Documentar

Toda a documentação da ontologia foi gerada no ambiente *Protégé* versão 4.0 beta, com comentários em português e inglês. A figura 15 traz a documentação inicial para esta ontologia, constando de informações sobre a versão da ontologia (número, descrição, data) e de um breve comentário indicando o domínio da ontologia.

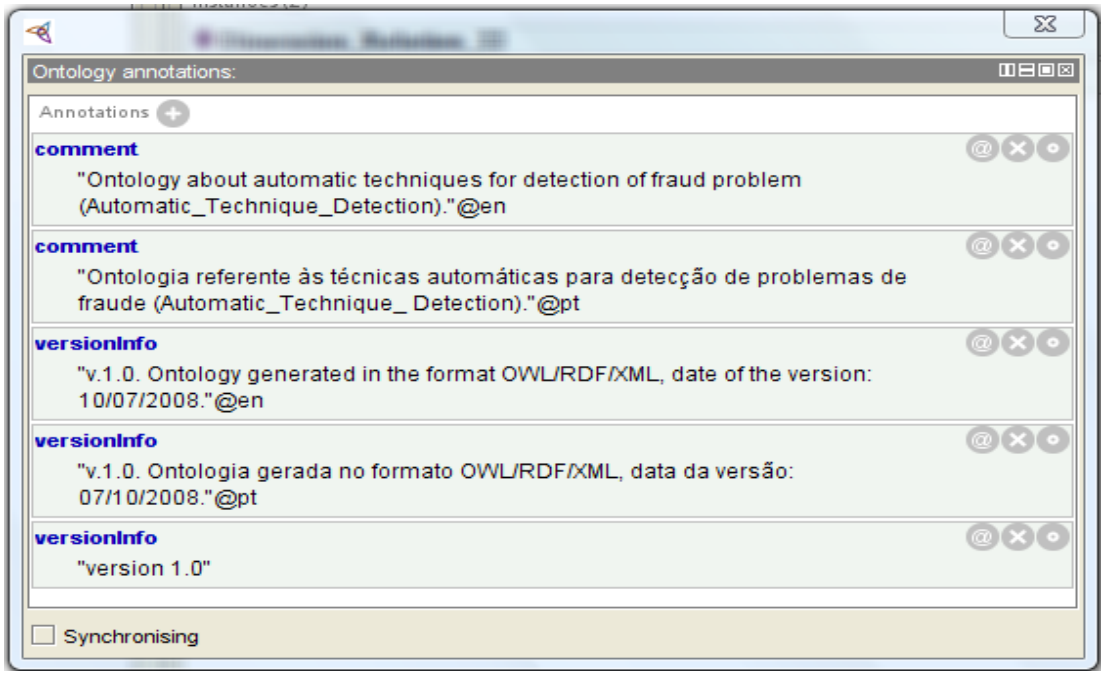


Figura 15 - Documentação inicial para a ontologia Técnicas para Detecção Automática de Fraude (Automatic_Technique_Detection.owl).
Nota: elaboração própria

## 4.2 PROBLEMAS DE FRAUDE

A finalidade desta ontologia é compor, juntamente com a ontologia de técnicas para detecção automática de problemas de fraude, uma base de conhecimentos, cujos objetivos são: permitir a construção de inferências sobre problemas de fraude e identificar quais as técnicas para detecção automática de fraudes são as mais adequadas a um domínio de fraude específico.

Nesta ontologia, a taxonomia para os problemas de fraude foi baseada na proposta apresentada por Allen (1999), tendo como raiz (*root*) o elemento vítima ou *target*. Desta

forma, esta taxonomia tem a perspectiva de quem sofre a fraude (elemento vítima ou *target*) (ver item 2.3).

Na dissertação, a ontologia foi estendida para incluir a taxonomia para dimensões de avaliação, baseada no conceito *Intelligence Density*, elaborado por Dhar e Stein (1997), e usada para caracterizar técnicas para detecção automática de fraudes (ver item 2.4.1). Na taxonomia utilizada para esta ontologia, os conceitos definidos para dimensões de avaliação são utilizados para definir as características que deverão ter os resultados gerados por uma técnica para detecção automática de fraudes junto a um domínio específico.

As questões que se seguem, deverão ser respondidas por esta ontologia:

α) *Quais são os problemas de fraude identificados, considerando a perspectiva da vítima ou do elemento alvo (target)*?

β) *Quais são as relações entre problemas de fraude e dimensões de avaliação, usadas para caracterizar técnicas para detecção automática de fraude*?

Gestores e especialistas em fraudes são potenciais usuários desta ontologia, objetivando entender, por exemplo, quais os requisitos (dimensões de avaliação) necessários para que uma técnica para detecção automática de fraudes seja considerada adequada a um domínio de fraude específico.

### 4.2.1 Construir ontologia

A construção desta ontologia segue a abordagem detalhada no início deste capítulo.

A lista de termos está contida no Apêndice A, e foi construída com base na Revisão Estruturada contida no anexo I desta dissertação, na literatura especializada constante no anexo III e a partir da experiência acumulada pelo autor da dissertação, como especialista junto às áreas de inteligência (voltada para o combate a fraudes) e de risco operacional no Ministério da Previdência Social. Esta lista contém termos referentes a conceitos, relações e instâncias ou indivíduos:

1) Os conceitos (classes) e relações (propriedades) contêm as representações relacionadas ao domínio:

a) Classe chamada de “Dimensão” (*Dimension*), contendo os termos referentes às dimensões para avaliação de técnicas em relação aos seus contextos de uso. Tem como subclasses:

a.1)*Engineering: Compactness, Ease_of_Use, Embeddability, Flexibility, Scalability*;

a.2)*Logistical: Computing_Ease, Development_Time, Independence_from_Experts;*

a.3)*Quality: Accuracy, Explainability, Response_Time;*

a.4)*Resource: Learning_Curve, Tolerance_for_Complexity, Tolerance_for_Noise_in_Data, Tolerance_for_Sparse_Data;*

*b)* Classe chamada de “Problemas de Fraude” (*Fraud*), contendo termos referentes a problemas de fraude, tendo como principal fonte a taxonomia proposta por Allen (1999); ver item 2.3. Tem como subclasses: *Individuals, Individuals_and_Institutuions, Institutuions.*

c) Propriedades:

c.1) Representando as relações mais importantes entre a classe *Fraud* e as sub-classes pertencentes à classe *Dimension*: *need_compactness_value, need_ease_of_use_value, need_embeddability_value, need_flexibility_value, need_scalability_value, need_computing_ease_value, need_development_time_value, need_independence_from_experts_value, need_accuracy_value, need_explainability_value, need_response_time_value, need_learning_curve_value, need_tolerance_for_complexity_value, need_tolerance_for_noise_in_data_value, need_tolerance_for_sparse_data_value.*

2) Indivíduos, referentes:

a) Às instâncias possíveis para cada uma das subclasses da classe *Dimension*:

a.1)*Accuracy: Accuracy_High, Accuracy_Moderate e Accuracy_Low*;

a.2)*Compactness: Compactness_High, Compactness_Moderate, Compactness_Low;*

a.3)*Computing_Ease: Computing_Ease_High, Computing_Ease_Moderate, Computing_Ease_Low;*

a.4)*Development_Time: Development_Time_High, Development_Time_Moderate, Development_Time_Low;*

a.5)*Ease_of_Use: Ease_of_Use_High, Ease_of_Use_Moderate, Ease_of_Use_High;*

a.6)*Embeddability: Embeddability_High, Embeddability_Moderate, Embeddability_Low;*

a.7)*Explainability: Explainability_High, Explainability_Moderate, Explainability_Low;*

a.8)*Flexibility: Flexibility_High, Flexibility_Moderate, Flexibility_Low;*

a.9)*Independence_from_Experts: Independence_from_Experts_High, Independence_from_Experts_Moderate, Independence_from_Experts_Low;*

a.10)*Learning_Curve: Learning_Curve_High, Learning_Curve_Moderate, Learning_Curve_Low;*

a.11)*Response_Time: Response_Time_High, Response_Time_Moderate, Response_Time_Low;*

a.12)*Scalability: Scalability_High, Scalability_Moderate, Scalability_Low;*

a.13)*Tolerance_for_Complexity: Tolerance_for_Complexity_High, Tolerance_for_Complexity_Moderate, Tolerance_for_Complexity_Low;*

a.14)*Tolerance_for_Noise_in_Data: Tolerance_for_Noise_in_Data_High, Tolerance_for_Noise_in_Data_Moderate, Tolerance_for_Noise_in_Data_Low;*

a.15)*Tolerance_for_Sparse_Data: Tolerance_for_Sparse_Data_High, Tolerance_for_Sparse_Data_Moderate, Tolerance_for_Sparse_Data_Low;*

b) Às instâncias possíveis para a classe *Fraud*, considerando a taxonomia para os problemas de fraude, baseada na proposta apresentada por Allen (1999), que tem como raiz (*root*) o elemento vítima ou *target*, sendo, portanto, estruturada

a partir da perspectiva de quem sofre a fraude (elemento vítima ou *target*) (ver item 2.3). São elas:

b.1)*Individuals: Confidence_Games, Consumer_Fraud;*

b.2)*Individuals_and_Institutuions: Advanced_Fee_Fraud, Financial_Fraud, Investment_Fraud, Science_Fraud;*

b.3)*Institutuions: Government_Fraud, Operations_Fraud, Public_Fraud, Telecomm_Fraud.*

As taxonomias presentes na ontologia são:

a) Taxonomia-base contendo os principais grupos de classes (Figura 16):

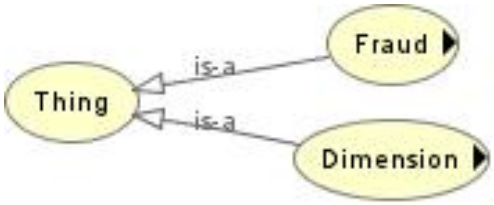


Figura 16 - Taxonomia-base para ontologia “Problemas de Fraudes”
Nota: elaboração própria

b) Taxonomia para a classe *Fraud* (Figura 17):

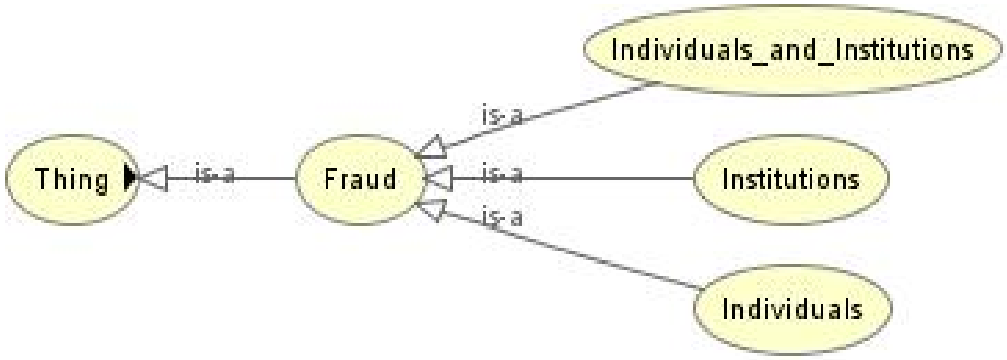


Figura 17 - Taxonomia para a classe *Fraud*
Nota: elaboração própria

c) Taxonomia para a classe *Dimension* e suas sub-classes (Figura 18):


Figura 18 - Taxonomia para a classe *Dimension* e suas sub-classes
Nota: elaboração própria.

Os principais axiomas contidos nesta ontologia são:

a) Um problema de fraude do tipo Fraude em Adiantamento de Taxa *(Advance_Fee_Fraud)* necessita que uma técnica de detecção automática de fraude permita independência alta de especialistas, apresente facilidade alta de uso, e possua curva de aprendizagem alta.

Visando atender a este axioma, temos:

Restrição necessária (em linguagem natural):

*Se um problema de fraude é do tipo Advance_Fee_Fraud (Fraud), então ele necessita que uma técnica de detecção automática de fraude, permita independência alta de especialistas (instância de Dimension), apresente facilidade alta de uso (instância de Dimension), e possua curva de aprendizagem alta (instância de Dimension).*

Axioma que define a restrição necessária:
*Advance_Fee_Fraud*
*and need_independence_from_experts_value value Independence_from_Experts_High*
*and need_ease_of_use_value value Ease_of_Use_High*
*and need_learning_curve_value value Learning_Curve_High.*

b) Um problema de fraude do tipo Quebra de Confiança *(Confidence_Games)* necessita que uma técnica de detecção automática de fraude, permita independência alta de especialistas, apresente facilidade alta de uso, e possua curva de aprendizagem alta.

Visando atender a este axioma, temos:

Restrição necessária (em linguagem natural):
*Se um problema de fraude é do tipo Confidence_Games (instância de Fraud), então ele necessita que uma técnica de detecção automática de fraude, permita independência alta de especialistas (instância de Dimension), apresente facilidade alta de uso (instância de Dimension), e possua curva de aprendizagem alta (instância de Dimension).*

Axioma que define a restrição necessária:
*Confidence_Games*
*and need_independence_from_experts_value value Independence_from_Experts_High*
*and need_ease_of_use_value value Ease_of_Use_High*
*and need_learning_curve_value value Learning_Curve_High.*

c) Um problema de fraude do tipo Fraude ao Consumidor *(Consumer_Fraud)* necessita que uma técnica de detecção automática de

fraude, possua tolerância alta a dados incompletos, e possua tolerância alta a dados inconsistentes.

Visando atender a este axioma, temos:

Restrição necessária (em linguagem natural):
*Se um problema de fraude é do tipo Consumer_Fraud (instância de Fraud), então ele necessita que uma técnica de detecção automática de fraude, possua tolerância alta a dados incompletos (instância de Dimension), e possua tolerância alta a dados inconsistentes (instância de Dimension).*

Axioma que define a restrição necessária:
*Consumer_Fraud*
*and need_tolerance_for_sparse_data_value value Tolerance_for_Sparse_Data_High*
*and need_tolerance_for_noise_in_data_value value Tolerance_for_Noise_in_Data_High.*

d) Um problema de fraude do tipo Fraude Financeira *(Financial_Fraud)* necessita que uma técnica de detecção automática de fraude, possua tolerância alta à complexidade, gere resultado com tempo baixo de resposta, e gere resultado com precisão alta.

Visando atender a este axioma, temos:

Restrição necessária (em linguagem natural):
*Se um problema de fraude é do tipo Financial_Fraud (instância de Fraud), então ele necessita que uma técnica de detecção automática de fraude, possua tolerância alta à complexidade (instância de Dimension), gere resultado com tempo baixo de resposta (instância de Dimension), e gere resultado com precisão alta (instância de Dimension).*

Axioma que define a restrição necessária:
*Financial_Fraud*
*and need_tolerance_for_complexity_value value Tolerance_for_Complexity_High*
*and need_response_time_value value Response_Time_Low*
*and need_accuracy_value value Accuracy_High.*

e) Um problema de fraude do tipo Fraude Governamental *(Government_Fraud)* necessita que uma técnica de detecção automática de fraude, possua tolerância alta à complexidade, possua tolerância alta a dados incompletos, gere resultado com explicabilidade alta, e possua tolerância alta a dados inconsistentes.

Visando atender a este axioma, temos:

Restrição necessária (em linguagem natural):
*Se um problema de fraude é do tipo Government_Fraud (instância de Fraud), então ele necessita que uma técnica de detecção automática de fraude, possua tolerância alta à complexidade (instância de Dimension), possua tolerância alta a dados incompletos (instância de Dimension), gere resultado com explicabilidade alta (instância de Dimension), e possua tolerância alta a dados inconsistentes (instância de Dimension).*

Axioma que define a restrição necessária:
*Government_Fraud*
*and need_tolerance_for_complexity_value value Tolerance_for_Complexity_High*
*and need_tolerance_for_sparse_data_value value Tolerance_for_Sparse_Data_High*
*and need_explainability_value value Explainability_High*
*and need_tolerance_for_noise_in_data_value value Tolerance_for_Noise_in_Data_High.*

f) Um problema de fraude do tipo Fraude em Investimento *(Investment_Fraud)* necessita que uma técnica de detecção automática de fraude, possua tolerância alta à complexidade, e gere resultado com precisão alta.

Visando atender a este axioma, temos:

Restrição necessária (em linguagem natural):
*Se um problema de fraude é do tipo Investment_Fraud (instância de Fraud), então ele necessita que uma técnica de detecção automática de fraude, possua*

*tolerância alta à complexidade (instância de Dimension), e gere resultado com precisão alta (instância de Dimension).*

Axioma que define a restrição necessária:
*Investment_Fraud*
*and need_tolerance_for_complexity_value value Tolerance_for_Complexity_High*
*and need_accuracy_value value Accuracy_High.*

g) Um problema de fraude do tipo Fraude em Operações *(Operations_Fraud)* necessita que uma técnica de detecção automática de fraude, permita compacidade alta, possua flexibilidade alta, permita encapsulamento alto e escalabilidade alta.

Visando atender a este axioma, temos:

Restrição necessária (em linguagem natural):
*Se um problema de fraude é do tipo Operations_Fraud (instância de Fraud), então ele necessita que uma técnica de detecção automática de fraude, permita compacidade alta (instância de Dimension), possua flexibilidade alta (instância de Dimension), permita encapsulamento alto (instância de Dimension) e escalabilidade alta (instância de Dimension).*

Axioma que define a restrição necessária:
*Operations_Fraud*
*and need_compactness_value value Compactness_High*
*and need_flexibility_value value Flexibility_High*
*and need_embeddability_value value Embeddability_High*
*and need_scalability_value value Scalability_High.*

h) Um problema de fraude do tipo Fraude do Interesse Público *(Public_Fraud)* necessita que uma técnica de detecção automática de fraude, possua tolerância alta a dados inconsistentes e tolerância alta a dados incompletos.

Visando atender a este axioma, temos:

Restrição necessária (em linguagem natural):

*Se um problema de fraude é do tipo Public_Fraud (instância de Fraud), então ele necessita que uma técnica de detecção automática de fraude, possua tolerância alta a dados inconsistentes (instância de Dimension) e tolerância alta a dados incompletos (instância de Dimension).*

Axioma que define a restrição necessária:
*Public_Fraud*
*and need_tolerance_for_noise_in_data_value value Tolerance_for_Noise_in_Data_High*
*and need_tolerance_for_sparse_data_value value Tolerance_for_Sparse_Data_High.*

i) Um problema de fraude do tipo Fraude Científica *(Science_Fraud)* necessita que uma técnica de detecção automática de fraude, possua flexibilidade alta.

Visando atender a este axioma, temos:

Restrição necessária (em linguagem natural):
*Se um problema de fraude é do tipo Science_Fraud (instância de Fraud), então ele necessita que uma técnica de detecção automática de fraude, possua flexibilidade alta (instância de Dimension).*

Axioma que define a restrição necessária:
*Science_Fraud*
*and need_flexibility_value value Flexibility_High.*

j) Um problema de fraude do tipo Fraude em Telecomunicações *(Telecomm_Fraud)* necessita que uma técnica de detecção automática de fraude, possua tolerância alta à complexidade, e gere resultado com precisão alta.

Visando atender a este axioma, temos:

Restrição necessária (em linguagem natural):
*Se um problema de fraude é do tipo Telecomm_Fraud (instância de Fraud), então ele necessita que uma técnica de detecção automática de fraude, possua*

*tolerância alta à complexidade (instância de Dimension), e gere resultado com precisão alta (instância de Dimension).*

Axioma que define a restrição necessária:
*Telecomm_Fraud*
*and need_tolerance_for_complexity_value value Tolerance_for_Complexity_High*
*and need_accuracy_value value Accuracy_High.*

A compilação de todos os conhecimentos capturados é feita através da linguagem OWL, a partir da interface disponibilizada pela ferramenta *Protégé* versão 4.0 beta.

```
<!—http://www.semanticweb/ontologies/2009/6/Fraud_Problem.owl#Science_Fraud -->
  <Individuals_and_Institutions rdf:about="#Science_Fraud">
    <rdf:type rdf:resource="&owl;Thing"/>
    <rdfs:label xml:lang="pt">FraudeCientífica</rdfs:label>
    <rdfs:comment xml:lang="en">
      (Science Fraud) is a fraud problem where the perpetrator or target has origin in the
      academic area...
    </rdfs:comment>
    <rdfs:comment xml:lang="pt">
```

Figura 19 - Trecho de código OWL, contendo parte da definição do indivíduo *Science_Fraud* Nota: elaboração própria

Como exemplo, tem-se na figura 19 parte do código gerado em OWL para a definição do indivíduo *Science_Fraud* pertencente à classe *Fraud*.

### 4.2.2 Avaliar

Assim como no item 4.1.2, também foram aplicados dois procedimentos para a avaliação da ontologia. O primeiro através do algoritmo *Fact plus plus* e da aplicação *Pellet 1*.5, e o segundo através de *queries* a partir da aplicação *DL Query*.

A figura 20 ilustra o resultado da inferência sobre a ontologia, a partir da aplicação do algoritmo *Fact plus plus*, onde é possível perceber que nenhuma classe é atribuída à superclasse *Nothing*, implicando na inexistência de classes ambíguas ou sem referência na estrutura hierárquica proposta.

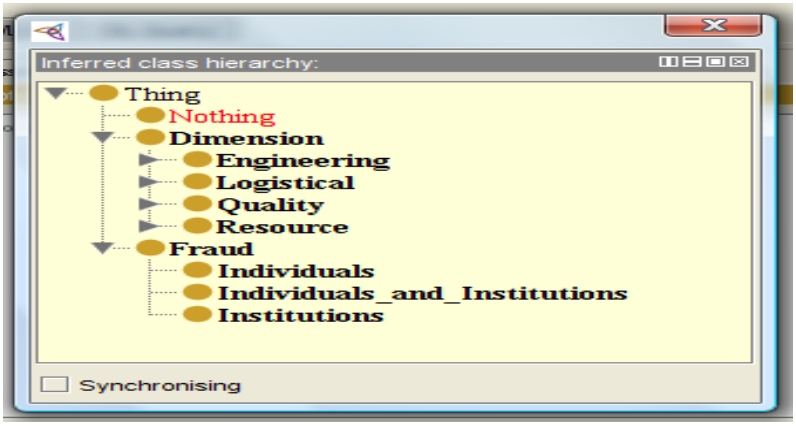


Figura 20 - Visão de inferência para hierarquia de classes, gerada após aplicação do algoritmo *Fact plus plus*
Nota: elaboração própria

Quanto à aplicação da DL Query, podemos verificar que as seguintes questões foram respondidas:

a) "*Quais são os problemas de fraude identificados, considerando a perspectiva da vítima ou do elemento alvo (target)*?" (Figura 21).

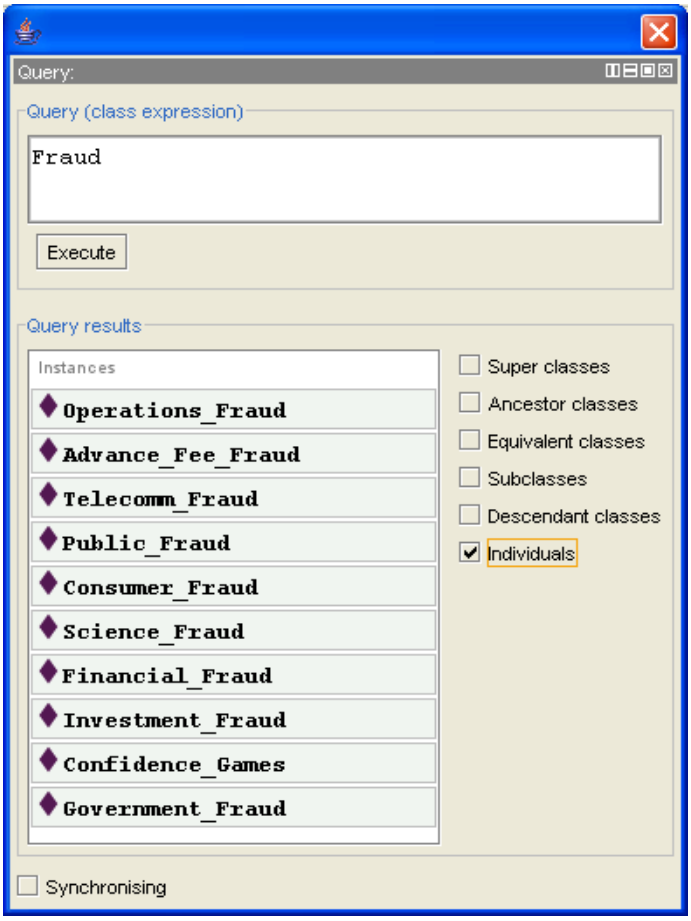


Figura 21 - Relação de indivíduos pertencentes à classe *Fraud*
Nota: elaboração própria

b) "*Quais são as relações entre problemas de fraude e dimensões de avaliação, a partir das quais é possível determinar as técnicas para detecção automática de fraude mais adequadas a um domínio específico?*" (Figura 22).

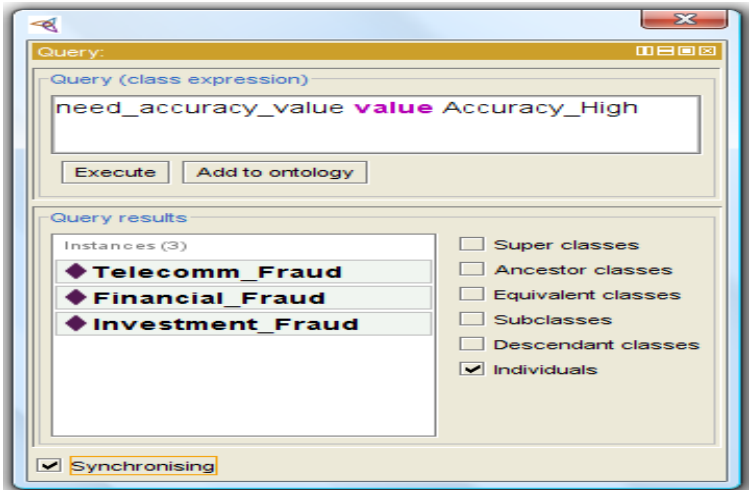


Figura 22 - Exemplo de relação entre problemas de fraude (instância de *Fraud*) e dimensões de avaliação (instância de *Dimension*)

Nota: neste exemplo, são indicados os problemas de fraude, *Telecomm_Fraud, Financial_Fraud e InvestmentFraud*, que necessitam técnicas para detecção automática de fraudes que gerem respostas com precisão alta.

Elaboração própria

### 4.2.3 Documentar

Toda a documentação da ontologia foi gerada no ambiente *Protégé* versão 4.0 beta, com comentários em português e inglês. A figura 23 traz a documentação inicial para esta ontologia.

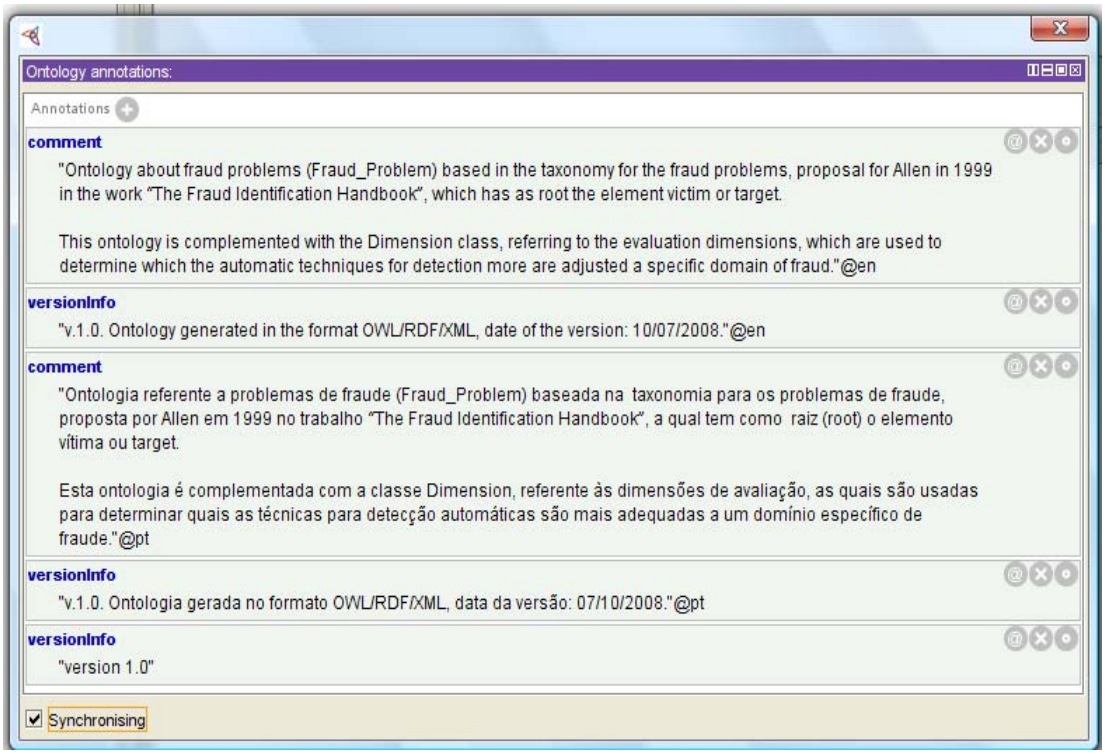


Figura 23 - Documentação inicial para a ontologia Problemas de Fraude (F*raud_Problem.owl*)

Nota: elaboração própria

## 4.3 DETECÇÃO AUTOMÁTICA DE FRAUDES (MERGE)

A finalidade desta ontologia é complementar a base de conhecimentos, composta pelas ontologias para técnicas para detecção automática de fraude e problemas de fraude, a partir do *merging* entre estas. Deste modo, é possível responder à questão colocada no início desta dissertação:

a) *Qual ou quais as técnicas para detecção automática são mais adequadas a um domínio de problema de fraude específico*?

Gestores e especialistas em fraudes, além de especialistas e fornecedores de tecnologias para detecção automática de fraudes, são potenciais usuários desta base de conhecimentos, quando desejarem aplicar uma técnica para detecção automática de fraudes.

### 4.3.1 Construir ontologia

A construção desta ontologia foi feita através do *merging* entre as ontologias, "Técnicas para Detecção Automática de Fraude" ou "*Automatic_Technique_Detection.owl"*, e "Problemas de Fraude" ou "*Fraud_Problem.owl"*. Este procedimento foi executado por meio da opção "*Merge ontologies*" disponível no *Protégé* versão 4.0 beta.

Após execução do *merging*, obteve-se uma nova ontologia, "Detecção Automática de Fraudes" ou "*Fraud_Detection.owl"*, conforme observado nas figuras 24 e 25:

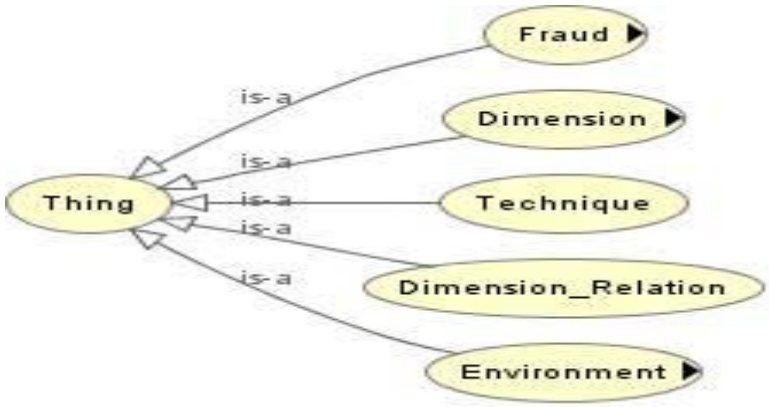


Figura 24 - Taxonomia-base para ontologia "Detecção Automática de Fraudes" (*Fraud_Detection.owl)*
Nota: elaboração própria

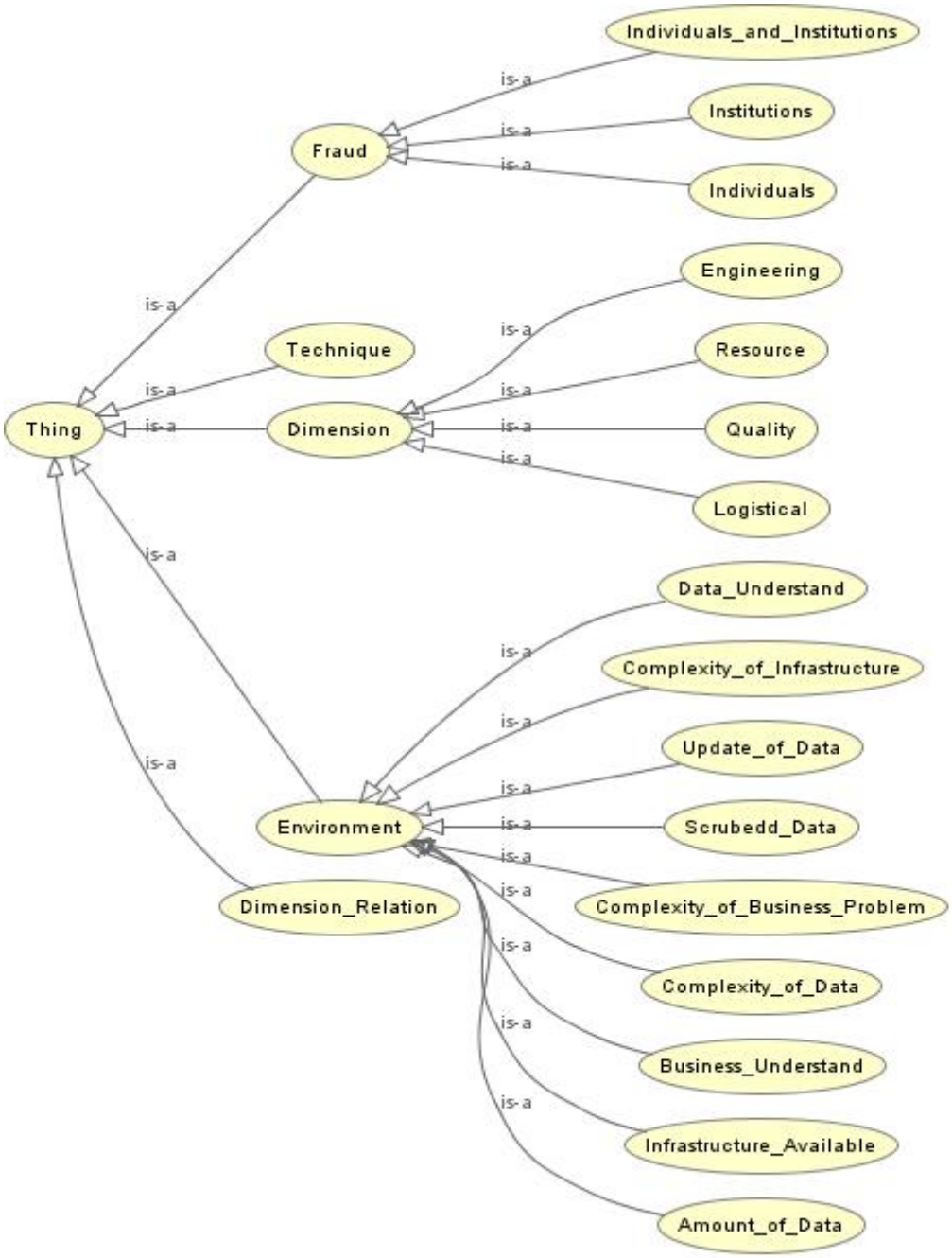


Figura 25 - Detalhamento da taxonomia-base para ontologia “Detecção Automática de Fraudes” (*Fraud_Detection.owl)*.
Nota: elaboração própria

Assim, conforme demonstrado pelas figuras 24 e 25, o *merging* entre as ontologias dispõe as classes de acordo com suas hierarquias originais, eliminando as redundâncias através da supressão das classes coincidentes, como é o caso da classe *Dimension*, presente nas ontologias “Técnicas para Detecção Automática de Fraude” e “Problemas de Fraude”.

### 4.3.2 Avaliar

Para avaliar a ontologia gerada, foram utilizados os mesmos procedimentos já citados anteriormente (itens 4.1.2 e 4.2.2), obtendo, por exemplo, a visão de inferência para hierarquia de classes, gerada após aplicação do algoritmo *Fact plus plus*, conforme demonstrada na figura 26.

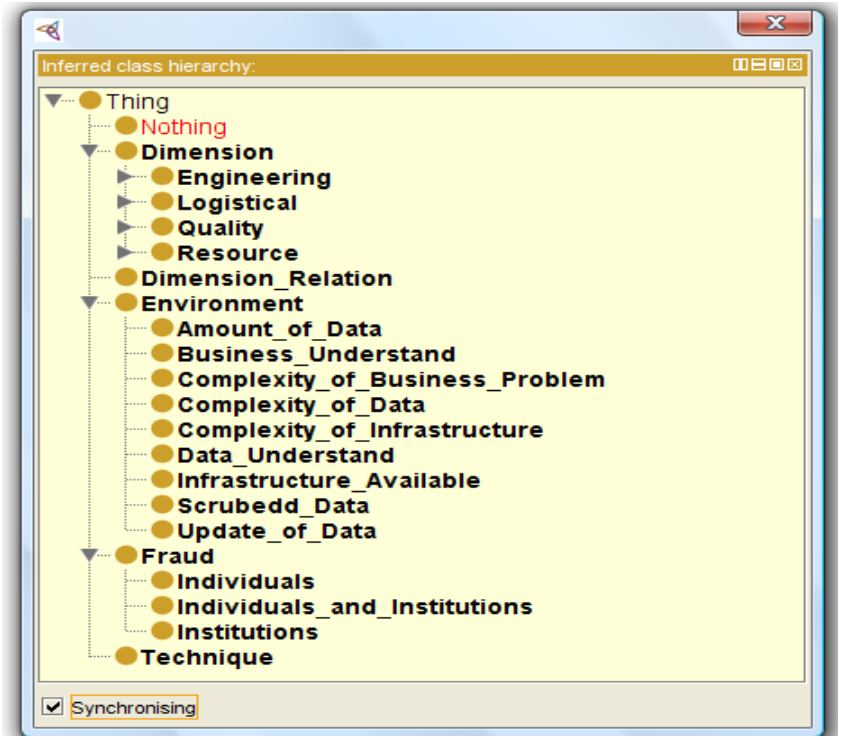


Figura 26 - Visão de inferência para hierarquia de classes, gerada após aplicação do algoritmo *Fact plus plus*.
Nota: elaboração própria

Em relação à utilização da aplicação *DL Query*, procuramos responder a questão fundamental no contexto desta ontologia:

a) "*Qual ou quais as técnicas para detecção automática são mais adequadas a um domínio de problema de fraude específico*?" (Figuras 27, 28 e 29):

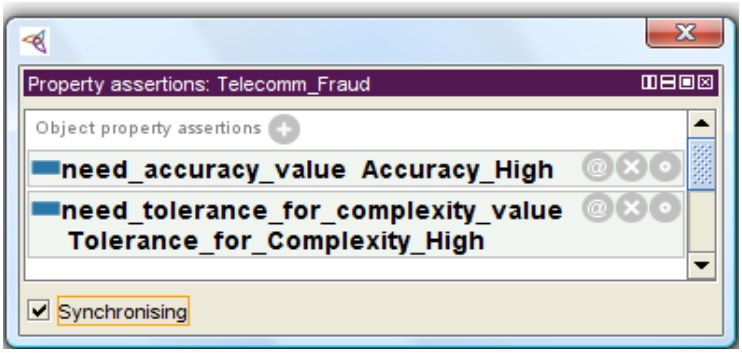


Figura 27 – Relação de dimensões de avaliação (instâncias de *Dimension*)

Notas: dado o tipo de problema de fraude, Fraude em Telecomunicações (*Telecomm_Fraud*), identificamos quais as dimensões de avaliação (instâncias de *Dimension*) necessitarão ser atendidas, preferencialmente, por uma ou mais técnicas para detecção automática de fraudes.

Elaboração própria

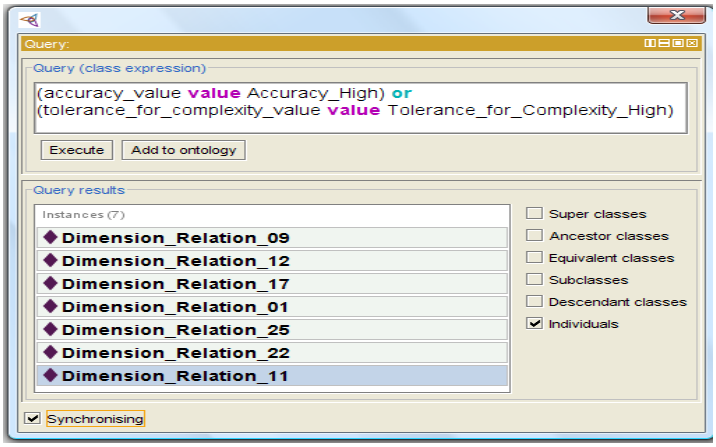


Figura 28 - Relações de dimensão (instâncias de *Dimension_Relation*), que possuem pelo menos uma das dimensões de avaliação principais, as quais satisfazem às necessidades demandadas pelo problema Fraude em Telecomunicações (*Telecomm_Fraud*).

Nota: elaboração própria

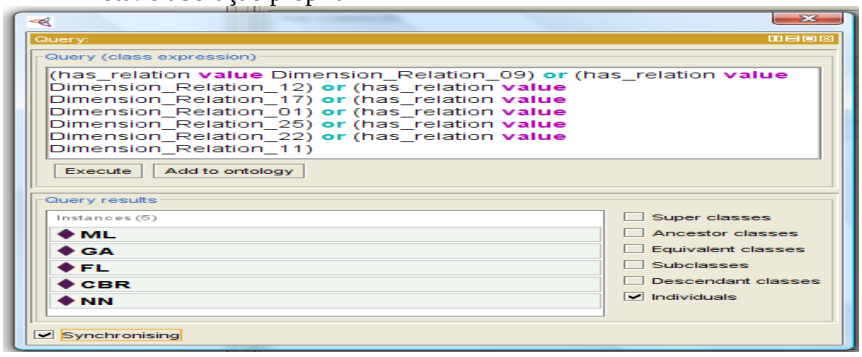


Figura 29 – Relação de técnicas mais adequadas para detecção automáticas de fraudes em telecomunicações.

Notas: identificamos quais as técnicas para detecção automática de fraude (instâncias de *Technique*) que se relacionam com as relações de dimensão (instâncias de *Dimension_Relation*), as quais possuem as principais dimensões de avaliação que satisfazem às necessidades demandadas pelo problema de Fraude em Telecomunicações (*Telecomm_Fraud*).

Elaboração própria

Assim, dado um tipo de problema de fraude, por exemplo, Fraude em Telecomunicações (*Telecomm_Fraud*), identificamos quais são as principais dimensões de avaliação (instâncias de *Dimension*) que necessitarão ser atendidas, preferencialmente, por uma ou mais técnicas para detecção automática de fraudes neste domínio. No caso de Fraude em Telecomunicações, as principais dimensões de avaliação são: precisão alta (*Accuracy_High*) e tolerância alta a complexidade (*Tolerance_for_Complexity_High*), conforme demonstrado pela figura 27.

Em seguida, levantamos quais as relações de dimensão (instâncias de *Dimension_Relation*) possuem as principais dimensões de avaliação que satisfazem às necessidades demandadas pelo problema de fraude, Fraude em Telecomunicações (Figura 28).

Por fim, identificamos quais as técnicas para detecção automática de fraude (instâncias de *Technique*) que se relacionam com as relações de dimensão (instâncias de *Dimension_Relation*) levantadas, obtendo assim a indicação daquelas mais adequadas para serem aplicadas junto a problemas de fraudes em telecomunicações, considerando o contexto existente. De acordo com a figura 29, as técnicas identificadas são: ML (Machine Learning/Recursive Partitioning Algorithm), GA (Genetic Algorithm), FL (Fuzzy Logic), CBR (Case Based Reasoning ) e NN (Neural Networks).

### 4.3.3 Documentar

Como nas ontologias anteriores, toda a documentação da ontologia foi gerada no ambiente *Protégé* versão 4.0 beta, com comentários em português e inglês. A figura 30 traz a documentação inicial para esta ontologia.

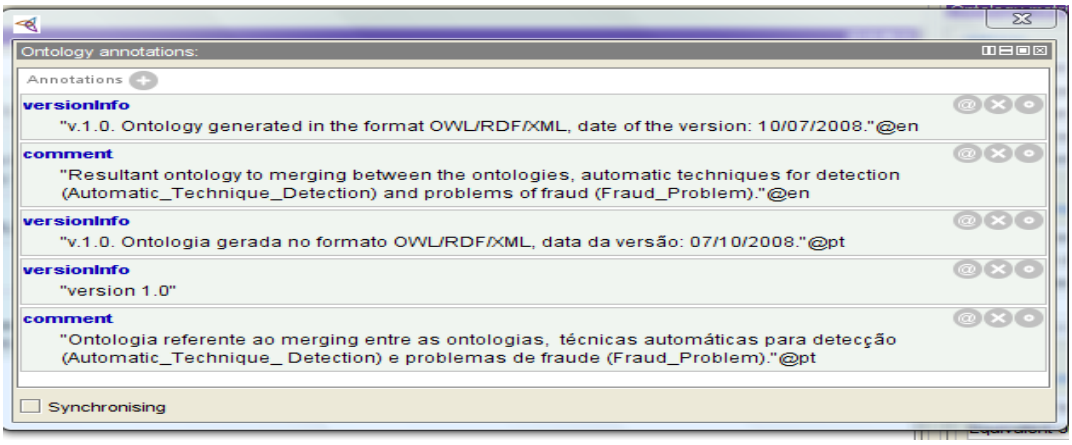


Figura 30 - Documentação inicial para a ontologia Detecção de Fraudes (*Fraud_Detection.owl*)

Nota: elaboração própria

## 4.4 CENÁRIOS DE APLICAÇÃO DAS ONTOLOGIAS

Os cenários para a aplicação das ontologias são múltiplos e estão concentrados na definição das técnicas para detecção automática de fraudes mais adequadas a um determinado domínio.

Conforme demonstrado no item anterior 4.3.2, é possível, com base na ontologia para Detecção de Fraude (*Fraud_Detection*), identificar qual a técnica de detecção automática mais adequada a fim de atender requisitos específicos de um problema de fraude.

É possível identificar os requisitos estabelecidos para os problemas de fraude, definidos na ontologia para Problemas de Fraude (*Fraud_Problem*), conforme demontrado no item 4.2.2.

Outra aplicação imediata para a base de conhecimentos formada pelas ontologias Técnicas para Detecção Automática de Fraude (*Automatic_Technique_Detection.owl*), Problemas de Fraude (*Fraud_Problem.owl*) e o *merging* entre elas (Detecção Automática de Fraude ou *Fraud_Detection.owl*), refere-se à possibilidade de se identificar dimensões de avaliação para as técnicas para detecção automática de fraudes e os fatores ambientais que poderão gerar impactos sobre estas dimensões.

Por fim, as ontologias disponibilizadas podem ser referências para a construção de conhecimentos mais detalhados em relação aos domínios tratados pelas mesmas.

## 4.5 CONCLUSÃO

As ontologias Técnicas para Detecção Automática de Fraudes e Problemas de Fraude, foram implementadas com base na metodologia Uschold & King´s *method*, tendo sido efetuado um *merging* entre ambas, resultando numa terceira ontologia chamada de “Detecção Automática de Fraudes”.

Foram utilizados recursos, como a linguagem OWL, a ferramenta *Protégé* versão 4.0 beta e seus *plug-ins* associados.

Como resultado, foi gerada uma base de conhecimentos, prevista nos objetivos definidos para esta dissertação, a qual permite associar de forma efetiva, os domínios de problemas de fraudes e de técnicas para detecção automática, a serem utilizadas durante o processo de detecção de fraude.

# CAPÍTULO 5

## 5 CONSIDERAÇÕES FINAIS

Nesta dissertação, foram apresentados os aspectos conceituais para construção de uma base de conhecimentos, capaz de apoiar a escolha de técnicas mais adequadas à detecção automática de fraudes, a partir de um domínio de fraude específico.

Segundo a literatura sobre o assunto, a definição criteriosa e adequada destas técnicas a serem empregadas com esta finalidade, é um dos fatores críticos de sucesso dentro do Ciclo de Vida da Gestão de Fraudes (BOLTON; HAND, 2002).

A dissertação emprega ontologias para resolver tal questão. Nela, tal escolha é detalhadamente justificada, a partir da apresentação dos conceitos fundamentais sobre ontologia, seus propósitos, seus tipos, suas principais linguagens de representação e as ferramentas mais conhecidas para apoio ao processo de construção ontológica. Além disto, as principais metodologias para construção de ontologias são apresentadas, como também, um esquema para seleção daquela que será empregada no processo de implementação.

O domínio de fraudes apresenta particularidades que o torna único. Dentre estas, a mais significativa é aquela que se refere ao Ciclo de Vida de Gestão de Fraudes, o qual se configura numa rede de nós interdependentes e inter-relacionados, e onde através desta rede os diversos domínios de fraudes são entendidos.

Esta dissertação tratou a questão referente a um destes estágios descritos no Ciclo de Vida, detecção de fraude, e para tanto, apresentou os conceitos, as caracterizações, e os elementos constitutivos de uma ocorrência fraudulenta, visando o entendimento do tema. A partir destas definições, foram apresentadas duas propostas de taxonomias para fraudes, uma baseada no elemento vítima e outra no elemento fraudador, tendo sido escolhida a primeira por caracterizar de modo mais claro os problemas de fraudes.

O mesmo foi feito para o domínio, técnicas para detecção automática de fraudes, que também compõe a base de conhecimentos. Conceitos e tipologias foram apresentados em relação à taxonomia empregada.

O processo de *merging* aplicado junto às duas ontologias criadas veio completar a base de conhecimentos implementada.

A aplicação das ontologias definidas neste trabalho, ao apontar uma técnica como mais adequada para um problema de fraude, vem ao encontro de uma necessidade de vários segmentos econômicos importantes, como financeiro e de seguros, cujo volume de ativos empregados no estágio de detecção, em 2007, representou algo entre nove a dez bilhões de dólares (ECONOMIST INTELLIGENCE UNIT AND KROLL, 2007).

Assim, a escolha de uma técnica mal adequada para um problema, seja devido à qualidade dos dados envolvidos seja pela complexidade existente, pode implicar na inviabilidade, por exemplo, de um projeto de combate a ocorrência de fraudes dentro de uma organização.

Nesta perspectiva, a existência de uma base de conhecimentos, desenvolvida de forma pretérita e de modo colaborativo, torna-se uma grande ferramenta de apoio às ações de gestão de fraudes, uma vez que, além de orientar uma escolha mais precisa quanto à compra de ferramentas para serem empregadas na detecção, reduz a possibilidade de aquisições indevidas, e retira das organizações o ônus de desenvolver o conhecimento referente à relação entre técnicas para detecção automática e problemas de fraudes.

Sendo assim, o trabalho efetuado pode se constituir numa base inicial a ser tratada como referência a projetos futuros para detecção automática de fraudes.

## 5.1 CONTRIBUIÇÕES

Como contribuições, este trabalho apresenta:

1) Disponibilização de uma base de conhecimentos com capacidade para apoiar a indicação de uma técnica mais adequada a um problema de fraude, reduzindo deste modo, escolhas inadequadas, tanto em termos de efetividade quanto em termos de custos;

2) Disponibilização da mesma base de conhecimentos para apoiar a implementação de aplicações baseadas nas três ontologias desenvolvidas (Problemas de Fraude, Técnicas para Detecção Automática de Fraudes, e Detecção Automática de Fraudes, resultante de um *merging* entre as duas primeiras);

3) Proposição de um *framework* para a seleção de metodologias, com o objetivo de construir ontologias, considerando os seus propósitos e os contextos nos quais elas se encontram, baseado no esquema de avaliações de metodologias elaborado originalmente por Gómez-Pérez *et al.* (2004).

## 5.2 LIMITAÇÕES

As limitações encontradas na construção deste trabalho se concentraram na disponibilização de fontes de informações sobre os temas envolvidos e na aplicação prática da base de conhecimentos desenvolvida.

Segundo Allen (1999), a maior dificuldade em desenvolver pesquisas sobre o universo de fraudes é que “quem pratica não declara e quem é vítima quase sempre se cala.”. Deste modo, o material de pesquisa é escasso e a probabilidade de se desenvolver uma pesquisa *in loco* é muito difícil.

O conservadorismo também é outro fator limitante das pesquisas nesta área. De acordo com Phua (2003), o histórico sobre o conjunto de técnicas para detecção automática empregado é muito restrito, o que dificulta, por exemplo, uma análise mais qualificada sobre a efetividade das técnicas empregadas junto aos domínios.

Em relação ao uso de ontologias, uma das mais críticas limitações diz respeito ao acesso de ontologias já disponíveis sobre este tema, pelas razões apresentadas acima, o que torna bastante difícil a reusabilidade de conhecimentos. Além disto, existe uma carência na disponibilidade de ferramentas de apoio atualizadas e estáveis.

O *Protégé* versão 4.0 beta usado nesta dissertação, apresenta um desempenho melhor do que suas versões anteriores, mas ainda apresenta instabilidades, como por exemplo, a perda de referência para a URI relacionada a uma ontologia que está sendo construída.

## 5.3 TRABALHOS FUTUROS

No trabalho apresentado alguns pontos não foram tratados e, portanto, não vieram a compor o mesmo.

Em relação às ontologias, ambas tem um propósito de compor uma base de conhecimentos, construída com a finalidade de indicar a viabilidade do uso desta tecnologia na solução da

questão tratada nos objetivos desta dissertação. O aprofundamento na construção das ontologias é fator preponderante para tornar a base de conhecimentos desenvolvida, capaz de oferecer maior precisão às aplicações a serem construídas utilizando a mesma.

Em se tratando de aprofundar os conhecimentos expostos pelas ontologias disponibilizadas, cabe registrar a necessidade do aperfeiçoamento da ontologia “Técnicas para Detecção Automática de Fraude”, com a inclusão de conhecimentos relativos aos métodos para efetuar a comparação entre os dados observados de uma amostra e os respectivos valores esperados, durante a fase de detecção de fraudes.

Estes métodos são importantes para os processos de aplicação dos algoritmos que compõem as técnicas, pois tem uma relação direta no modo como as mesmas são utilizadas. Aqueles com maior freqüência de uso no domínio de fraudes são:

a) *Supervised Learning*: (o mais usado) é o método onde as respostas do algoritmo para cada padrão de entrada são diretamente comparadas com as respostas esperadas, sendo dado um *feedback* para o algoritmo a fim de que ele possa corrigir os possíveis erros (PHUA, 2003; BOLTON; HAND, 2002);

b) *Unsupervised Learning*: é o método onde o algoritmo, por ele mesmo, procura descobrir correlações e similaridades entre os padrões de entrada dos conjuntos treinados, a fim de agrupá-los em diferentes clusters. Não há *feedbacks* do ambiente para que haja comparação de respostas (PHUA, 2003; BOLTON; HAND, 2002);

c) *Score-based*: é o método que usa números em uma faixa especificada, os quais indicam, com risco relativo, que uma instância em particular pode ser fraudulenta, a fim de estabelecer um ranking entre as instâncias (PHUA, 2003);

d) *Rule-based*: é o método que usa regras da forma BODY -> HEAD, onde BODY descreve as condições sob as quais as regras são geradas e HEAD é tipicamente uma *class label* (identificação da classe para a qual as instâncias são referenciadas) (PHUA, 2003).

O método mais utilizado é *Supervised Learning* conforme consta nos trabalhos de Phua (PHUA, 2003; PHUA *et al.*, 2005) e de Bolton e Hand (2002), havendo variações como

*Hybrid Learning* (PHUA *et al.*, 2005; BOLTON; HAND, 2002), onde inicialmente é aplicado o método *Unsupervised* para gerar *clusters*, e depois é aplicado o método *Supervised* para a descrição destes *clusters*.

Outro ponto a ser tratado em trabalhos futuros é uma maior caracterização dos conceitos e relações presentes na ontologia de "Problemas de Fraude", incluindo atributos e instâncias, as quais darão maior completeza à base de conhecimentos atual, aumentando também a precisão da mesma.

Por fim, como em qualquer atividade, é necessário que ocorra a efetiva utilização das ontologias disponibilizadas por esta dissertação, a fim de garantir a evolução das mesmas como ferramentas importantes no domínio de fraudes.

## 5.4 ANÁLISE COMPARATIVA ENTRE TRABALHOS CORRELATOS

Apesar da questão apresentada por Provost (PROVOST *apud* BOLTON; HAND, 2002) no que diz respeito à inexistência de estudos e outros mecanismos que ajudem na determinação de uma relação adequada entre técnicas para detecção de fraudes e os domínios sobre os quais elas devem ser empregadas, durante a pesquisa para esta dissertação foram encontradas várias referências tratando sobre este tema. Em especial, cinco surveys, que foram fundamentais na construção das taxonomias que compõem a base de conhecimentos aqui apresentada.

Os quatro *surveys* contidos nos trabalhos de Abbott *et al.* (1998), de Phua (PHUA *et al.*, 2005; PHUA, 2003) e de Bolton e Hand (2002), procuram identificar técnicas para detecção automáticas de fraudes com maior frequência de uso, suas descrições e os domínios aos quais estão relacionadas. Eles fazem um inventário preciso, a partir do qual é possível construir relações, associando uma técnica a um domínio, procurando, deste modo, identificar informações que sugerem porque a mesma foi utilizada.

Mais recentemente, Yufeng *et al.* (2004) e um grupo de pesquisadores da universidade de Virgínia, nos Estados Unidos, e da universidade de Tatung, em Taiwan, desenvolveram um *survey*, procurando associar domínio de fraude e técnicas de detecção. Nele estão identificadas as três áreas com maior concentração de problemas de fraudes nos Estados Unidos e em Taiwan: cartões de crédito (*Credit Card Fraud*), invasão de sistemas computacionais (*Computer Intrusion*) e telecomunicações (*Telecomm Fraud*); e as técnicas para detecção geralmente utilizadas em cada um deles.

Estes cinco trabalhos foram fundamentais na construção das ontologias presentes nesta dissertação.

# REFERÊNCIAS

ABBOTT, D. W.; MATKOVSKY, I. P.; ELDER IV, J F. An evaluation of high-end data mining tools for fraud detection. In: IEEE INTERNATIONAL CONFERENCE ON SYSTEM, MAN, AND CYBERNETICS, 4., 1998, Tóquio, Japão. **Proceedings …,** v. 3, p. 2836 - 2841. Disponível em: <http://www.citeulike.org/user/robincha/article/940990>. Acesso em: 23 jun. 2007.

ABIDOGUN, O. A. **Data mining, fraud detection and mobile telecommunications:** call pattern analysis with unsupervised neural networks. 2005. Disponível em: <www.uwc.ac.za/library/theses/>. Acesso em: 20 ago. 2006.

ALEXOPOULOS, P. et al **Towards a generic fraud ontology in e-government**. 2006. Disponível em: <www.iwebcare.iisa-innov.com>. Acesso em: 13 jan. 2007.

ALLEN, G. B. **The fraud identification handbook.** Highlands Ranch, EUA: Pp Preventive Press, 1999.

ALLEMBERG, D.; HENDLER, J. **Semantic web for the working ontologist.** Burlington, Canadá: Morgan Kaufmann Publishers, 2008.

ARPÍREZ, J. *et al.* **WebODE:** a scalable workbench for ontological engineering. 2003. Disponível em: <http://webode.dia.fi.upm.es/WebODEWeb/index.html>. Acesso em: 04 mar. 2007.

BARBIERI, C. **BI – Business Intelligence:** modelagem & tecnologia. Rio de Janeiro, Brasil: Axcel Books do Brasil Editora, 2001.

BOLTON, R. J.; HAND, D. J. Statistical fraud detection. **Statistical Science**, Filadélfia, EUA, v. 17, n. 3, p. 235-255, 10 ago. 2002.

BRAUSE, R.; LANGSDORF, T.; HEPP, M. **Neural data mining for credit card fraud detection**. 1999. Disponível em: <www.citeseer.ist.psu.edu/>. Acesso em: 11 ago. 2006.

BURGE, P.; SHAWE-TAYLOR, J. **Detecting cellular fraud using adaptive prototypes**. The ACM Library Digital. 1997. Disponível em: <https://www.aaai.org/Papers/Workshops/1997/WS-97-07/WS97-07-002.pdf>. Acesso em: 11 ago. 2006.

CAHILL, M. H. *et al.* **Detecting fraud in the realword**. 2000. Disponível em: <ftp://cm.bell-labs.com/cm/ms/departments/sia/doc/hmds.pdf>. Acesso em: 13 ago. 2006.

CHIU, C.; TSAI, C. A. Web services-based colaborative scheme for credit card fraud detection. In: IEEE INTERNATIONAL CONFERENCE 2004, 2004, Nova Orleans, EUA. **Proceedings…**v. 1, p. 177-181. Disponível em: <www2.computer.org/portal/web/csdl/doi/10.1109/EEE.2004.1287306>. Acesso em: 20 ago. 2006.

CORCHO, O.; FERNÁNDEZ-LÓPEZ, M.; GÓMEZ-PÉREZ, A.. Ontological engineering: principles, methods, tools and languages. In: CALERO, C.; RUIZ, F.; PIATTINI, M. **Ontologies for software engineering and software technology.** Berlin, Alemanha: Springer-Verlag, 2006. p. 1-39.

CORCHO, O. *et al.* **Building legal ontologies with METHONTOLOGY and WebODE.** 2003. Disponível em: <www.springerlink.com/content/cvkt0nql0t97xurb/>. Acesso em: 14 jun. 2007.

DHAR, V.; STEIN, R. **Seven methods for transforming corporate data into business intelligence**. Nova Jersey, EUA: Prentice Hall, 1997.

ECONOMIST INTELLIGENCE UNIT AND KROLL (Org.). **Global fraud report:** annual edition 2007/2008. Londres, 2007.

FALBO, R. A.; S. MENEZES, C.Experiences in using a method for building domain ontologies. In: SIXTEENTH INTERNATIONAL CONFERENCE ON SOFTWARE ENGINEERING & KNOWLEDGE ENGINEERING (SEKE'04), 16., 2004, Alberta, Canadá. **Proceedings...** Alberta, Canadá: International Workshop On Ontology In Action, Oia, 2004. v. 1, p. 474 - 477. Disponível em: <www.inf.ufes.br/~falbo/download/pub/2004-OIA.pdf>. Acesso em: 20 dez. 2006.

FALBO, R. A.; MENEZES, C. S.; ROCHA, A. R. C. A Systematic approach for building ontologies. In: PROGRESS IN ARTIFICIAL INTELLIGENCE - IBERAMIA, 6., 1998, Lisboa, Portugal. **Proceedings …** Berlin, Alemanha: Springer-verlag, 1998. v. 1484, p. 349 - 360.

FARQUHAR, A.; FIKES, R.; RICE, J. **The ontolingua server:** a tool for collaborative ontology construction. 1996. Disponível em: <ksl.stanford.edu/KSL_Abstracts/KSL-96-26.html>. Acesso em: 20 dez. 2006.

FAWCETT, T.; PROVOST, F. **Adaptive fraud detection**: data mining and knowledge discovery. Boston: Kluwer Academic Publishers, 1997.

FEDERAL BUREAU OF IVESTIGATION – FBI. **Insurance fraud**. 2006. Disponível em: <denver.fbi.gov/documents/insurance_trifold_final.pdf>. Acesso em: 12 jun. 2007.

FERNÁNDEZ-LÓPEZ, M.; GÓMEZ-PÉREZ, A. Overview of methodologies for building ontologies. In: IJCAI-99 WORKSHOP ON ONTOLOGIES AND PROBLEM-SOLVING METHODS: LESSONS AND FUTURE TRENDS, 16., 1999, Estocolmo, Suécia. **Proceedings …** Estocolmo: CEUR Publications, 1999. p. 33 - 37.

GANGEMI, A.; PISANELLI, D. M.; STEVE, G. **An overview of the ONIONS project:** applying ontologies to the integration of medical terminologies. 1999 , Elsevier Science Publishers B.V., E.U.A. Disponível em: <http://reference.kfupm.edu.sa/content/o/v/an_overview_of_the_onions_project__apply_54699.pdf>. Acesso em: 10 out. 2006.

GOLDSCHMIDT, R.; PASSOS, E. **Data mining**: um guia prático. Rio de Janeiro, Brasil: Elsevier Editora, 2005.

GÓMEZ-PÉREZ, A.; FERNÁNDEZ-LÓPEZ, M.; CORCHO, O. **Ontological engineering**. London: Springer-Verlag, 2004.

GROTH, R. **Data mining:** building competitive advantage. New Jersey: Prentice Hall PTR, 2000.

GRUBER, T. R. **Toward Principles for the design of ontologies used for konowledge sharing.** 1993a. Disponível em: <http://iir.ruc.edu.cn/pdf/Toward%20Principles%20for%20the%20Design%20of%20Ontologies.pdf>. Acesso em: 23 ago. 2007.

GRUBER, T. R. A Translation approach to portable ontology specifications. **Knowledge Acquisition**, Londres, Inglaterra, v. 5, p.199-220, 01 jun. 1993. 1993b.

GRÜNINGER, M.; FOX, M. S.. Methodology for the design and evaluation of ontologies. In: IJCAI95 WORKSHOP ON BASIC ONTOLOGICAL ISSUES IN KNOWLEDGE SHARING, 14., 1995, Montreal, Canadá. **Proceedings …** Montreal, Canadá: IJCAI, 1995. v. 6, p. 1 - 10. Disponível em: <http://ijcai.org/Past%20Proceedings/IJCAI-95-VOL%201/content/content.htm>. Acesso em: 20 ago. 2007.

JEN, N. C.; JASON, S. C. Topical clustering of MRD senses based on information retrieval techniques. **Computational Linguistics**, Cambridge, EUA, v. 24, p.61-95, 01 mar. 1998.

KIMBALL, R. **The data warehouse toolkit**. Nova York, EUA: John Wiley & Sons INC., 1996.

LACY, L.W. **OWL:** representing information using the web ontology language. Victoria: Trafford Publishing, 2005.

LI, Y. *et al.* Data mining ontology development for high user usability. **Wuhan University Journal Of Natural Sciences,** Hubei, China, p. 10-14. 11 jan. 2006. Disponível em: <http://d.wanfangdata.com.cn/Periodical_whdxxb-e200601011.aspx>. Acesso em: 20 ago. 2007.

MARTINS, L. G. A. **Fundamentos básicos de lógica proposicional**. 2007. Disponível em: <www.facom.ufu.br/~gustavo/Logica/ LogicaProposicional.pdf>. Acesso em: 11 abr. 2009

NILES, I.; PEASE, A.. Towards a Standard Upper Ontology. In: INTERNATIONAL CONFERENCE ON FORMAL ONTOLOGY IN INFORMATION SYSTEMS, 2., 2001, Maine, EUA. **Proceedings ...** Nova York: Acm Publisher, 2001. p. 2 - 9. Disponível em: <http://portal.acm.org/citation.cfm?id=505168.505170>. Acesso em: 20 jul. 2007.

NOY, N. F.; MCGUINNESS, D. L. **Ontology development 101:** a guide to creating your first ontology. 2001. Disponível em: <www.ksl.stanford.edu/people/dlm/papers/ontology101/ontology101-noy-mcguinness.html>. Acesso em: 13 mar. 2007

NOY, N. F.; MUSEN, M. A. PROMPT: Algorithm and tool for automated ontology merging and alignment. In: NATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE (AAAI-2000), 7., 2000, Austin, EUA. **Proceedings … .** Boston, EUA: AAAI Press, 2000. p.

163-169. Disponível em: <dit.unitn.it/~p2p/RelatedWork/Matching/SMI-2000-0831.pdf>. Acesso em: 21 ago. 2006.

PARODI, L. **Manual das fraudes**. Rio de Janeiro, Brasil: Brasport Livros e Multimídia Ltda., 2005.

PHUA, C. W. C. **Investigative data mining in fraud detection**. 2003. Disponível em: <bsys.monash.edu.au>. Acesso em: 20 dez. 2006.

PHUA, C. *et al.* A comprehensive survey of data mining : based fraud detection research. **Artificial Intelligence Review**, Amsterdan, Holanda, n. , p.34-39, 15 fev. 2005. Disponível em: <http://clifton.phua.googlepages.com/fraud-detection-survey.pdf>. Acesso em: 23 ago. 2006.

PROTÉGÉ. **The protégé ontology editor and knowledge acquisition system.** 2005. Disponível em: <protege.stanford.edu>. Acesso em: 13 out. 2006.

RECTOR, A. **Introduction to ontologies & OWL**. 2005. Disponível em: <www.co-ode.org?resources/tutorials/intro/slides/Introduction.ppt>'. Acesso em: 05 jul. 2007.

SILVERSTONE, H.; DAVIA, H. R. **Fraud 101:** techniques and strategies for detection. New Jersey, EUA: John Wiley & Sons, 2005.

SONG, L.; BROWN, D. E. **An outlier-based data association method for linking criminal incidents.** Elsevier Science B.V. 2004. Disponível em: <psu.edu/doi:10.1016/j.dss.2004.06.005>. Acesso em: 13 out. 2006.

STUDER, R.; BENJAMIN, V. R.; FENSEL, D. **Knowledge Engineering: Principles And Methods.** Elsevier Science B.V. 1998. Disponível em: <protege.stanford.edu>. Acesso em: 13 out. 2006.

TANIGUCHI, M. *et al*. Fraud detection in communications networks using neural and probabilistic methods. In: IEEE INTERNATIONAL CONFERENCE ON ACOUSTICS, SPEECH AND SIGNAL PROCESSING, 17., 1998, Seattle, EUA. **Proceedings …** Nova Jersey, EUA: Ieee Press, 1998. v. 6, p. 2043-2046. Disponível em: <http://www.cis.hut.fi/jhollmen/Publications/icassp98.pdf>. Acesso em: 11 ago. 2006.

TUDORACHE, T. Representation and management of reified relationships in protégé. In: PROTÉGÉ CONFERENCE, 7., 2004, **Annals…** Maryland, EUA Palo Alto, EUA: University Of Stanford, 2004. p. 105 - 109. Disponível em: <protege.stanford.edu/conference/2004/abstracts/Tudorache.pdf>. Acesso em: 02 set. 2006.

USCHOLD, M.; GRÜNINGER, M. Ontologies: principles, methods and applications. **Knowledge Engineering Review**, Cambridge, Inglaterra, v. 11, n. 2, p.10-14, 0 jun. 1996. Disponível em: <https://eprints.kfupm.edu.sa/55793/1/55793.pdf>. Acesso em: 12 dez. 2006.

USCHOLD, M.; KING, M. Towards a methodology for building ontologies. In: IJCAI-95 WORKSHOP ON BASIC ONTOLOGICAL ISSUES IN KNOWLEDGE SHARING, 14., 1995, Montreal, Canadá. **Proceedings...** Montreal, Canadá: Morgan Kaufmann, 1995. p. 6.1 -

6.10. Disponível em: <http://citeseerx.ist.psu.edu/10.1.1.55.5357.pdf>. Acesso em: 11 jul. 2006.

WILHELM, W. K. The fraud management lifecycle theory: a holistic approach to fraud management. **Journal Of Economic Crime Management Spring,** Utica, EUA, p. 2-39. 10 abr. 2004. Disponível em: <https://library.utica.edu/academic/institutes/ecii/publications/articles/BA309CD2-01B6-DA6B-5F1DD7850BF6EE22.pdf>. Acesso em: 12 jul. 2006.

W3C WORKING GROUP (Org.). **Defining N-ary relations on the semantic web.** 2006. Disponível em: <www.w3.org/TR/2006/NOTE-swbp-n-aryRelations-20060412/>. Acesso em: 13 mar. 2007.

YUFENG, K. *et al.* Survey of fraud detection techniques. In: NETWORKING, SENSING AND CONTROL, IEEE INTERNATIONAL CONFERENCE, 1., 2004, Taipei, Taiwan. **Proceedings …** Nova York, EUA: IEEE Press, 2004. v. 2, p. 749 - 754. Disponível em: <http://ieeexplore.ieee.org/servlet/opac?punumber=9086>. Acesso em: 23 mar. 2004.

ZADEH, L. A. From computing with numbers to computing withwords. **International Journal of applied mathematics and computer science (amcs)**, v.12, n.3, p. 307–324. 05 fev. 2002. Disponível em: <www-bisc.cs.berkeley.edu/ZadehCW2002.pdf>. Acesso em: 20.07.2009.

ZHANG, S.; ZHANG, C.; YU, J. X. **An efficient strategy for mining exceptions in multi-databases**. Elsevier Science B.V. 2004. Disponível em: <www-staff.it.uts.edu.au/~zhangsc/scpaper/inszzyu.pdf>. Acesso em: 20.07.2006.

ZHAO, G.; LEARY, R. **Topical ontology of fraud** : release report of FFPOIROT deliverable 2.3, Information Society Technologies. Bruxelas, Bélgica, 2005. Disponível em: <starpc15.vub.ac.be/Publications/ffpoirot.D6.8.AKEMinPOIROT.pdf>. Acesso em: 12.12.2007.

ZHAO, G.; MEERSMAN, R. **Towards a topical ontology of fraud**. Disponível em: <www.springerlink.com/content/7564777715q6gw68/>. 2006. Acesso em: 23.03.2007.

# APÊNDICE A - Lista de termos para as ontologias técnica para detecção automática de fraudes e problemas de fraude

| Nome | Acrônimo | Descrição | Tipo |
|---|---|---|---|
| Técnica | *Technique* | Técnica automática para detecção de informações, notadamente àquelas referentes a fraudes. | Conceito |
| Raciocínio Baseado em Casos | *CBR* | Técnica que se aproveita dos conhecimentos prévios para resolver um determinado problema. Cada tentativa de solução passada é armazenada como um registro (caso), formando uma base de casos (case base), a qual se torna então um modelo. | Indivíduo |
| Suporte a Decisão Orientada a Dados | *DDDS* | Designativo genérico para englobar as tecnologias de Data Warehousing e de aplicações OLAP (On-Line Analytical Programming).<br>Com a tecnologia de Data Warehousing é possível armazenar em um único local, séries de dados oriundas de fontes distintas, integrá-las, e então manipulá-los através do uso de aplicações baseadas na tecnologia OLAP, com o objetivo de se obter conhecimentos para tomada de decisões. | Indivíduo |
| Lógica Fuzzy | *FL* | Técnica que simula um método de raciocínio, o qual permite descrições de regras. A mesma pode descrever um particular fenômeno ou processo, linguisticamente e então, representá-lo através de um pequeno número de regras muito flexíveis, montando uma base de conhecimento. | Indivíduo |
| Algoritmo Genético | *GA* | Técnica que envolve o uso de três componentes: um conjunto de variáveis, utilizadas para descrever os vários aspectos de problema específico; um conjunto de restrições (*constraints*), utilizadas para restringir os valores permitidos para cada uma das variáveis; e, um conjunto de objetivos, que definem os resultados esperados considerando o universo de dados disponíveis. | Indivíduo |

| Nome | Acrônimo | Descrição | Tipo |
|---|---|---|---|
| Aprendizagem de Máquina/Algoritmo de Particionamento Recursivo | *ML* | Técnica que cria regras e árvores de regras, através do uso de pesquisa heurística sobre um conjunto de dados a fim de obter relacionamentos e padrões estatísticos, e assim, definir cluster de registros em categorias específicas. Dentre os algoritmos baseados na técnica de ML, um dos mais recorrentes é o de Particionamento Recursivo ou Recursive Partitioning Algorithm, o qual "aprende" a partir dos dados, de modo similar à técnica de NN, mas tentando encontrar relacionamentos explícitos como regras, tal qual a técnica RBS. | Indivíduo |
| Redes Neurais | *NN* | Técnica é utilizada para gerar modelos (generalizações) a partir de um conjunto de dados, completo ou incompleto, consistente ou inconsistente (ruidoso), procurando "aprender" diretamente da massa de dados, buscando obter padrões através de exames repetidos dos mesmos, da busca de relacionamento entre eles, da construção automática de modelos, e da correção repetida destes modelos caso sejam identificados erros. | Indivíduo |
| Sistema Baseado em Regras | *RBS* | Técnica visa estabelecer o que é verdadeiro ou falso em relação a uma determinada afirmação. Tipicamente, armazena fatos referentes à solução heurística de problemas, em uma base especial chamada de base de regras (*rule base*). Os fatos são armazenados sob a forma de regras do tipo "IF-THEN", e os mesmos são necessários para resolver problemas quando são confrontados com dados. | Indivíduo |
| Dimensão | *Dimension* | Contêm as dimensões utilizadas para avaliação de técnicas para detecção automática de fraudes, conforme abordagem proposta por Dhar e Stein (1997). | Conceito |
| Engenharia | *Engineering* | Dimensão orientada para a avaliação dos custos envolvidos no longo prazo (manutenção, atualização, modificação). | Conceito |
| Logística | *Logistical* | Dimensão orientada para os requisitos logísticos demandados (desenvolvimento, programação, orçamento). | Conceito |
| Qualidade | *Quality* | Dimensão voltada para a avaliação dos recursos humanos e de infra-estruturas disponíveis na organização. | Conceito |

| Nome | Acrônimo | Descrição | Tipo |
|---|---|---|---|
| Recurso | *Resource* | Dimensão orientada para avaliação dos requisitos disponíveis no ambiente de processamento. | Conceito |
| Compacidade | *Compactness* | Refere-se ao tamanho (quantidade em bytes) de uma técnica, para que possa ser implementada. | Conceito |
| Facilidade de Uso | *Ease_of_Use* | Refere-se ao nível de complexidade de uma técnica, para que a mesma possa ser usada diariamente pelos especialistas em negócio. | Conceito |
| Encapsulamento | *Embeddability* | Refere-se ao nível de facilidade para que uma técnica possa ser incorporada à infraestrutura de uma organização. | Conceito |
| Flexibilidade | *Flexibility* | Refere-se ao nível de facilidade que uma técnica possa oferece, para que as relações entre variáveis ou entre variáveis e os respectivos domínios possam ser alteradas, ou os objetivos estabelecidos possam ser modificados. | Conceito |
| Escalabilidade | *Scalability* | Refere-se à capacidade de uma determinada técnica, em suportar a adição de mais variáveis ao problema ou o incremento da faixa de valores que as variáveis possam vir a assumir. | Conceito |
| Facilidade computacional | *Computing_Ease* | Refere-se ao grau de implementação para uma técnica sem que haja requisição de hardware e software específicos. | Conceito |
| Tempo de desenvolvimento | *Development_Time* | Refere-se ao tempo que uma organização gastaria para desenvolver uma solução baseada numa técnica específica. | Conceito |
| Independência de especialistas | *Independence_from_Experts* | Refere-se ao grau de independência para projetar, construir e testar uma solução baseada numa técnica sem uso de especialistas. | Conceito |
| Precisão | *Accuracy* | Refere-se ao grau de precisão oferecido por uma técnica para as saídas geradas, em termos de correção ou de melhor decisão ou de opção. | Conceito |
| Explicabilidade | *Explainability* | Refere-se ao nível de descrição que uma técnica oferece para o processo pelo qual uma conclusão foi alcançada. | Conceito |
| Tempo de resposta | *Response_Time* | Refere-se ao tempo gasto por uma técnica para completar a análise no nível desejado de precisão. | Conceito |
| Curva de aprendizagem | *Learning_Curve* | Refere-se ao grau que uma organização deve alcançar para se tornar suficientemente competente na solução de problemas usando uma determinada técnica. | Conceito |

| Nome | Acrônimo | Descrição | Tipo |
|---|---|---|---|
| Tolerância a complexidade | *Tolerance_for_Complexity* | Refere-se ao grau de comprometimento no nível de qualidade oferecido por uma técnica, devido ao número de interações entre os vários componentes do processo modelado (por exemplo, muitas interações não-lineares entre variáveis) ou a complexidade do conhecimento para modelar um processo. | Conceito |
| Tolerância a dados inconsistentes | *Tolerance_for_Noise_in_Data* | Refere-se ao grau de comprometimento no nível de qualidade oferecido por uma técnica, notadamente no que se refere a precisão, devido a inconsistência no conteúdo dos dados. | Conceito |
| Tolerância a dados incompletos | *Tolerance_for_Sparse_Data* | Refere-se ao grau de comprometimento no nível de qualidade oferecido por uma técnica, devido à falta de dados ou pela existência de dados incompletos. | Conceito |
| Ambiente | *Environment* | Refere-se aos fatores ambientais que tem impacto sobre as dimensões utilizadas para avaliação de técnicas para detecção automática, conforme abordagem proposta por Dhar e Stein (1997). | Conceito |
| Quantidade de dados | *Amount_of_Data* | Fator ambiental relacionado com a quantidade de bytes da amostra de dados presente na fonte de informação. | Conceito |
| Entendimento do negócio | *Business_Understand* | Fator ambiental relacionado com o grau de complexidade do domínio ao qual se relaciona a amostra de dados presente na fonte de informação. | Conceito |
| Complexidade do problema de negócio | *Complexity_of_Business_Problem* | Fator ambiental relacionado com o grau de complexidade do problema referente ao domínio sob o qual se encontra a amostra de dados envolvida na fonte de informação. | Conceito |
| Complexidade dos dados | *Complexity_of_Data* | Fator ambiental relacionado com o grau de complexidade da amostra de dados envolvida na fonte de informação. | Conceito |
| Complexidade da infraestrutura | *Complexity_of_Infrastructure* | Fator ambiental relacionado com o grau de complexidade da infraestrutura requerida para o processamento da amostra de dados presente na fonte de informação. | Conceito |
| Entendimento dos dados | *Data_Understand* | Fator ambiental relacionado com o grau de complexidade da amostra de dados presente na fonte de informação. | Conceito |

| Nome | Acrônimo | Descrição | Tipo |
|---|---|---|---|
| Infraestrutura disponível | *Infrastructure_Available* | Fator ambiental relacionado com o grau de complexidade da infraestrutura disponível para o processamento da amostra de dados envolvida na presente na fonte de informação. | Conceito |
| Dados limpos | *Scrubbed_Data* | Fator ambiental relacionado com o grau de eficiência da representação e do conteúdo da amostra de dados (uso de matrizes esparsas, redução de redundância, redução de inconsistências) presente na fonte de informação. | Conceito |
| Dados atualizados | *Update_of_Data* | Fator ambiental relacionado com o nível de atualização da amostra de dados envolvida na fonte de informação. | Conceito |
| Relação de Dimensão de Avaliação | *Dimension_Relation* | Classe contendo a relação entre dimensões e fatores ambientais, utilizada para avaliação de uma técnica automática de detecção de fraudes, conforme abordagem proposta por Dhar e Stein (1997). | Conceito |
| Relacao de Dimensao 01 | *Dimension_Relation_01* | Relação referente à dimensão, precisão alta, combinada com quantidade grande de dados. | Indivíduo |
| Relacao de Dimensao 02 | *Dimension_Relation_02* | Relação referente à dimensão, flexibilidade alta. | Indivíduo |
| Relacao de Dimensao 03 | *Dimension_Relation_03* | Relação referente à dimensão, independência alta de especialistas, combinada com complexidade baixa ou moderada do problema de negócio. | Indivíduo |
| Relacao de Dimensao 04 | *Dimension_Relation_04* | Relação referente à dimensão, tempo de resposta baixo, combinada com complexidade baixa ou moderada do problema de negócio e quantidade de dados pequena ou média e disponibilidade grande de infra-estrutura. | Indivíduo |
| Relacao de Dimensao 05 | *Dimension_Relation_05* | Relação dimensão referente à escalabilidade alta. | Indivíduo |
| Relacao de Dimensao 06 | *Dimension_Relation_06* | Relação referente à dimensão, facilidade alta de computação. | Indivíduo |
| Relacao de Dimensao 07 | *Dimension_Relation_07* | Relação referente à dimensão, facilidade alta de uso, combinada com complexidade baixa ou moderada de infra-estrutura. | Indivíduo |
| Relacao de Dimensao 08 | *Dimension_Relation_08* | Relação referente à dimensão, tempo baixo de resposta. | Indivíduo |
| Relacao de Dimensao 09 | *Dimension_Relation_09* | Relação referente à dimensão, precisão alta, combinada com entendimento alto do negócio. | Indivíduo |

| Nome | Acrônimo | Descrição | Tipo |
|---|---|---|---|
| Relacao de Dimensao 10 | *Dimension_Relation_10* | Relação referente à dimensão, compacidade alta. | Indivíduo |
| Relacao de Dimensao 11 | *Dimension_Relation_11* | Relação referente à dimensão, tolerância alta a complexidade, combinada com complexidade baixa ou moderada dos dados e complexidade baixa ou moderada da infra-estrutura. | Indivíduo |
| Relacao de Dimensao 12 | *Dimension_Relation_12* | Relação referente à dimensão, precisão alta, combinada com complexidade baixa do problema de negócio. | Indivíduo |
| Relacao de Dimensao 13 | *Dimension_Relation_13* | Relação referente à dimensão, tempo baixo de desenvolvimento, combinada com complexidade baixa ou moderada do problema de negócio. | Indivíduo |
| Relacao de Dimensao 14 | *Dimension_Relation_14* | Relação referente à dimensão, encapsulamento alto, combinada com complexidade baixa ou moderada do problema de negócio e complexidade baixa ou moderada de infra-estrutura. | Indivíduo |
| Relacao de Dimensao 15 | *Dimension_Relation_15* | Relação referente à dimensão, flexibilidade alta, combinada com complexidade baixa ou moderada dos dados e complexidade baixa ou moderada de infra-estrutura. | Indivíduo |
| Relacao de Dimensao 16 | *Dimension_Relation_16* | Relação referente à dimensão, tempo baixo de resposta, combinada com complexidade baixa do problema de negócio | Indivíduo |
| Relacao de Dimensao 17 | *Dimension_Relation_17* | Relação referente à dimensão, precisão alta, combinada com complexidade baixa ou moderada dos dados e complexidade baixa ou moderada de infra-estrutura. | Indivíduo |
| Relacao de Dimensao 18 | *Dimension_Relation_18* | Relação referente à dimensão, encapsulamento alto. | Indivíduo |
| Relacao de Dimensao 19 | *Dimension_Relation_19* | Relação referente à dimensão, explicabilidade alta, combinada com complexidade baixa dos dados. | Indivíduo |
| Relacao de Dimensao 20 | *Dimension_Relation_20* | Relação referente à dimensão, flexibilidade alta, combinada com grande quantidade de dados. | Indivíduo |
| Relacao de Dimensao 21 | *Dimension_Relation_21* | Relação referente à dimensão, escalabilidade alta, combinada com complexidade baixa dos dados e quantidade pequena de dados. | Indivíduo |
| Relacao de Dimensao 22 | *Dimension_Relation_22* | Relação referente à dimensão, precisão alta, combinada com complexidade baixa ou moderada do problema de negócio. | Indivíduo |
| Relacao de Dimensao 23 | *Dimension_Relation_23* | Relação referente à dimensão, flexibilidade alta, combinada com entendimento alto dos dados. | Indivíduo |

| Nome | Acrônimo | Descrição | Tipo |
|---|---|---|---|
| Relacao de Dimensao 24 | *Dimension_Relation_24* | Relação referente à dimensão, independência alta de especialistas. | Indivíduo |
| Relacao de Dimensao 25 | *Dimension_Relation_25* | Relação referente à dimensão, tolerância alta à complexidade. | Indivíduo |
| Relacao de Dimensao 26 | *Dimension_Relation_26* | Relação referente à dimensão, tolerância alta a dados inconsistentes, combinada com entendimento moderado ou grande dos dados. | Indivíduo |
| Relacao de Dimensao 27 | *Dimension_Relation_27* | Relação referente à dimensão, tolerância alta a dados incompletos, combinada com entendimento moderado ou grande dos dados. | Indivíduo |
| Relacao de Dimensao 28 | *Dimension_Relation_28* | Relação referente à dimensão, tempo baixo de desenvolvimento, combinada com quantidade pequena ou média de dados | Indivíduo |
| Relacao de Dimensao 29 | *Dimension_Relation_29* | Relação referente à dimensão, explicabilidade alta. | Indivíduo |
| Compacidade alta | *Compactness_High* | | Indivíduo |
| Compacidade moderada | *Compactness_Moderate* | | Indivíduo |
| Compacidade baixa | *Compactness_Low* | | Indivíduo |
| Facilidade de Uso alta | *Ease_of_Use_High* | | Indivíduo |
| Facilidade de Uso moderada | *Ease_of_Use_Moderate* | | Indivíduo |
| Facilidade de Uso baixa | *Ease_of_Use_Low* | | Indivíduo |
| Encapsulamento alto | *Embeddability_High* | | Indivíduo |
| Encapsulamento moderado | *Embeddability_Moderate* | | Indivíduo |
| Encapsulamento baixo | *Embeddability_Low* | | Indivíduo |
| Flexibilidade alta | *Flexibility_High* | | Indivíduo |
| Flexibilidade moderada | *Flexibility_Moderate* | | Indivíduo |
| Flexibilidade baixa | *Flexibility_Low* | | Indivíduo |
| Escalabilidade alta | *Scalability_High* | | Indivíduo |
| Escalabilidade moderada | *Scalability_Moderate* | | Indivíduo |
| Escalabilidade baixa | *Scalability_Low* | | Indivíduo |
| Facilidade computacional alta | *Computing_Ease_High* | | Indivíduo |
| Facilidade computacional moderada | *Computing_Ease_Moderate* | | Indivíduo |
| Facilidade computacional baixa | *Computing_Ease_Low* | | Indivíduo |
| Tempo de desenvolvimento alta | *Development_Time_High* | | Indivíduo |
| Tempo de desenvolvimento moderado | *Development_Time_Moderate* | | Indivíduo |
| Tempo de desenvolvimento baixo | *Development_Time_Low* | | Indivíduo |
| Independência de especialistas alta | *Independence_from_Experts_High* | | Indivíduo |
| Independência de especialistas moderada | *Independence_from_Experts_Moderate* | | Indivíduo |

| Nome | Acrônimo | Descrição | Tipo |
|---|---|---|---|
| Independência de especialistas baixa | *Independence_from_Experts_Low* | | Indivíduo |
| Precisão alta | *Accuracy_High* | | Indivíduo |
| Precisão moderada | *Accuracy_Moderate* | | Indivíduo |
| Precisão baixa | *Accuracy_Low* | | Indivíduo |
| Explicabilidade alta | *Explainability_High* | | Indivíduo |
| Explicabilidade moderada | *Explainability_Moderate* | | Indivíduo |
| Explicabilidade baixa | *Explainability_Low* | | Indivíduo |
| Tempo de resposta alto | *Response_Time_High* | | Indivíduo |
| Tempo de resposta moderado | *Response_Time_Moderate* | | Indivíduo |
| Tempo de resposta baixo | *Response_Time_Low* | | Indivíduo |
| Curva de aprendizagem alta | *Learning_Curve_High* | | Indivíduo |
| Curva de aprendizagem moderada | *Learning_Curve_Moderate* | | Indivíduo |
| Curva de aprendizagem baixa | *Learning_Curve_Low* | | Indivíduo |
| Tolerância a complexidade alta | *Tolerance_for_Complexity_High* | | Indivíduo |
| Tolerância a complexidade moderada | *Tolerance_for_Complexity_Moderate* | | Indivíduo |
| Tolerância a complexidade baixa | *Tolerance_for_Complexity_Low* | | Indivíduo |
| Tolerância a dados inconsistentes alta | *Tolerance_for_Noise_in_Data_High* | | Indivíduo |
| Tolerância a dados inconsistentes moderada | *Tolerance_for_Noise_in_Data_Moderate* | | Indivíduo |
| Tolerância a dados inconsistentes baixa | *Tolerance_for_Noise_in_Data_Low* | | Indivíduo |
| Tolerância a dados incompletos alta | *Tolerance_for_Sparse_Data_High* | | Indivíduo |
| Tolerância a dados incompletos moderada | *Tolerance_for_Sparse_Data_Moderate* | | Indivíduo |
| Tolerância a dados incompletos baixa | *Tolerance_for_Sparse_Data_Low* | | Indivíduo |
| Quantidade de dados grande | *Amount_of_Data_Large* | | Indivíduo |
| Quantidade de dados média | *Amount_of_Data_Medium* | | Indivíduo |
| Quantidade de dados pequena | *Amount_of_Data_Small* | | Indivíduo |
| Entendimento do negócio alto | *Business_Understand_High* | | Indivíduo |
| Entendimento do negócio moderado | *Business_Understand_Moderate* | | Indivíduo |
| Entendimento do negócio baixo | *Business_Understand_Low* | | Indivíduo |
| Complexidade do problema de negócio alta | *Complexity_of_Business_Problem_High* | | Indivíduo |
| Complexidade do problema de negócio moderada | *Complexity_of_Business_Problem_Moderate* | | Indivíduo |
| Complexidade do problema de negócio baixa | *Complexity_of_Business_Problem_Low* | | Indivíduo |

| Nome | Acrônimo | Descrição | Tipo |
|---|---|---|---|
| Complexidade dos dados alta | *Complexity_of_Data_High* | | Indivíduo |
| Complexidade dos dados moderada | *Complexity_of_Data_Moderate* | | Indivíduo |
| Complexidade dos dados baixa | *Complexity_of_Data_Low* | | Indivíduo |
| Complexidade da infraestrutura alta | *Complexity_of_Infrastructure_High* | | Indivíduo |
| Complexidade da infraestrutura moderada | *Complexity_of_Infrastructure_Moderate* | | Indivíduo |
| Complexidade da infraestrutura baixa | *Complexity_of_Infrastructure_Low* | | Indivíduo |
| Entendimento dos dados alto | *Data_Understand_High* | | Indivíduo |
| Entendimento dos dados moderado | *Data_Understand_Moderate* | | Indivíduo |
| Entendimento dos dados baixo | *Data_Understand_Low* | | Indivíduo |
| Infraestrutura disponível grande | *Infrastructure_Available_Large* | | Indivíduo |
| Infraestrutura disponível média | *Infrastructure_Available_Medium* | | Indivíduo |
| Infraestrutura disponível pequena | *Infrastructure_Available_Small* | | Indivíduo |
| Dados não limpos | *Scrubbed_Data_No* | | Indivíduo |
| Dados limpos | *Scrubbed_Data_Yes* | | Indivíduo |
| Dados desatualizados | *Update_of_Data_Out_of_Date* | | Indivíduo |
| Dados atualizados | *Update_of_Data_Recent* | | Indivíduo |
| valor para a dimensão precisão (Relação de Dimensão de Avaliação, Precisão) | *accuracy_value* | Valor para dimensão precisão. | Relação |
| valor para a dimensão explicabilidade (Relação de Dimensão de Avaliação, Explicabilidade) | *explainability_value* | Valor para dimensão explicabilidade. | Relação |
| valor para a dimensão tempo de resposta (Relação de Dimensão de Avaliação,Tempo de Resposta ) | *response_time_value* | Valor para dimensão tempo de resposta. | Relação |
| valor para a dimensão escalabilidade (Relação de Dimensão de Avaliação, Escalabilidade) | *scalability_value* | Valor para dimensão escalabilidade. | Relação |
| valor para a dimensão compacidade (Relação de Dimensão de Avaliação, Compacidade) | *compactness_value* | Valor para dimensão compacidade. | Relação |
| valor para a dimensão flexibilidade (Relação de Dimensão de Avaliação, Flexibilidade) | *flexibility_value* | Valor para dimensão flexibilidade. | Relação |

| Nome | Acrônimo | Descrição | Tipo |
|---|---|---|---|
| valor para a dimensão encapsulamento (Relação de Dimensão de Avaliação, Capacidade de encapsulamento) | *embeddability_value* | Valor para dimensão capacidade de encapsulamento. | Relação |
| valor para a dimensão facilidade de uso (Relação de Dimensão de Avaliação, Facilidade de uso) | *ease_of_use_value* | Valor para dimensão facilidade de uso. | Relação |
| valor para a dimensão tolerância a dados inconsistentes (Relação de Dimensão de Avaliação, Tolerância a dados inconsistentes) | *tolerance_for_noise_in_data_value* | Valor para dimensão tolerância a dados inconsistentes. | Relação |
| valor para a dimensão tolerância a dados incompletos (Relação de Dimensão de Avaliação, Tolerância a dados incompletos) | *tolerance_for_sparse_data_value* | Valor para dimensão tolerância a dados incompletos. | Relação |
| valor para a dimensão tolerância a complexidade (Relação de Dimensão de Avaliação, Tolerância a complexidade) | *tolerance_for_complexity_value* | Valor para dimensão tolerância a complexidade. | Relação |
| valor para a dimensão curva de aprendizagem (Relação de Dimensão de Avaliação, Curva de aprendizagem) | *learning_curve_value* | Valor para dimensão tolerância a curva de aprendizagem. | Relação |
| valor para a dimensão independência de especialistas (Relação de Dimensão de Avaliação, Independência de especialistas) | *independence_from_experts_value* | Valor para dimensão independência de especialistas. | Relação |
| valor para a dimensão tempo de desenvolvimento (Relação de Dimensão de Avaliação, Tempo de desenvolvimento) | *development_time_value* | Valor para dimensão tempo de desenvolvimento. | Relação |
| valor para a dimensão facilidade computacional (Relação de Dimensão de Aval., Facilidade computacional) | *computing_ease_value* | Valor para dimensão facilidade computacional. | Relação |
| tem relação (Técnica, Relação de Dimensão de Avaliação) | *has_relation* | Define a relação entre uma técnica de detecção de fraudes e as dimensões e os fatores ambientais, que definem a sua melhor performance. | Relação |

| Nome | Acrônimo | Descrição | Tipo |
|---|---|---|---|
| valor para quantidade de dados (Relação de Dimensão de Avaliação, Quantidade de dados) | *amount_of_data_value* | Valor para fator ambiental quantidade de dados. | Relação |
| valor para entendimento do negócio (Relação de Dimensão de Avaliação, Entendimento do negócio) | *business_understand_value* | Valor para fator ambiental entendimento do negócio. | Relação |
| valor para complexidade do problema de negócio (Relação de Dimensão de Avaliação, Complexidade do problema de negócio) | *complexity_of_business_problem_value* | Valor para fator ambiental complexidade do problema de negócio. | Relação |
| valor para complexidade dos dados (Relação de Dimensão de Avaliação, Complexidade dos dados) | *complex_of_data_value* | Valor para fator ambiental complexidade dos dados. | Relação |
| valor para complexidade da infraestrutura (Relação de Dimensão de Avaliação, Complexidade da infraestrutura) | *complex_of_infrastructure_value* | Valor para fator ambiental complexidade da infraestrutura. | Relação |
| valor para entendimento dos dados (Relação de Dimensão de Avaliação, Entendimento dos dados) | *data_nderstand_value* | Valor para fator ambiental entendimento dos dados. | Relação |
| valor para infraestrutura disponível (Relação de Dimensão de Avaliação, Infraestrutura disponível) | *infrastructure_available_value* | Valor para fator ambiental infraestrutura disponível. | Relação |
| valor para dados limpos (Relação de Dimensão de Avaliação, Dados limpos) | *scrubbed_data_value* | Valor para fator ambiental dados limpos. | Relação |
| valor para dados atualizados (Relação de Dimensão de Avaliação, Dados atualizados) | *update_of_data_value* | Valor para fator ambiental dados atualizados. | Relação |
| Problema de Fraude | *Fraud* | Conceito baseada em taxonomia de problemas de fraude proposta por Allen (1999). | Conceito |
| Indivíduos | *Individuals* | Refere-se a fraudes onde o *perpetrator* (autor) e o *target* (alvo ou vítima) são pessoas. | Conceito |
| Indivíduos e Instituições | *Individuals_and_Institutions* | Refere-se a fraudes onde o *perpetrator* (autor) e o *target* (alvo ou vítima) são pessoas ou organizações. | Conceito |
| Instituições | *Institutions* | Refere-se a fraudes onde o *perpetrator* (autor) e o *target* (alvo ou vítima) são organizações. | Conceito |

| Nome | Acrônimo | Descrição | Tipo |
|---|---|---|---|
| Quebra de Confiança | *Confidence_Games* | Trata do estabelecimento de confiança pelo *perpetrator* (autor) da fraude a fim de subtrair algo de valor de um *target* (alvo ou vítima). Alguns domínios de fraudes correlacionados ao mesmo: falsificação de identidade (*Impersonation* ou *Identity Fraud*); jogo ilegal (*Gambling*); esquemas de fraudes envolvendo cultos religiosos onde idosos (e outros) são vítimas de perdas financeiras (*Cult Fraud*). | Indivíduo |
| Fraude ao Consumidor | *Consumer_Fraud* | Ocorre quando o *perpetrator* comete um ato fraudulento durante a compra de um bem ou de um serviço. Como domínios relacionados, temos como exemplos: fraude com cartão de crédito (*Credit Card Fraud*); fraude na emissão de diplomas e certificados na área de educação (*Education Fraud*); simulação fraudulenta de defeitos em veículos automotores (*Motor Vehicles Fraud*). | Indivíduo |
| Fraude em Adiantamento de Taxa | *Advance_Fee_Fraud* | É descrito como uma fraude onde o *perpetrator* convence o *target* a fazer um pagamento adiantado por uma entrega futura de um bem ou de um serviço, que ou não será entregue ou não será entregue nas condições contratadas. Não há domínios específicos para este tipo de problema de fraude. | Indivíduo |
| Fraude Financeira | *Financial_Fraud* | Problema de fraude concentrado nas áreas bancária, financeira e de seguro. Deste modo, os domínios relacionados são: fraude envolvendo operações e empréstimos bancários (*Banking and Lending Fraud*); fraude envolvendo operações financeiras não bancárias (exceto ativos) (*Financial Statement Fraud*); fraude referente ao mercado de seguro privado (*Insurance Fraud*). | Indivíduo |
| Fraude em Telecomunicações | *TelecommFraud* | (*Telecommunication* (*Telecomm*) *Fraud*) problema de fraude basicamente concentrado em três domínios: fraudes em serviços de telefonia fixa (*Hard-line Telephone Service Fraud*); fraudes em serviços de telefonia móvel ou comunicação móvel (*Wireless Phone Service*); fraudes relacionados com a Internet (*Internet Service Fraud*). | Indivíduo |

| Nome | Acrônimo | Descrição | Tipo |
|---|---|---|---|
| Fraude em Investimentos | *Investment_Fraud* | Normalmente ocorre em três situações: o *perpetrator* vende um bem que não possui ou que não tem a posse; o *perpetrator* deturpa as características, o valor ou retorno potencial de um recurso a ser negociado; o *perpetrator* gerencia mal um recurso para privar o seu proprietário de usufruir do seu potencial de uso ou de investimento. Alguns domínios para este problema de fraude são: fraude envolvendo objetos de arte (*Art Fraud*); fraude relacionada com o mercado de gemas, pedras ou minerais (*Gem and Mineral Fraud*); fraude em operações com papéis de riscos referentes a débitos de organizações privadas e de governos (*Bond Fraud*). | Indivíduo |
| Fraude Científica | *Science_Fraud* | Problema de fraude onde o *perpetrator* ou o target tem origem na área acadêmica. Em geral, assim como o *Telecomm_Fraud*, também está concentrado em três domínios: fraudes relacionadas com a geração de requisições, relatórios e conteúdos falsos (*Criminal Science Fraud*); geração de perdas financeiras para a instituição (governo ou não) a partir de ações de negligência, participação em eventos fictícios ou de pouca significância, não cumprimento de contratos e prazos (*Civil Science Fraud*); e, falta de conduta experimental (projeto experimental intencionalmente falho, uso de experimentos sem conclusão, etc), falta de conduta para publicação (falsificação de dados e de métodos, informação sobre experimentos fantasmas, etc), falsificação de currículo vitae, entre outros (*Ethical Science Fraud*). | Indivíduo |
| Fraude em Operações | *Operation_Fraud* | Tipo de fraude que se dá no universo das operações de uma organização: internamente, quando há o envolvimento de empregados e bens; externamente, quando envolve licitação, contratação e competição. No contexto externo, alguns dos domínios que se aliam a este problema de fraude são: fraude em processos de licitação (*Bidding Fraud*); fraude em contratos, por exemplo, com subfaturamentos (*Contract Fraud*); fraude provocada por um empregado, por exemplo, falsificação de suas credenciais (*Employee or Internal Fraud*). | Indivíduo |

| Nome | Acrônimo | Descrição | Tipo |
|---|---|---|---|
| Fraude Governamental | *Government_Fraud* | Ocorre quando o agente governamental ou é o *perpetrator* (autor) ou é o *target* (alvo ou vítima). Existem dois tipos de domínios para este problema de fraude: aqueles relacionados como os chamados Programas de Direito (*Entitlement Programs*), dentro dos quais temos fraudes em educação envolvendo estudantes fantasmas e desvio de verbas (*Education Fraud*), fraudes em programas sociais e de assistência social (*Welfare Fraud*), fraudes em programas de saúde e de seguro saúde (*Medicare and Medicaid Fraud*), fraudes na previdência social pública (*Social Security Fraud*); e aqueles relacionados com os chamados Programas de Serviços, onde temos fraudes na área alfandegária (*Customs Fraud*), na área de defesa (*Defense Fraud*), na área postal (*Postal Fraud*) entre outras. | Indivíduo |
| Fraude do Interesse Público | *Public_Fraud* | Ocorre quando o agente que presta algum tipo de serviço público (pertencente ao poder público ou não) ou é o *perpetrator* ou é o *target*.<br>Tem como exemplos de domínios: desvio de fundos de caridade (*Charity Fraud*); desvio de valores devidos em taxas e impostos (*Tax Fraud*); fraude no resultado de uma eleição (*Election Fraud*). | Indivíduo |
| necessita valor de precisão (Problema de Fraude, Precisão) | *need_accuracy_value* | Valor que o problema de fraude necessita para precisão. | Relação |
| necessita valor de explicabilidade (Problema de Fraude, Explicabilidade) | *need_explainability_value* | Valor que o problema de fraude necessita para explicabilidade. | Relação |
| necessita valor de tempo de resposta (Problema de Fraude,Tempo de Resposta ) | *need_response_time_value* | Valor que o problema de fraude necessita para tempo de resposta. | Relação |
| necessita valor de escalabilidade (Problema de Fraude, Escalabilidade) | *need_scalability_value* | Valor que o problema de fraude necessita para escalabilidade. | Relação |
| necessita valor de compacidade (Problema de Fraude, Compacidade) | *need_compactness_value* | Valor que o problema de fraude para compacidade. | Relação |
| necessita valor de flexibilidade (Problema de Fraude, Flexibilidade) | *need_flexibility_value* | Valor que o problema de fraude necessita para flexibilidade. | Relação |
| necessita valor de encapsulamento (Problema de Fraude, Capacidade de encapsulamento) | *need_embeddability_value* | Valor que o problema de fraude necessita para capacidade de encapsulamento. | Relação |

| Nome | Acrônimo | Descrição | Tipo |
|---|---|---|---|
| necessita valor de facilidade de uso (Problema de Fraude, Facilidade de uso) | *need_ease_of_use_value* | Valor que o problema de fraude necessita para facilidade de uso. | Relação |
| necessita valor de tolerância a dados inconsistentes (Problema de Fraude, Tolerância a dados inconsistentes) | *need_tolerance_for_noise_in_data_value* | Valor que o problema de fraude necessita para tolerância a dados inconsistentes. | Relação |
| necessita valor de tolerância a dados incompletos (Problema de Fraude, Tolerância a dados incompletos) | *need_tolerance_for_sparse_data_value* | Valor que o problema de fraude necessita para tolerância a dados incompletos. | Relação |
| necessita valor de tolerância a complexidade (Problema de Fraude, Tolerância a complexidade) | *need_tolerance_for_complexity_data_value* | Valor que o problema de fraude necessita para tolerância a complexidade. | Relação |
| necessita valor de curva de aprendizagem (Problema de Fraude, Curva de aprendizagem) | *need_learning_curve _value* | Valor que o problema de fraude necessita para tolerância a curva de aprendizagem. | Relação |
| necessita valor de independência de especialistas (Problema de Fraude, Independência de especialistas) | *need_independence_from_experts_value* | Valor que o problema de fraude necessita para independência de especialistas. | Relação |
| necessita valor de tempo de desenvolvimento (Problema de Fraude, Tempo de desenvolvimento) | *need_development_time_value* | Valor que o problema de fraude necessita para tempo de desenvolvimento. | Relação |
| necessita valor de facilidade computacional (Problema de Fraude, Facilidade computacional) | *need_computing_ease_value* | Valor que o problema de fraude necessita para facilidade computacional. | Relação |

# APÊNDICE B – Códigos das ontologias implementadas

- *Automatic_Technique_Detection.owl*: Técnicas para Detecção Automática de Fraude;
- *Fraud_Problem.owl*: Problemas de Fraude;
- *Fraud_Detection.owl*: Detecção Automática de Fraudes (obtida a partir da atividade de *merging* entre as duas anteriores).

Observação: o código das ontologias está gravado no CD que acompanha a dissertação escrita.

# ANEXO A - Texto referente à revisão sistemática “Técnicas para Detecção Automática de Fraudes: Uma Revisão Sistemática”

Texto referente à revisão sistemática “Técnicas para Detecção Automática de Fraudes: Uma Revisão Sistemática”, sem o apêndice contendo os formulários utilizados na mesma:

## Técnicas para Detecção Automática de Fraudes: Uma Revisão Sistemática

**Reinaldo de F. Almeida**[1]

[1]Mestrado em Sistemas e Computação – Universidade Salvador (UNIFACS)

Salvador – BA – Brasil

**reifa28@gmail.com**

## 1 INTRODUÇÃO

A demanda pela utilização de tecnologias de informação que automatizem os processos de detecção de fraudes tem crescido na mesma proporção em que tem aumentado o nível de complexidade e de extensão das ocorrências de fraudes nos mais diversos domínios da atividade humana.

Pesquisas têm sido produzidas junto a diversas atividades, através de experimentos e estudos de caso, no sentido de identificar quais as melhores tecnologias a serem empregadas para identificar automaticamente, de modo eficaz e eficiente, as ocorrências de fraudes, tanto sob a perspectiva da detecção quanto da prevenção.

Entretanto, apesar da existência de *surveys* identificando o emprego de diversas tecnologias frente a vários domínios, existe uma enorme carência de estudos demonstrando a associação entre o emprego de determinada tecnologia para detecção automática de fraudes com um domínio específico, considerando as características do mesmo e os históricos de resultados.
Nesse sentido, a condução desta revisão sistemática busca efetuar um primeiro levantamento, restringindo a sua abrangência para experimentos ou estudos de casos que registrem o emprego de uma tecnologia específica junto a um domínio específico, a fim de que se possa obter um retrato inicial do emprego de técnicas para detecção automática de fraudes.

# 2 PROTOCOLO DE REVISÃO

O Protocolo de Revisão Sistemática utilizado foi baseado nos modelos propostos por (BIOLCHINI *et al.*, MENDES; KITCHENHAM *apud* MAFRA; TRAVASSOS, 2005).

*Objetivo:*

O Protocolo de Revisão Sistemática visa identificar e avaliar técnicas para Detecção Automática de Fraudes com o propósito de caracterizá-las com respeito à aplicabilidade num contexto de um domínio específico.

*Formulação da Pergunta:*

Quais as técnicas para Detecção Automática de Fraudes para um domínio específico?

1. *Intervenção:* Técnicas para Detecção Automática de Fraudes.
2. *População:* Processos de identificação de fraudes para um domínio específico.
3. *Resultados:* Técnicas para Detecção Automática de Fraudes para um domínio específico.
4. *Aplicação:* Projetos de auditoria para identificação de fraudes para um domínio específico.

*Critérios de Seleção de Fontes:*

1. Disponibilidade de consulta de documentos através da *web*;
2. Disponibilidade de acesso a bases de dados;
3. Presença de mecanismos de busca através de palavras-chave.

*Métodos de Busca de Fontes:*

As fontes serão acessadas apenas pela *web*.

*Palavras-chave:*

Fraud detection, automated fraud detection, automated fraud detection technique, fraud prevention, automated fraud prevention; data mining, data mining application, artificial intelligence.

*Listagem de Fontes:*

Elsevier – *databases;* The ACM Digital Library; Springer Link – *databases*; *web*.

*Tipo de Documentos:*

Documentos descrevendo experimentos ou estudo de caso.

*Idioma dos Documentos:*

Inglês. A opção pelo inglês se deve pela universalidade deste idioma. Em relação ao português, justifica-se a exclusão pela necessidade dos estudos serem passíveis de repetição em diferentes contextos de outros países.

*Critérios para Inclusão e Exclusão de Documentos:*

1. Os documentos devem estar disponíveis na *web;*
2. Os documentos devem apresentar textos completos em formato eletrônico;
3. Os documentos devem estar escritos em inglês;
4. Os documentos devem descrever a execução de experimentos ou estudo de caso, investigando o emprego de uma técnica para Detecção Automática de Fraudes;
5. Os documentos devem descrever a execução de experimentos ou estudo de caso para Detecção Automática de Fraudes para um domínio específico.

A escala envolvida nos cinco critérios é nominal envolvendo duas opções: *sim* ou *não*.

*Processo de Seleção dos Estudos Primários:*

O processo de seleção consistirá dos seguintes passos:

1. Pesquisador executa o processo de busca em cada uma das fontes selecionadas para identificar os documentos que contenham descrições de emprego de técnicas para Detecção Automática de Fraudes;
2. Os documentos encontrados são obtidos da fonte e documentados na lista de documentos encontrados, presente no Formulário de Condução da Revisão;

3. Os documentos encontrados pelo processo de busca são selecionados, através da verificação dos critérios de inclusão e exclusão estabelecidos; a verificação é executada mediante a leitura do *abstract* do documento;
4. Os documentos incluídos e excluídos são registrados na lista de incluídos e de excluídos, respectivamente, presentes no Formulário de Condução da Revisão, juntamente com a justificativa de sua inclusão ou exclusão;
5. Os documentos incluídos são avaliados mediante a leitura do artigo inteiro; os documentos incluídos são registrados no Formulário de Seleção de Estudos e no Formulário de Extração de Dados. Os documentos excluídos são registrados na lista de documentos excluídos junto com a justificativa de exclusão no Formulário de Seleção de Estudos.

*Estratégia de Extração de Informações:*

Para cada estudo primário selecionado será utilizada uma cópia do Formulário de Extração de Dados.

*Sumarização dos Resultados:*

Os resultados serão tabulados e avaliados. Não há geração de meta-análise.

# 3 CONDUÇÃO DA REVISÃO

A Revisão Sistemática foi conduzida no período de 28 de Novembro até 24 de Dezembro de 2006. Ao todo foram analisados 29 documentos dos quais 10 foram selecionados e tiveram os seus dados extraídos e avaliados, de acordo com as recomendações previstas no protocolo de revisão.

Algumas fontes importantes, como as bases de dados do IEEE, não foram contempladas pelo Protocolo de Revisão Sistemática, devido ao fato da UNIFACS não possuir acesso às mesmas.

# 4 RESULTADOS OBTIDOS

## 4.1 DESCRIÇÃO DAS TÉCNICAS IDENTIFICADAS

Segue abaixo, a descrição sucinta das técnicas de *Data Mining* encontradas nos documentos incluídos na Revisão Sistemática.

***Decision Tree***:

Descrição – É uma técnica de mineração de dados voltada para a construção de modelos de classificação, baseados no particionamento recursivo de dados, iniciando com um conjunto de dados, seguindo com a divisão do mesmo em dois ou mais sub-conjuntos baseados em valores de um ou mais atributos, sendo que este processo é repetido até seja alcançado um nível apropriado (LI, 2004). A estrutura da árvore é organizada de tal forma que (CARVALHO, 2005): cada nó interno (não-folha) é rotulado com o nome de um dos atributos "previsores"; os ramos (ou arestas) saindo de um nó interno são rotulados com valores do atributo naquele nó; cada folha é rotulada com uma classe, a qual é a classe prevista para exemplos que pertençam àquele nó folha.

Em resumo, podemos inferir que o modelo gerado por uma *Decision Tree* como um conjunto de regras "*if-then*" (LI, 2004).

***Outlier:***

Descrição – É uma técnica que gera uma observação tão diferente de outras observações que geram suspeitas (HAWKINS *apud* LIN; BROWN, 2004). Tradicionalmente o estudo de *outlier* pode ser classificado em duas categorias principais (LIN; BROWN, 2004): *outlier accommodation* e *outlier identification*. Em *outlier accommodation*, a meta é o desenvolvimento de uma estimativa robusta que é insensível à existência de *outliers (observações distintas)*, isto é, para um conjunto de dados com pouca variação, os estudos se concentram na estimativa das medidas da tendência e dispersão centrais que não se situaram nos *outliers* (LIN; BROWN, 2004). Em *outlier identification*, ao contrário, os *outliers* são considerados pelos pesquisadores como sinais significativos e sobre como são concentradas as análises (LIN; BROWN, 2004).

***Case-Based Reasoning (CBR):***

Descrição – *case-based reasoning* (CBR) ou Raciocínio Baseado em Casos, é uma técnica de Inteligência Artificial, que busca aprender sobre padrões a partir de casos selecionados, visando classificar novos casos, além de poder se adaptar a novos padrões na medida que eles vão emergindo. Assim como outras técnicas baseadas em Inteligência Artificial, CBR é usado

contra problemas de natureza não-linear, ruidoso, contraditório e não endereçável (WEELER; AITKEN, 2000).

***Neural Networks:***

Descrição – *Neural Networks* ou Redes Neurais Artificiais (RNA) são ferramentas de Inteligência Artificial que possuem a capacidade de se adaptar e de aprender a realizar uma certa tarefa, ou comportamento, a partir de um conjunto de exemplos de dados, sendo aplicadas em domínios que possuem como características: robustez, generalização, paralelismo e tolerância ao ruído (OSÓRIO; BITTENCOURT, 2000).

***Support Vector Machine* (SVM)*:***

Descrição – é uma técnica voltada para processos de classificação e regressão, proposta por Vapnik e o seu grupo de pesquisa nos AT&T Bell Laboratories (VAPNIK *et al. apud* PANG *et al.*, 2003), que pode ser utilizada para processos de classificação binária ou de classificação de múltiplas classes, comumente empregada na identificação de padrões de comportamento de séries de dados (PANG *et al.*, 2003).

***Fraud Indicators – PROBIT model:***

Descrição – Indicadores são parâmetros para medir a diferença entre a situação desejada e a situação atual, ou seja, indicarão um problema (MILET *et al.*, 1993). *Fraud Indicators* é uma técnica que envolve a determinação de indicadores e a aplicação de um modelo probabilístico a fim de determinar o grau de probabilidade de fraudes a partir da análise dos indicadores.

***Artificial Immune System* (AIS):**

Descrição – é um tipo de algoritmo inspirado em princípios e processos do sistema imunológico vertebrado, explorando características de aprendizagem e de memória para resolver problemas. Ele é acoplado a inteligência artificial e com relacionamento próximo a algoritmo genético. Incluem a simulação de processos para reconhecimento de padrões, hipermutação e seleção clonal para B células, seleção negativa, maturação por afinidade e teoria de rede imunológica (WIKIPEDIA, 03/2007).

## 4.2 TABULAÇÃO

Tabela

| Tecnologia | Domínio | Documento | Observação |
|---|---|---|---|
| Decision Tree. | *E-commerce*; Intrusão em Redes de Computadores. | "Data Mining Prototype for detecting E-Commerce fraud [Research in progress]"; "A scalable decision tree system and its application in pattern recognition and intrusion detection". | Reconhecimento de padrões; detecção de intrusos em redes de computadores. |
| *Case-based Reasoning* (CBR). | Aprovação de crédito financeiro. | "Multiple algorithms for fraud detection". | Classificação e identificação de fraudes. |
| Neural Networks. | Processos de requisição médica; *Mobile Telecommunication.* | "A Medical Claim Fraud/Abuse Detection System based on Data Mining: A Case Study in Chile"; "Detecting Cellular Fraud using Adaptive Prototypes"; *"Data Mining, Fraud Detection and Mobile Telecommunications: Call Pattern Analysis with Unsupervised Neural Networks"*. | Geração de modelos de comportamento (padrões). |
| *Support Vector Machine* (SVM). | *Customer Relationship Management* (CRM). | "Fraud Detection using Support Vector Machine Ensemble". | Identificação, monitoração e predição de ocorrências de fraudes. |
| Indicadores – procedimentos probabilísticos (PROBIT *model*). | Seguros de automóveis. | "A Model for the Detection of Insurance Fraud". | Predição de ocorrências de fraudes. |
| Outlier. | Ativos financeiros – *stock market*. | *"Unsupervised Fraud Detection in Time Series data"*. | |
| *Artificial Immune System* (AIS) – baseado em *Association Rules.* | Fraudes financeiras no setor de Varejo. | *"Design of an Artificial Immune System as a Novel Anomaly Detector for Combating Financial Fraud in the Retail Sector"*. | |

Nota: elaboração própria

# REFERÊNCIAS

ABIDOGUN, O. A. **Data mining, fraud detection and mobile telecommunications:** Call Pattern Analysis with Unsupervised Neural Networks. 2005. Disponível em: <www.uwc.ac.za/library/theses/>. Acesso em: 20.07.2006.

BELHADJI, E. L. B.; DIONNE, G.; TARKHANI, F. A model for the detection of insurance fraud. **The Geneva Papers on Risk and Insurance.** West Sussex, Inglaterra: Blackwell Publishers, v. 25, n. 4, p. 448-471, 13 out. 2000.

BRAUSE, R.; LANGSDORF, T.; HEPP, M. **Neural data mining for credit card fraud detection**. 1999. Disponível em: <www.citeseer.ist.psu.edu/>. Acesso em: 11 ago. 2006.

BURGE, P.; SHAWE-TAYLOR, J. **Detecting cellular fraud using adaptive prototypes**. 1997. Disponível em: <https://www.aaai.org/Papers/Workshops/1997/WS-97-07/WS97-07-002.pdf>. Acesso em: 11 ago. 2006.

CAHILL, M. H. *et al.* **Detecting fraud in the realword**. 2000. Disponível em: <ftp://cm.bell-labs.com/cm/ms/departments/sia/doc/hmds.pdf>. Acesso em: 13 ago. 2006.

CHIU, C.; TSAI, C. A. Web services-based colaborative scheme for credit card fraud detection. In: IEEE INTERNATIONAL CONFERENCE 2004, 2004, Nova Orleans, EUA. **Proceedings…** v. 1, p. 177 – 181. Disponível em: <www2.computer.org/portal/web/csdl/doi/10.1109/EEE.2004.1287306>. Acesso em: 20 ago. 2006.

DESOUZA, K. C. Artificial Intelligence for Healthcare Management. In: INTERNATIONAL CONFERENCE ON MANAGEMENT OF HEALTHCARE AND MEDICAL TECHNOLOGY, 1., 2001, Enschede, Holanda. **Proceedings …** 2001. v. 4, p. 401-420. 2001. Disponível em: <www.stuart.iit.edu>. Acesso em: 22 set. 2006.

FERDOUSI, Z.; MAEDA, A. **Unsupervised fraud detection in time series data.** 2006. Disponível em: <www.db.soc.i.kyoto-u.ac.jp/DEWS2006/>. Acesso em: 23 set. 2006.

HE, Z.; XU, X.; DENG, S. **Discovering cluster-based local outliers.** Elsevier Science B.V. 2002. Disponível em: <zengyouhe.googlepages.com/PRL03.pdf>. Acesso em: 23.set. 2006.

KIM, J.; ONG, A.; OVERILL, R. E. **Design of an Artificial immune system as a novel anomaly detector for combating financial fraud in the retail sector**. 2003. Disponível em: <psu.edu/10.1.1.5.8960.pdf>. Acesso em: 23 set.2006.

LEK, M. *et al.* Data Mining Prototype for Detecting E-Commerce Fraud. In: THE EUROPEAN CONFERENCE ON INFORMATION SYSTEMS, 9., 2001, Bled, Eslovênia. **Proceedings …** 2001. v.1, p. 27-29. Disponível em: <is2.lse.ac.uk/asp/aspecis/20010066.pdf>. Acesso em: 23 set. 2006.

LI, X. **A scalable decision tree system and its application in pattern recognition and intrusion detection**. Elsevier Science B.V. 2004. Disponível em: <www.sciencedirect.com/doi:10.1016/j.dss.2004.06.016/>. Acesso em: 23 set. 2006.

MAFRA, S. N.; TRAVASSOS, G. H. Técnicas de leitura de software: uma revisão sistemática. In: SBES, 19., 2005. Uberlândia. **Proceedings ...** 2005. Disponível em: <www.sbbd-sbes2005.ufu.br>. Acesso em 21 ago. 2006.

MILET, E. *et al.* **Indicadores de qualidade e produtividade para a área de informática**. Rio de Janeiro: LTC/MCG, 1993.

OSÓRIO, F. S.; BITTENCOURT, J. R. Sistemas Inteligentes baseados em Redes Neurais Artificiais aplicados ao Processamento de Imagem. In: I WORKSHOP DE INTELIGÊNCIA ARTIFICIAL – UNIVERSIDADE DE SANTA CRUZ DO SUL, 2000, Santa Cruz do Sul, . **Proceedings...** 2000 p. 14-26. Disponível em: <inf.unisinos.br/~osorio/wia-unisc/wia2000-full.pdf>. Disponível em: 21 ago. 2006.

ORTEGA, P. A.; FIGUEROA, C. J.; RUZ, G. A. **A medical claim fraud/abuse detection system based on data mining:** a case study in Chile. 2006. Disponível em: **<**www.mec.cf.ac.uk/~scegr2/pub/DMI5560.pdf>. Acesso em: 20 set. 2006.

PANG, S. N.; KIM, D.; BANG, S. Y. **Fraud detection using support vector machine ensemble**. elsevier science b.v. 2003. disponível: <appsrv.cse.cuhk.edu.hk/~apnna/proceedings/iconip2001/papers/138a.pdf>. Acesso em: 20 set. 206.

SONG, L.; BROWN, D. E.. **An outlier-based data association method for linking criminal incidents.** Elsevier Science B.V. 2004. Disponível em: <protege.stanford.edu>. Acesso em: 13 out. 2006.

TODESCO, J. L.. **Reconhecimento de padrões usando rede neuronal artificial com uma função de base radial:** uma aplicação na classificação de cromossomos humanos. 1995. 2 v. Tese (Doutorado) - Curso de Engenharia de Produção, Univers, Florianópolis, Santa Catarina, 1995. Disponível em: <www.eps.ufsc.br/teses/todesco/indice/index.html#index.>. Acesso em: 20 ago. 2006.

VIAENE, S.; DERRIG, R. A.; DEDENE, G. **A case study of applying boosting naive bayes to claim fraud diagnosis**. 2004. Disponível em: <gkmc.utah.edu/7910F/papers/IEEE%20TKDE%20claim%20fraud%20diagnosis.pdf>. Acesso em: 21 ago. 2006.

WEELER, R.; AITKEN, S. **Multiple algorithms for fraud detection**. Elsevier Science B.V. 2000. Disponível em: <pdu.edu/10.1.1.14.7153.pdf>. Acesso em: 21 ago. 2006.

ZHANG, S.; ZHANG, C.; YU, J. X. **An efficient strategy for mining exceptions in multi-databases**. Elsevier Science B.V. 2004. Disponível em: <www-staff.it.uts.edu.au/~zhangsc/scpaper/inszzyu.pdf>. Acesso em: 20.07.2006.

# ANEXO B - Exemplo para detalhamento de pacotes presentes ao modelo TOF´s Fraud Architecture

Exemplo para detalhamento de pacotes presentes ao modelo TOF´s *Fraud Architecture* (ZHAO; LEARY, 2005):

- Exemplo 1: detalhamento do pacote *Fraud Type*.

a) Para *Fraud Type* se têm a hierarquia abaixo:

a.1) *By Perpetrator*

1. *Employee* / *occupational fraud*

2. *Management fraud*

3. *Investment fraud*

4. *Vendor fraud*

5. *Customer fraud*

a.2) *By Victim*

1. *Bank fraud*

a.3) *By Stolen Object*

1. *Securities fraud*

2. *Tax evasion*

a.4) *By Instrument*

1. *Computer fraud*

a.5) *By Scheme*

1. *Kickback*

2. *Bribery*

- Exemplo 2: detalhamento do pacote *Trust*.

b) Para *Trust*, subclasse de *Attribute* presente no pacote *Fraud Configuration*, que remete a origem da fraude, têm-se:

b.1) *Sympathy*

b.2) *Social roles* (ocupação estabelecida, etc)

b.3) *Justification* (direito de causa)

b.4) *Association* (associado com personagem estabelecida ou representante de instituições)

b.5) *Affinity* (experiência similar com vítimas)

b.6) *Work ethics*

b.7) *Past record*

- Exemplo 3: detalhamento do pacote *Pressure*.

c) Para *Pressure*, subclasse de *Perpetration* presente no pacote *Motivation*, que remete às pressões que motivaram a fraude, têm-se:

c.1) *Financial*

1. *Greed*

2. *Expenses*

3. *Debts*

4. *Needs* (não esperados)

5. *Losses*

c.2) *Vice*

1. *Gambling*

2. *Drugs*

3. *Alcohol*

c.3) *Social*

1. *Expensive extramarital relationship*

2. *Peer pressure*

c.4) *Work Related*

1. *Recognition*

2. *Promotion*

3. *Expectation*

c.5) *Threat*

c.6) *Obligation*

1. *Responsibility*

2. *Contractual*

# ANEXO C - Literaturas foram utilizadas para identificar os termos usados na construção das ontologias desenvolvidas

Relação de livros, artigos científicos, dissertações e teses relacionadas aos temas: fraude, detecção de fraudes e, técnicas para detecção e para detecção automática de fraudes. Estas literaturas foram utilizadas para identificar os termos usados na construção das ontologias, e para definir os conceitos-chave, as relações-chave e os indivíduos referentes aos termos:

ABBOTT, D. W.; MATKOVSKY, I. P.; ELDER IV, J F. An Evaluation of High-end Data Mining Tools for Fraud Detection. In: IEEE INTERNATIONAL CONFERENCE ON SYSTEM, MAN, AND CYBERNETICS, 4., 1998, Tóquio, Japão. **Proceedings…** v. 3, p. 2836 - 2841. Disponível em: <http://www.citeulike.org/user/robincha/article/940990>. Acesso em: 23 jun. 2007.

ABIDOGUN, O. A. **Data mining, fraud detection and mobile telecommunications: call pattern analysis with unsupervised neural networks.** 2005. Disponível em: <www.uwc.ac.za/library/theses/>. Acesso em: 20 ago. 2006.

ALLEN, G. B. **The fraud identification handbook.** Highlands Ranch, EUA: Pp Preventive Press, 1999.

APTÉ, C.; WEISS, S. Data mining with decision trees and decision rules. **Future Generation Computer Systems**, Pittsburgh, EUA, v. 3, n. 13, p. 197-210, 10 dez. 1997. Disponível em: <www.elsevier.com/locate/future>. Acesso em: 23 ago. 2006.

BARBIERI, C. **BI – Business Intelligence:** modelagem & tecnologia. Rio de Janeiro, Brasil: Axcel Books do Brasil Editora, 2001.

BOLTON, R. J.; HAND, D. J. Statistical Fraud Detection: A Review. **Statistical Science**, Filadélfia, EUA, v. 17, n. 3, p. 235-255, 10 ago. 2002.

BOSE, I.; MAHAPATRA, R.. Business data mining: a machine learning perspective. **Information & Management**, Nova York, EUA, v. 39, n. , p.211-225, 20 dez. 2001. Disponível em: <www.elsevier.com/locate/dsw>. Acesso em: 23 ago. 2006.

BRAUSE, R.; LANGSDORF, T.; HEPP, M. **Neural data mining for credit card fraud detection**. 1999. Disponível em: <www.citeseer.ist.psu.edu/>. Acesso em: 11 ago. 2006.

BURGE, P.; SHAWE-TAYLOR, J. **Detecting cellular fraud using adaptive prototypes**. The ACM Library Digital. 1997. Disponível em: <https://www.aaai.org/Papers/Workshops/1997/WS-97-07/WS97-07-002.pdf>. Acesso em: 11 ago. 2006.

CAHILL, M. H. *et al*. **Detecting fraud in the realword**. 2000. Disponível em: <ftp://cm.bell-labs.com/cm/ms/departments/sia/doc/hmds.pdf>. Acesso em: 13 ago. 2006.

CRAVEN, M. W.; SHAVLIK, J. W. Using neural networks for data mining. **Future generation computer systems**, Pittsburgh, EUA, v. 3, p. 211-229, 12 nov. 1997. Disponível em:www.elsevier.com/locate/future. Acesso em: 23 ago. 2006.

CHIU, C.; TSAI, C. A. Web services-based colaborative scheme for credit card fraud detection. In: IEEE INTERNATIONAL CONFERENCE 2004, 2004, Nova Orleans, EUA. **Proceedings…** v. 1, p. 177-181. Disponível em: <www2.computer.org/portal/web/csdl/doi/10.1109/EEE.2004.1287306>. Acesso em: 20 ago. 2006.

DHAR, V.; STEIN, R. **Seven methods for transforming corporate data into business intelligence**. Nova Jersey, EUA: Prentice Hall, 1997.

ECONOMIST INTELLIGENCE UNIT AND KROLL (Org.). **Global fraud report: Annual** Edition 2007/2008**.** Londres, Inglaterra, 2007.

EDMONDS, J. *et al.* Mining for empty spaces in large data sets. **Theoretical Computer Science**, Essex, Inglaterra, v. 296, p. 435-452, 15 mar. 2003. Disponível em: <www.elsevier.com/locate/tcs>. Acesso em: 13 set. 2006.

EIDE, A. *et al.* Data mining and neural networks for knowledge discovery. **Nuclear Instruments And Methods In Physics Research**, Paris, França, seção A, p.251-254, 1997. Disponível em: <cat.inist.fr>. Acesso em: 13 set. 2006.

FAWCETT, T.; PROVOST, F. **Adaptive fraud detection**: data mining and knowledge discovery. Boston: Kluwer Academic Publishers, 1997.

FEDERAL BUREAU OF IVESTIGATION – FBI. **Insurance fraud**. 2006. Disponível em: <denver.fbi.gov/documents/insurance_trifold_final.pdf>. Acesso em: 12 jun. 2007.

FENG, Y. L.; MCCLEAN, S.. A data mining approach to the prediction of corporate failure. **Knowledge-based Systems**, Belfast, Irlanda do Norte, v. 14, n. 3-4, p.189-195, 2001. Disponível em: <www.sciencedirect.com/doi:10.1016/S0950-7051>. Acesso em: 13 set. 2006.

GOLDSCHMIDT, R.; PASSOS, E. **Data mining**: um guia prático. Rio de Janeiro, Brasil: Elsevier Editora, 2005.

GROTH, R. **Data mining:** building competitive advantage. New Jersey: Prentice Hall PTR, 2000.

JEN, N. C.; JASON, S. C.. Topical clustering of MRD senses based on information retrieval techniques. **Computational Linguistics**, Cambridge, EUA, v. 24, p.61-95, 01 mar. 1998.

KIMBALL, R. **The data warehouse toolkit**. Nova York: John Wiley & Sons INC., 1996.

LI, Y. *et al.* Data mining ontology development for high user usability. **Wuhan University Journal Of Natural Sciences,** Hubei, China, p. 10-14. 11 jan. 2006. Disponível em: <d.wanfangdata.com.cn/Periodical_whdxxb-e200601011.aspx>. Acesso em: 20 ago. 2007.

LUNDIN, E. **Aspects of employing fraud and intrusion detection systems.** 2002. Disponível em: <www.ce.chalmers.se/~emilie/papers/lundin_lic.pdf>. Acesso em: 10 ago. 2006.

PARODI, L. **Manual das fraudes**. Rio de Janeiro: Brasport Livros e Multimídia Ltda., 2005.

PHUA, C. W. C. **Investigative data mining in fraud detection**. 2003. Disponível em: <bsys.monash.edu.au>. Acesso em: 20 dez. 2006.

PHUA, C. *et al.* A comprehensive survey of data mining - based fraud detection research. **Artificial Intelligence Review**, Amsterdan, Holanda, n. , p.34-39, 15 fev. 2005. Disponível em: <http://clifton.phua.googlepages.com/fraud-detection-survey.pdf>. Acesso em: 23 ago. 2006.

SOUSA, M. S. R. **Mineração de dados:** uma implementação fortemente acoplada a um sistema gerenciador de banco de dados paralelo. 2007. 75 f. Dissertação (Mestrado) - Departamento de Coppe, Ufrj, Rio de Janeiro, Brasil, 2007. Disponível em: <http://www.cos.ufrj.br/~marta/papers/TeseMauroS.pdf>. Acesso em: 13 ago. 2006.

RUBIN, S. H. A Fuzzy approach towards inferential data mining. **Computers & Industrial Engineering**, Nova York, Eua, v. 35, p. 267-270, 1998. Elsevier Science Ltd. Disponível em: <www.sciencedirect.com/doi:10.1016/S0360-8352>. Acesso em: 20 set. 2006.

SILVERSTONE, H.; DAVIA, H. R. **Fraud 101:** techniques and strategies for detection. New Jersey, EUA: John Wiley & Sons, 2005.

SONG, L.; BROWN, D. E. **An outlier-based data association method for linking criminal incidents.** 2004. Disponível em: <psu.edu/doi:10.1016/j.dss.2004.06.005>. Acesso em: 13 out. 2006.

TANIGUCHI, M. *et al*. Fraud Detection in Communications Networks using Neural and Probabilistic Methods. In: IEEE INTERNATIONAL CONFERENCE ON ACOUSTICS, SPEECH AND SIGNAL PROCESSING, 1998, Seattle, EUA. **Proceedings …** Nova Jersey, EUA: Ieee Press, 1998. v. 6, p. 2043-2046. Disponível em: <http://www.cis.hut.fi/jhollmen/Publications/icassp98.pdf>. Acesso em: 11 ago. 2006.

TZUNG-PEI, H.; CHAN-SHENG, K.; SHENG-CHAI, C. Mining association rules from quantitative data. **Intelligent Data Analysis**, Nova York, Eua, v. 14, n. , p.363-376, 1999. Elsevier Science Ltd. Disponível em: <www.sciencedirect.com/doi: 10.1016/S1088-467X>. Acesso em: 20 set. 2006.

YUFENG, K. *et al*. Survey of Fraud Detection Techniques. In: NETWORKING, SENSING AND CONTROL, IEEE INTERNATIONAL CONFERENCE, 1., 2004, Taipei, Taiwan. **Proceedings ….** Nova York, EUA: IEEE Press, 2004. v. 2, p. 749 - 754. Disponível em: <http://ieeexplore.ieee.org/servlet/opac?punumber=9086>. Acesso em: 23 mar. 2004.

WILHELM, W. K. The fraud management lifecycle theory: a holistic approach to fraud management. **Journal Of Economic Crime Management Spring,** Nova York, EUA, p. 2-9. 10 maio 2004. Disponível em: <http://www.utica.edu/academic/institutes/ecii/publications/articles/BA309CD2-01B6-DA6B-5F1DD7850BF6EE22.pdf>. Acesso em: 23 ago. 2006.

ZADEH, L. A. From Computing With Numbers To Computing Withwords. **International Journal of applied mathematics and computer science (amcs)**, v.12, n.3, p. 307–324. 05

fev. 2002. Disponível em: <www-bisc.cs.berkeley.edu/ZadehCW2002.pdf>. Acesso em: 20 jul.2009.

ZHANG, S.; ZHANG, C.; YU, J. X. **An efficient strategy for mining exceptions in multi-databases**. 2004. Disponível em: <www-staff.it.uts.edu.au/~zhangsc/scpaper/inszzyu.pdf>. Acesso em: 20.07.2006.

ZHAO, G.; MEERSMAN, R. **Towards a topical ontology of fraud**. Disponível em: <www.springerlink.com/content/7564777715q6gw68/>. 2006. Acesso em: 23 mar.2007.