Universidade Federal de Santa Catarina
Programa de Pós-Graduação em
Engenharia de Produção

UM MÉTODO DE TRADUÇÃO DE FONTES DE INFORMAÇÃO EM UM FORMATO PADRÃO QUE VIABILIZE A EXTRAÇÃO DE CONHECIMENTO POR MEIO DE LINK ANALYSIS E TEORIA DOS GRAFOS

Alessandro Botelho Bovo

Dissertação apresentada ao
Programa de Pós-Graduação em
Engenharia de Produção da
Universidade Federal de Santa Catarina
como requisito parcial para obtenção
do título de Mestre em
Engenharia de Produção

Florianópolis
2004

Alessandro Botelho Bovo

UM MÉTODO DE TRADUÇÃO DE FONTES DE INFORMAÇÃO EM UM FORMATO PADRÃO QUE VIABILIZE A EXTRAÇÃO DE CONHECIMENTO POR MEIO DE LINK ANALYSIS E TEORIA DOS GRAFOS

Esta dissertação foi julgada e aprovada para a obtenção do grau de **Mestre em Engenharia de Produção** no **Programa de Pós-Graduação em Engenharia de Produção** da Universidade Federal de Santa Catarina.

Florianópolis, 3 de julho de 2004.

_____
Prof. Edson Pacheco Paladini, Dr.
Coordenador do Curso

BANCA EXAMINADORA

_____
Prof. Roberto C. dos S. Pacheco, Dr.
Orientador
Universidade Federal de Santa Catarina

_____
Prof. Vinícius Medina Kern, Dr.
Universidade Federal de Santa Catarina

_____
Prof. Aran Bey Tcholakian Morales, Dr.
Universidade Federal de Santa Catarina

Agradecimentos

A Deus, por me acompanhar e iluminar o meu caminho.

Ao meu orientador, professor Roberto Carlos dos Santos Pacheco, e aos professores Aran Bey Tcholakian Morales e Vinícius Medina Kern, que muito contribuíram para o desenvolvimento desta dissertação.

À professora Maria Madalena Dias, da Universidade Estadual de Maringá (UEM), e ao André Vinícius Castoldi, que foram fundamentais para a minha vinda para Florianópolis.

Aos colegas e amigos do Grupo Stela, com os quais aprendi muito durante esses anos de mestrado.

Aos amigos do Grupo de Oração Universitário (GOU), que são a minha segunda família.

Aos meus pais, Getúlio e Leidí, e aos meus irmãos, Fábio e Eduardo, que, mesmo estando longe, foram fundamentais para que eu conseguisse chegar até aqui.

À minha esposa, Alessandra, por seu amor, amizade, compreensão e por estar sempre ao meu lado.

Sumário

Lista de figuras

Lista de quadros

Lista de siglas

BD	Banco de Dados
CNPq	Conselho Nacional de Desenvolvimento Científico e Tecnológico
CONSCIENTIAS	Comunidade para Ontologias em Ciência, Tecnologia e Informações de Aperfeiçoamento de Nível Superior
C&T	Ciência e Tecnologia
CT&I	Ciência, Tecnologia e Inovação
FAIS	FinCEN Artificial System
HCI	Human Computer Interaction
HTML	Hypertext Markup Language
IA	Inteligência Artificial
IES	Instituição de Ensino Superior
KDD	Knowledge Discovery in Databases
LA	Link Analysis
LMPL	Linguagem de Marcação da Plataforma Lattes
MD	Mineração de Dados
OLAP	On-Line Analytical Processing
PL	Plataforma Lattes
RDF	Resource Description Framework
SQL	Structured Query Language
SVG	Scalable Vector Graphics
TG	Teoria dos Grafos
XML	Extensible Markup Language
XSLT	Extensible Stylesheet Language Transformation

Resumo

BOVO, B. Alessandro. **Um método de tradução de fontes de informação em um formato padrão que viabilize a extração de conhecimento por meio de *Link Analysis* e Teoria dos Grafos**. 2004. 102 f. Dissertação (Mestrado em Engenharia de Produção) – Programa de Pós-Graduação em Engenharia de Produção, Universidade Federal de Santa Catarina, Florianópolis, 2004.

O conhecimento tem se configurado como um recurso estratégico nas organizações. Para elas, gerar, codificar, gerir e disseminar o conhecimento organizacional tornaram-se tarefas essenciais. Logo, é necessário o desenvolvimento de novas técnicas, metodologias e formas de extração de conhecimento a partir de fontes de informação que descrevem um domínio de aplicação. Nesse contexto, o objetivo do presente trabalho é propor um método que permita traduzir fontes de informação em um formato padrão de representação de relacionamentos entre elementos do domínio do problema, de forma a viabilizar a extração de conhecimento por meio da aplicação de *Link Analysis* e Teoria dos Grafos. Além disso, são apresentadas duas aplicações desse modelo na Plataforma Lattes de CT&I.

Palavras-chave: Extração do Conhecimento; Link Analysis; Teoria dos Grafos; Plataforma Lattes.

Abstract

Knowledge has been become a strategical resource in organizations. Thus, to generate, codify, manage and disseminate the organizational knowledge are essential tasks in organizations. Soon, it is necessary to develop new knowledge extraction techniques, methodologies and ways from information sources that describe an application domain. Inside of this context, the objective of this work is to present a method that allows to translate information sources in a standard format that represents the relationships between domain elements in order to allow knowledge extraction by using Link Analysis and Graph Teory methods. Moreover, we build two applications using this method in a Science, Technology and Innovation management platform.

Keywords: Knowledge Extraction; Link Analysis; Graph Theory; Science, Technology and Innovation management platform.

1 INTRODUÇÃO

Cada vez mais o conhecimento assume importância como um recurso estratégico nas organizações. Assim, gerar, codificar, gerir e disseminar o conhecimento organizacional tornaram-se tarefas essenciais às organizações pertencentes à sociedade do conhecimento. Para Oliveira (1992), essa é a era da economia do saber: "ganha a guerra quem sabe mais, quem sabe aprender e quem aprende mais depressa". Isso se aplica aos indivíduos, às organizações e aos países. Já Hesselbein et al. (1997) afirmam que essas transformações são tão profundas que é possível dizer que está havendo uma terceira revolução industrial, a qual pode ser chamada na verdade de revolução da informação. Assim, as organizações vencedoras serão aquelas que conseguirem acumular saber.

Nesse contexto, percebe-se a necessidade das organizações em utilizar sistemas de conhecimento (*i.e.,* sistemas especialistas, sistemas baseados em conhecimento e sistemas de informação de conhecimento intensivo (UNIVERSIDADE FEDERAL DE SANTA CATARINA, 2004[1]), cujos benefícios às organizações incluem aumento de produtividade, preservação de conhecimento, melhoria na qualidade de tomada de decisões, subsídios à capacitação organizacional e à valorização do trabalho (MARTIN; SUBRAMANIAN; YAVERBAUM, 1996).

Diante desse quadro de preocupação com o conhecimento nas organizações, compreende-se a importância das diferentes técnicas, metodologias e formas de extração de conhecimento a partir de fontes de informação que descrevem um domínio de aplicação. A maioria delas está voltada à apresentação de conhecimento textual ou representação de categorias e regras de comportamento em bases de dados, tais como classificação, agrupamento, regressão, sumarização, etc. (BERRY; LINOFF, 1997).

Contudo, uma das principais características com que o ser humano identifica padrões e conhecimentos está no comportamento visual e no acesso a imagens. Nesse sentido, um grupo particular de técnicas de extração de conhecimento traz vantagens porque incorpora a abordagem da exploração visual para descobrir ou

[1] Projeto enviado à Capes para aprovação do curso de Pós-Graduação em Engenharia e Gestão do Conhecimento da Universidade Federal de Santa Catarina.

identificar o conhecimento, permitindo maior abrangência, visão do todo ou compreensão da holística de um problema. Trata-se da área de *Link Analysis*, que envolve técnicas que tem por finalidade revelar a estrutura e o conteúdo de um conjunto de informações por meio de unidades (entidades ou objetos) interconectadas (GOLDBERG; SENATOR, 1998).

Quando a informação é representada por relacionamentos, além de permitir compreender melhor domínios por análise visual, há uma nova gama de estudos e conclusões sobre o domínio do problema à disposição do especialista. Essa é a base das teorias e dos métodos aplicados em áreas como redes sociais, por exemplo, em que se procura analisar a natureza das ligações entre os elementos do problema. A característica fundamental da análise de redes sociais é lidar com dados relacionais (WASSERMAN; FAUST, 1994; HANNEMAN, 2000), ou seja, dados que expressam relações (conexões ou laços) entre objetos diversos (nós, indivíduos, grupos). Assim como em redes sociais, o conhecimento implícito em relacionamentos também constitui o objeto-análise de diversas áreas de aplicação da Teoria dos Grafos, como problemas de distribuição, análise de fluxos, logística de transportes, alocação e outros. Outras áreas em que o estudo de relacionamentos revela conhecimentos implícitos incluem problemas-alvo referentes à teoria de *Link Analysis*, tais como categorização de sub-redes de telefonia e classificação de clientes por redes de relacionamentos.

Segundo Jensen (1998), muitos conjuntos de dados podem ser representados naturalmente como coleções de objetos conectados. O papel da Tecnologia da Informação nas organizações evoluiu do suporte às operações ao posicionamento estratégico em nível de decisão e posicionamento junto aos mercados. Em todas as suas fases de evolução, características comuns foram a abstração, a modelagem e o registro das informações organizacionais em bases de dados ou repositórios de textos. No primeiro caso, uma propriedade recorrente é a existência abundante de relacionamentos, os quais são resultado de concepção teórica de modelos Entidade–Relacionamento, que fundamentam a área de Bancos de Dados Relacionais. Portanto, é significativo o volume disponível de links e de relacionamentos nos repositórios de informação das organizações. A combinação destes com os estudos de redes sociais potencializa a geração de novos conhecimentos nas organizações, particularmente com a aplicação de *Link Analysis,*

a qual se centra na busca, identificação, criação, análise e apresentação visual dos relacionamentos de um domínio de informação. A Teoria dos Grafos, por sua vez, permite que relações representadas por unidades (vértices) e ligações entre eles (arestas) sejam passíveis de uma gama de métodos de análise, que vão desde o cálculo de distâncias entre vértices até a obtenção de subgrafos (sub-redes).

A combinação das duas teorias e a disponibilidade de fontes de informação que descrevem um domínio de aplicação trazem novas possibilidades para atender à demanda por geração e análise de conhecimentos nas organizações. A partir da combinação de ambas, pode-se traduzir bases de dados em arquivos XML em formato padrão de descrição de relacionamentos. As técnicas de *Link Analysis* podem ser úteis para identificar os relacionamentos entre elementos do domínio quando estes não são previamente conhecidos na base de dados. Depois disso, os arquivos descritores de relacionamentos são transformados em grafos e, a partir daí, aplicam-se métodos da Teoria dos Grafos ou *Link Analysis* para extrair novos conhecimentos sobre o domínio descrito pelas fontes originais de informação.

A presente dissertação tem como motivação combinar as áreas de *Link Analysis* e Teoria dos Grafos na exploração de fontes de informação. Para tal, considera-se o mapeamento dessas fontes de informação e a identificação de relacionamentos entre seus elementos. O resultado é a possibilidade de que novos conhecimentos sobre o domínio no qual foram desenvolvidos sistemas de informação sejam produzidos a partir das novas análises que os usuários realizam com base nos elementos visuais gerados.

1.1 Questões de pesquisa

Considerando-se os pressupostos mencionados, esta dissertação apresenta as questões de pesquisa descritas a seguir.

A combinação de técnicas de *Link Analysis* com métodos da Teoria dos Grafos pode ser útil à área de extração do conhecimento em fontes de informação mapeadas em arquivos XML?

Em caso afirmativo, como combiná-las? Quais são as características do domínio do problema em que se pode aplicar essa combinação de técnicas?

1.2 Objetivo geral

Propor um método que permita traduzir fontes de informação (banco de dados relacionais, *data warehouses* e documentos XML) em um formato padrão de representação de relacionamentos entre elementos do domínio do problema, de forma a viabilizar a extração de conhecimento por meio da aplicação de *Link Analysis* e Teoria dos Grafos.

1.2.1 Objetivos específicos

Os objetivos específicos deste trabalho são:

- estudar a área de *Knowledge Discovery in Databases* (KDD) com ênfase em *Link Analysis*, visando identificar os métodos de extração de conhecimento baseados no estudo de relacionamentos entre elementos de um domínio;
- estudar a área de Teoria dos Grafos e suas aplicações afins ao objetivo geral (e.g., redes sociais de colaboração), visando identificar métodos aplicáveis à extração de conhecimento a partir de informações representadas na forma de relacionamentos entre os elementos de um domínio;
- estabelecer um padrão de representação de relacionamentos entre elementos do domínio de um problema, de forma a permitir a aplicação de técnicas de análise de redes e extração de novos conhecimentos sobre o referido domínio; e
- com base nas metas anteriores, definir um método de tradução de fontes de informação para o padrão de representação de relacionamentos proposto, estender o padrão de modo a permitir representação gráfica da rede gerada e, finalmente, aplicar técnicas de Teoria dos Grafos ou *Link Analysis* para geração de novos conhecimentos sobre o domínio do problema.

- Para alcançar esses objetivos propõe-se um método que parta da transcrição de fontes de informação em arquivos XML em que os elementos e atributos descrevam unidades e relacionamentos do domínio das fontes. A partir desses arquivos, aplicam-se componentes de software que transformam os relacionamentos em redes gráficas (grafos), viabilizando a aplicação de diferentes procedimentos da Teoria dos Grafos e *Link Analysis*, o que permite aos especialistas realizarem novas descobertas sobre o domínio do problema tratado pelos sistemas de informação aos quais as fontes estão associadas.

1.3 Justificativa

Com o surgimento da sociedade do conhecimento questionam-se os sistemas de informação disponíveis quanto à sua suficiência para atender às novas exigências individuais e coletivas no que se refere à aquisição, geração, gestão e disseminação desse conhecimento. Assim, para que uma organização se coloque em vantagem competitiva em relação às demais são necessários novos desenvolvimentos nas áreas de engenharia e gestão do conhecimento.

Nesse contexto, acredita-se que este trabalho trará benefícios à área de KDD possibilitando a construção de sistemas de conhecimento que venham ao encontro das necessidades das organizações da sociedade do conhecimento. Esses sistemas permitirão a descoberta de novos conhecimentos analisando os relacionamentos entre os elementos do domínio do problema.

1.4 Metodologia

Para alcançar os objetivos desejados, este trabalho compõe-se de quatro etapas principais: (1) estudo sobre *Link Analysis*; (2) estudo sobre Teoria dos Grafos; (3) desenvolvimento do método de tradução de fontes de informação para o padrão de representação de relacionamentos; e (4) aplicação do método proposto na Plataforma Lattes de CT&I.

A primeira etapa está dividida em duas partes:

- estudar a área de extração do conhecimento; e
- estudar os conceitos e aplicaçõesreferentes à técnica de *Link Analysis*.

A segunda etapa compreende dois pontos principais:

- estudar os principais conceitos da Teoria dos Grafos; e
- estudar as áreas de aplicação dos algoritmos da Teoria dos Grafos.

A terceira etapa é composta de duas partes, a saber:

- estabelecer um formato padrão, em XML, de representação de relacionamentos entre elementos do domínio de um problema; e
- definir um modelo de tradução de fontes de informação para o padrão de representação de relacionamentos definido;

A quarta etapa consiste em aplicar o método proposto na construção de ferramentas no âmbito da Plataforma Lattes de CT&I.

A Figura 1.1 apresenta, de forma esquemática, a metodologia adotada no trabalho.

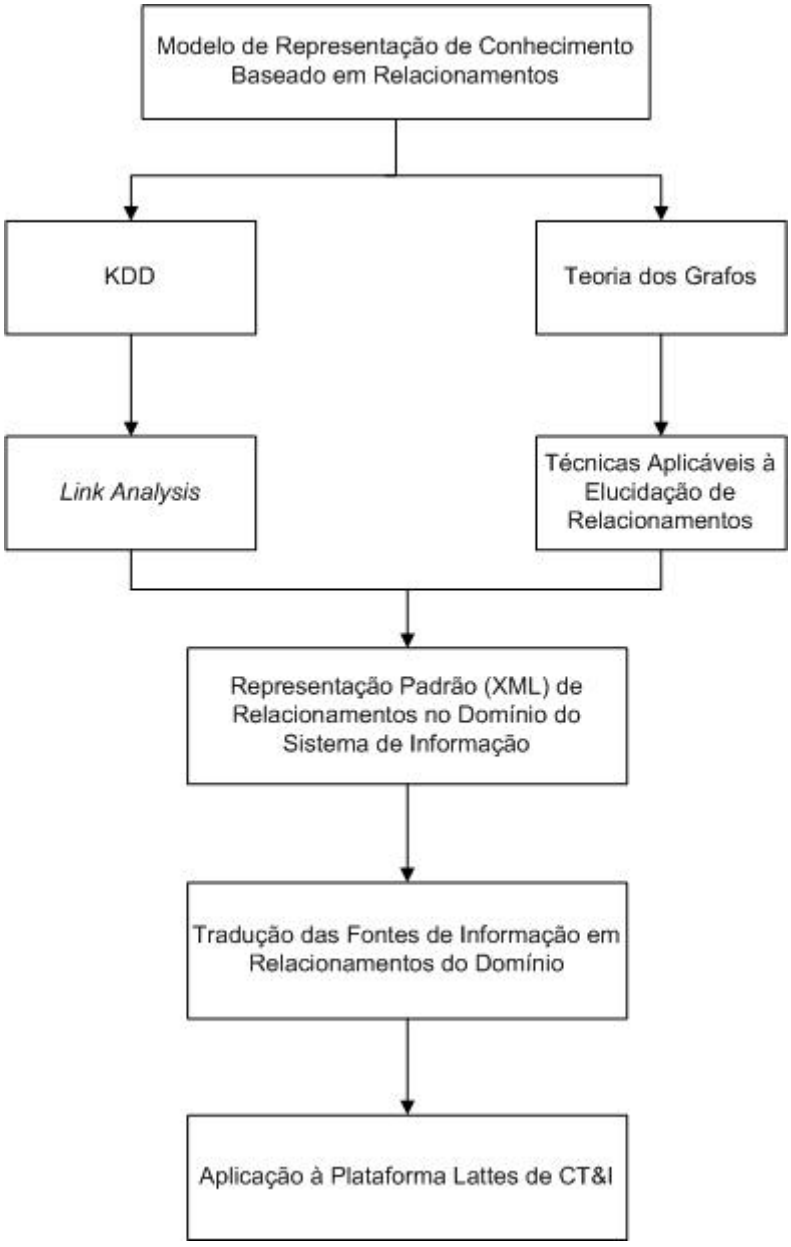

Figura 1.1 - Representação esquemática da metodologia adotada no trabalho

1.5 Estrutura do trabalho

Esta dissertação contém seis capítulos. Neles, apresentam-se a fundamentação teórica resultante da pesquisa bibliográfica, o método proposto, a aplicação do método em uma plataforma de CT&I e as conclusões e recomendações de trabalhos futuros. Abaixo segue a estrutura adotada para esta dissertação.

- Capítulo 2: *Link Analysis* – são apresentadas as definições, técnicas e aplicações da área de extração do conhecimento, com destaque para *Link Analysis*.
- *Capítulo 3: Teoria dos Grafos* – no tocante à essa teoria explicitam-se as definições correspondentes, os conceitos básicos e as áreas de aplicação, tais como redes sociais.
- *Capítulo 4: O método proposto* – este capítulo descreve o método de tradução de fontes de informação para o padrão de representação de relacionamentos entre os elementos do domínio.
- *Capítulo 5: Aplicação do método proposto* – este capítulo apresenta duas aplicações do método proposto à Plataforma Lattes de CT&I.
- *Capítulo 6: Conclusões e trabalhos futuros* – são descritas as conclusões obtidas na pesquisa e sugerem-se trabalhos futuros a ela relacionados.

2 LINK ANALYSIS

2.1 Introdução

Neste capítulo serão apresentadas as definições de *Link Analysis* (LA), Mineração de Dados (MD) e Processo de Descoberta de Conhecimento em Banco de Dados (KDD – *Knowledge Discovery in Databases*). Serão relacionados os conceitos básicos de LA bem como os pontos fortes e fracos que caracterizam essa teoria. Também serão mostradas algumas aplicações de LA visando identificar os métodos de extração de conhecimento baseados no estudo de relacionamentos entre os elementos de um domínio.

2.2 Definição

Em várias situações é possível identificar relacionamentos entre indivíduos, lugares, objetos ou mesmo conceitos, tais como os que acontecem entre pessoas que conversam por meio de ligações telefônicas, entre documentos de hipertextos ou entre pesquisadores, quando estes são coautores em publicações. Outros exemplos podem ser vistos em companhias aéreas e de transportes que ligam cidades, Estados e países. Também como exemplo de relacionamentos têm-se as citações bibliográficas e até mesmo um grupo de pessoas que se conhecem(Figura 2.1). Esses relacionamentos podem conter informações úteis e, para estudá-los, surgiu a técnica de *Link Analysis*.

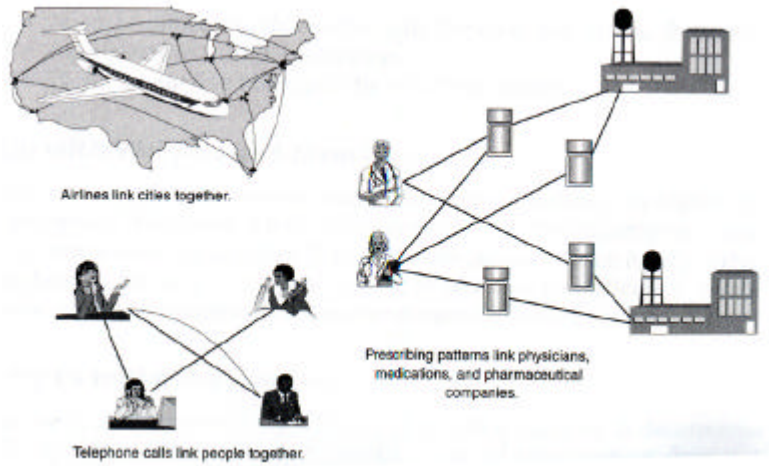

Figura 2.1 - Links representando relacionamentos
Fonte: BERRY e LINOFF (1997)

Trata-se de uma técnica de Mineração de Dados que tem por finalidade revelar a estrutura e o conteúdo de um conjunto de informações por meio de unidades (entidades ou objetos) interconectadas. A MD, por sua vez, pode ser definida como a aplicação de algoritmos de descoberta e análise de dados com o objetivo de encontrar padrões (ou modelos) sobre os dados (FAYYAD; PIATETSKY-SHAPIRO; SMYTH, 1996a). A MD é uma etapa do processo de descoberta de conhecimento que, por sua vez, consiste no campo que se concentra no desenvolvimento de técnicas e de ferramentas para a descoberta de conhecimento a partir de dados.

Na próxima seção apresenta-se o KDD com ênfase na etapa de MD.

2.3 Processo de descoberta de conhecimento

Segundo a visão de diversos autores (OLIVEIRA, 1992; MARTIN; SUBRAMANIAN; YAVERBAUM, 1996; HESSELBEIN et al., 1997; SCHREIBER et al., 2002), está se entrando em uma nova era em que o conhecimento é visto como o principal ativo das organizações. E para que uma organização possua um diferencial competitivo é necessário que tenha a capacidade de criar, gerenciar e distribuir esse conhecimento.

Ao mesmo tempo que vemos essa maior necessidade de novos conhecimentos para a tomada de decisões estratégicas, existe um crescente aumento da quantidade de dados gerados, os quais estão sendo produzidos e armazenados em

significativas quantidades. Como exemplos têm-se cartões de crédito e débito, compras feitas de casa, movimentações bancárias, internet, etc. Assim, percebe-se a necessidade de construção de ferramentas e metodologias que possibilitem a descoberta de conhecimento a partir desses dados.Nesse contexto, para auxiliar as organizações e/ou os indivíduos na obtenção de conhecimento, é possível contar com as contribuições oriundas da área de KDD.

Segundo Fayyad, Piatetsky-Shapiro e Smyth (1996a), KDD é um processo não trivial de identificação, a partir de dados, de padrões novos, válidos, potencialmente úteis e compreensíveis. Nessa definição, os dados representam um conjunto de fatos, e um padrão (*pattern*) é uma expressão em alguma linguagem que descreve um subconjunto de dados ou um modelo aplicável a esse subconjunto. Portanto, em KDD extrair um padrão consiste na atividade de adaptar um modelo aos dados ou descobrir alguma estrutura neles; ou, de maneira geral, encontrar alguma descrição de alto nível em um conjunto de dados.

O termo “processo” implica que KDD é composto de vários passos (Figura 2.2), os quais envolvem preparação dos dados, busca por padrões, avaliação do conhecimento e refinamento, que são repetidos em múltiplas iterações. Por “não trivial” entende-se que envolve alguma busca ou inferência e que não é apenas uma computação direta de valores predefinidos. Os padrões descobertos devem ser válidos perante os novos dados, com algum grau de certeza. Também é desejável que esses padrões sejam novos e potencialmente úteis. Isso quer dizer que eles devem trazer algum benefício para o usuário. Por último, os padrões devem ser compreensíveis. Se isso não for possível imediatamente, devem ser alvo, então, de algum método de pós-processamento. Na Figura 2.2 tem-se uma visão geral do processo de KDD.

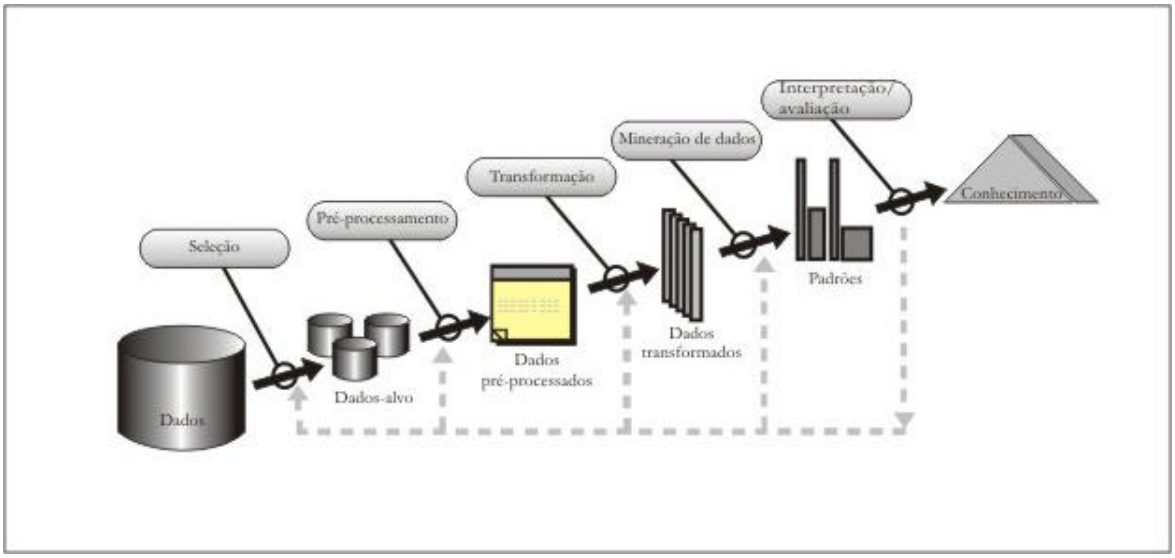

Figura 2.2 - Uma visão geral do processo de KDD
Fonte: adaptado de Fayyad, Piatetsky-Shapiro e Smyth (1996a)

Como pode ser visto na figura, trata-se de um processo repetitivo no qual todos os passos são importantes para se atingir o objetivo de descoberta de conhecimento. Deve ser visto como um método iterativo, e não como uma ferramenta de análise automática (MANNILA, 1996). O seu principal passo (e o mais estudado) é a Mineração de Dados, que é o assunto discutido na próxima seção.

2.3.1 Mineração de Dados

Enquanto KDD se refere a todo o processo de descoberta de conhecimento útil a partir dos dados, a Mineração de Dados (MD) é um passo específico desse processo. É importante fazer essa diferenciação entre os conceitos de MD e KDD devido a uma certa confusão existente na literatura sobre esses dois termos, como apontam Goebel e Gruenwald (1999) e Zhou (2003).

Segundo Goebel e Gruenwald (1999), o termo KDD é usado como o processo de tornar dados de baixo nível em conhecimento de alto nível, enquanto que a MD pode ser definida como a extração de padrões ou modelos dos dados analisados. A MD também pode ser entendida como “a exploração e a análise, por meios automáticos ou semi-automáticos, de grandes quantidades de dados, com o objetivo de descobrir padrões e regras significantes” (BERRY; LINOFF, 1997).

Quanto às metas da MD, Fayyad, Piatetsky-Shapiro e Smyth (1996b) apresentam dois tipos: verificação, em que o sistema é limitado a confirmar as hipóteses do usuário (teste de hipóteses); e descoberta, em que o sistema automaticamente encontra novos padrões. A descoberta é ainda dividida em (1) previsão, etapa em que o sistema procura padrões para a proposta de predição de comportamento futuro de algumas entidades (parte de diversas variáveis para prever outras variáveis ou valores desconhecidos); e (2) descrição, etapa em que o sistema procura por padrões com a proposta de apresentá-los ao usuário de forma compreensível.

Para alcançar as metas de KDD, Fayyad, Piatetsky-Shapiro e Smyth (1996b) apresentam as seguintes tarefas de MD:

- **classificação**: estabelecer uma função que mapeia (classifica) um item de dado em uma das várias classes predefinidas. Por exemplo, um pedido de crédito pode ser classificado como sendo de baixo, médio e alto risco;
- **regressão**: estimar um valor para alguma variável contínua, como, por exemplo, número de filhos em uma família ;
- **agrupamento (*clustering*)**: identificar um conjunto finito de categorias ou grupos (*clusters*) para descrever os dados. Nessa tarefa não há classes predefinidas, os itens de dados são agrupados de acordo com a semelhança que apresentam entre si. Por exemplo, agrupar clientes com comportamento de compra semelhante;
- **sumarização**: encontrar uma descrição compacta para um subconjunto de dados;
- **modelagem de dependência**: encontrar um modelo que descreve dependências significantes entre variáveis; e
- **detecção de desvio**: descobrir as mudanças mais significativas nos dados a partir de valores previamente medidos ou padronizados.

A Figura 2.3 apresenta, de forma resumida, descrições e exemplos das principais tarefas de KDD encontrados na literatura.

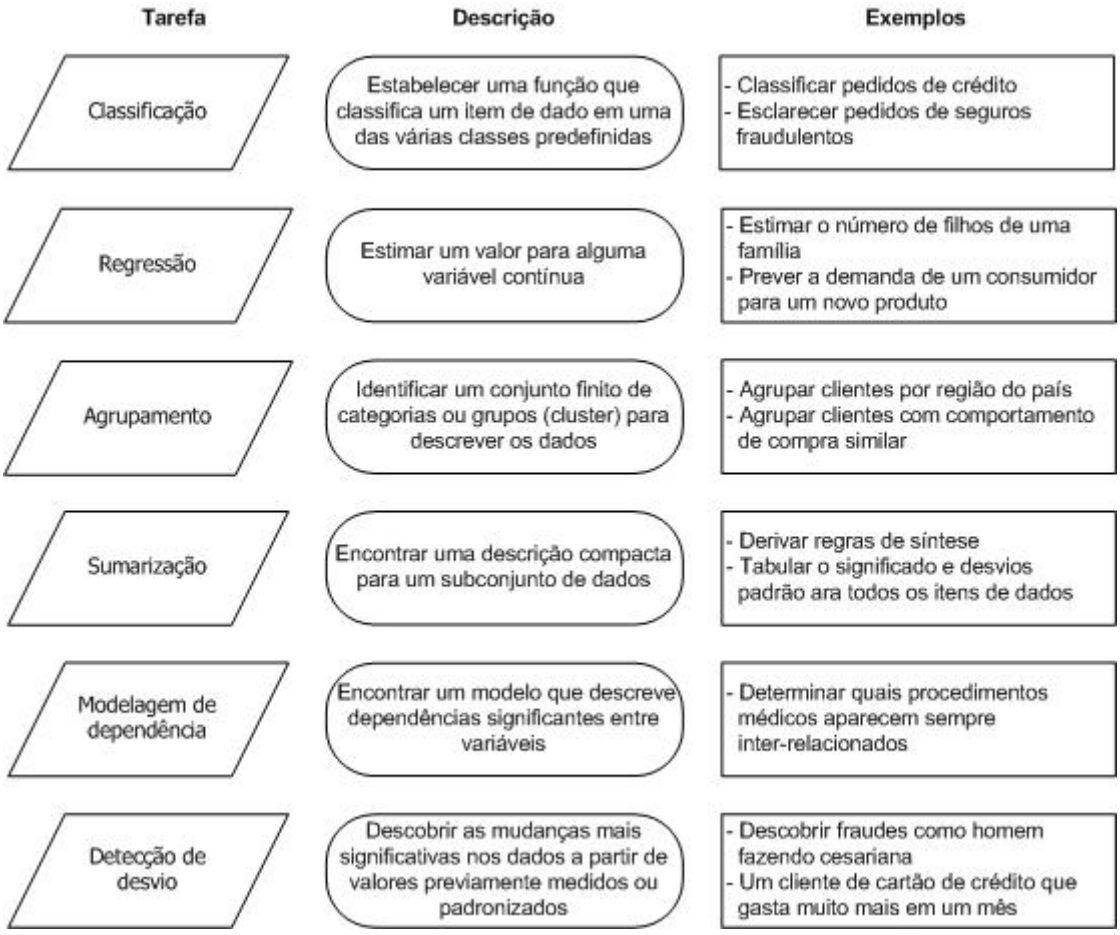

Figura 2.3 - Tarefas, descrição e exemplos de Mineração de Dados

A MD faz mais sentido quando há grandes volumes de dados. De fato, a maioria dos algoritmos de MD requer grandes quantidades de dados para ser possível construir e treinar os modelos que, então, serão usados para classificação, predição, estimativas ou outras tarefas de MD (BERRY; LINOFF, 1997).

Como já foi dito, na atualidade há um grande acúmulo de informações. Conforme Berry e Linoff (1997), armazenando-as se provê memória às organizações, embora de pouco uso se essa memória não for associada à inteligência, que nos permite olhar através de nossa memória e notar padrões, bem como fazer predições sobre o futuro. As ferramentas e técnicas de MD adicionam inteligência aos repositórios de informações, tais como *data warehouse*, internet, textos, etc.

Um exemplo disso pode ser visto no *marketing*, área na qual Berry e Linoff (1997) mostram algumas perguntas que podem ser respondidas com o uso de MD: Quem é mais provável de ser um cliente fiel e quem é mais provável de não ser mais cliente da empresa? Onde a próxima filial deveria estar localizada? Além dessas, há

também muitas outras perguntas, as quais atestam a importância no uso de técnicas e ferramentas de MD.

Existem algumas técnicas que podem ser utilizadas nas tarefas apresentadas na Figura 2.3. Segundo Harrison (1998), não há uma técnica que resolva todos os problemas de MD. Métodos diferentes são usados para propósitos diferentes, cada um com suas vantagens e desvantagens.

Berry e Linoff (1997) afirmam que "a escolha de uma particular combinação de técnicas a serem aplicadas em uma situação específica depende da natureza da tarefa de MD a ser executada e da natureza dos dados disponíveis". Para a escolha da técnica mais adequada é importante conhecer o domínio da aplicação de MD. Isso implica em conhecer os atributos importantes, os relacionamentos existentes, o que é útil para o usuário, e assim por diante. Além de *Link Analysis,* existem diversas técnicas encontradas na literatura que são utilizadas em MD: Redes Neurais Artificiais, Descoberta de Regras de Associação, Árvores de Decisão, Raciocínio Baseado em Casos, Algoritmos Genéticos, Conjuntos Difusos, etc (GOEBEL; GRUENWALD, 1999; BERRY; LINOFF, 1997; HARRISON, 1998).

Na próxima seção serão apresentados os conceitos básicos de *Link Analysis* encontrados na literatura.

2.4 Conceitos básicos de LA

A LA é baseada em uma área da matemática chamada Teoria dos Grafos[2] (TG). Como mencionado, a LA tem por objetivo desvendar a estrutura e o conteúdo de um conjunto de informações por meio de entidades interconectadas. Uma entidade (chamada de vértice ou nó) pode ser uma pessoa, um lugar, um documento, um objeto qualquer ou até mesmo um conceito. A conexão entre as unidades (chamada de aresta ou arco) representa a relação entre essas unidades.

Segundo Jensen (1998), muitos conjuntos de dados podem ser representados naturalmente como coleções de objetos ligados. Por exemplo, coleções de documentos podem ser representadas como documentos (vértices) conectados por

[2] Para maiores detalhes sobre Teoria dos Grafos, veja o Capítulo 3 deste trabalho.

citações e referências de hipertexto (arcos). Similarmente, organizações podem ser representadas como pessoas (vértices) conectadas por relacionamentos sociais e/ou padrões de comunicação (arestas). Outros exemplos são: coleção de dados de chamadas telefônicas (Figura 2.4), dados de transações financeiras entre contas bancárias, observações de encontros individuais, seus endereços, e outras interações comerciais e sociais.

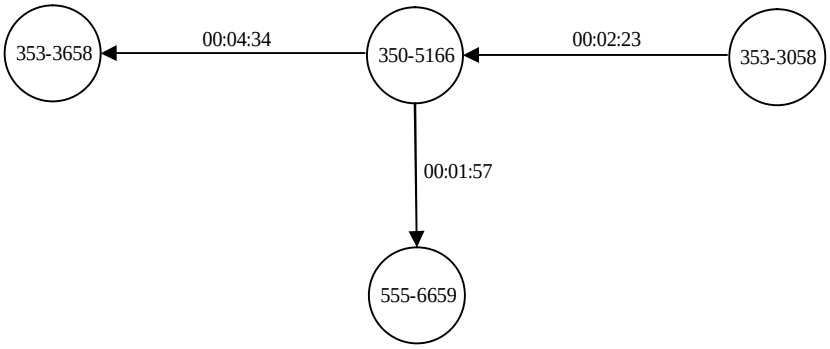

Figura 2.4 - Ligações telefônicas relacionando números de telefones

Na Figura 2.4 tem-se um exemplo de um grafo representando chamadas telefônicas: cada vértice representa um número de telefone; os arcos entre os vértices representam uma chamada; a orientação das setas indica quem originou a ligação; e os valores de cada link indicam a duração de cada ligação.

Esse tipo de representação de dados é também facilitado pela crescente disponibilidade de base de dados (orientadas a objetos e relacionais) e sistemas hipertextos. A representação na forma de relacionamentos está no âmbito da tarefa de Modelagem de Dependência em KDD. A natureza das bases de dados disponíveis na maior parte das organizações (*i.e.* bases relacionais) facilita a identificação de relacionamentos entre elementos de um domínio. Bases relacionais, orientadas a objetos e documentos hipertexto têm sua estrutura adequada ao tratamento de dependência. É nesse contexto que LA assume especial relevância.

A análise de tais dados, que pode ser realizada através de LA, está se tornado importante em diversos campos: investigações criminais, detecção de fraudes, epidemiologia, recuperação da informação, etc. Alguns dados ligados podem ser simples mas volumosos (e.g. chamadas de telefone), com uma uniformidade de nós

e tipo de links e com bastante regularidade. Outros dados podem ser extremamente ricos e variados, apesar de escassos (e.g. dados sobre investigação criminal), com elementos que possuem muitos atributos de domínio específico e também com valores que podem mudar com o tempo.

Segundo Goldberg e Senator (1998), para descobrir informações úteis e interessantes sobre uma pessoa específica ou sobre grupos de pessoas, é necessário primeiro identificar precisamente os indivíduos representados no banco de dados. Esse processo de tornar não ambíguo e de combinar a informação de identificação em chave única, as quais se referem a indivíduos específicos, é chamado de consolidação. E para, descobrir informações úteis, tais como anomalias que podem indicar fraude, freqüentemente se requer a construção de redes de indivíduos relacionando transações e um padrão de atividade. O processo de criar essas redes é chamado de formação de links, o qual pode ser usado em domínios em que há relacionamentos escondidos. A idéia de consolidação e formação de links foi apresentada na análise de pessoas e seus relacionamentos, masesses dois conceitos podem ser aplicados a outros tipos de entidades.

Um exemplo de formação de links é visto em Pinheiro e Sun (1998). Os autores desenvolveram um método para relacionar registros de diferentes bancos de dados. Nesse método foi usada uma medida de similaridade entre duas palavras. Assim, foi possível relacionar registros do tipo texto que não tinham um identificador único em comum.

Além das técnicas para consolidação e formação de links, em LA também se estuda como examinar, modificar, analisar, pesquisar e mostrar essas redes. Um de seus principais objetivos é a apresentação visual das relações para que o usuário possa melhor compreender o significado das inter-relações e, ainda, ver relações desconhecidas. Contudo, há limitações para a apresentação visual das informações. Nesse sentido, Grady, Tufano e Flanery Junior (1998) afirmam que há a necessidade de novas técnicas para organizar a exibição das informações eque vêm sendo realizadas pesquisas na área de HCI (*Human Computer Interaction*) com o objetivo de desenvolver interfaces mais adequadas para apresentação das informações ao usuário.

Como afirma Lyons (1998), as questões a seguir são freqüentemente consideradas em *Link Analysis*.

- Quais nós são chaves ou centrais(hubs) na rede formada?
- Quais links podem ser reforçados para aumentar a eficiência das operações da rede?
- É possível descobrir links ou nós não detectados a partir dos dados conhecidos?
- Existem similaridades na estrutura de partes da rede que podem indicar um relacionamento não conhecido?
- Quais são as sub-redes relevantes dentro de uma rede com muitos nós?
- Quais modelos de dados e níveis de agregação melhor revelam certos tipos de relações e sub-redes?

2.5 Aplicações de Link Analysis

Como vimos anteriormente, a *Link Analysis* pode ser utilizada em muitas áreas, de epidemiologia a detecção de fraudes, de investigação criminal a estudo de redes sociais. Dados relacionados são tipicamente modelados como um grafo, com nós representando entidades de interesse ao domínio e links representando relacionamentos.

Um exemplo de área de aplicação de LA é *database marketing*. Tal aplicação pode revelar características típicas dos melhores consumidores e/ou identificar padrões no comportamento das compras. Um outro campo de aplicação é a detecção de fraude, na qual o sistema ajuda na descoberta de situações suspeitas LA também é aplicada em operações referentes a seguros médicos (nas relações entre pacientes, médicos, procedimentos, seguros, etc) e na análise de comunicações, para a apresentação de padrões de comunicação e potenciais gargalos na rede.

Outro exemplo de área de aplicação é apontado por Grady, Tufano e Flanery Jurior (1998). Segundo esses autores, analistas de inteligência devem relacionar grandes quantidades de dados sobre pessoas-chave em políticas estrangeiras, terroristas, narcóticos e outras organizaçõesalvo. Desse modo, as análises são realizadas através de buscas por padrões e relacionamentos nessas informações.

Isso é correntemente um trabalho intensivo e que consome muito tempo. Uma ferramenta de LA pode mostrar uma visão geral da estrutura organizacional, identificar pontos de articulação e investigar lavagem de dinheiro e padrões de transferência ilegal de produtos.

Uma ferramenta de LA chamada de Watson (ANDERSON et al., 1994) busca e identifica associações entre entidades consultando bancos de dados. Dada uma entidade, essa ferramenta pode automaticamente realizar uma consulta para buscar registros relacionados. Esses registros são ligados à entidade dada, e o resultado é apresentado em forma gráfica. Assim, um analista pode examinar esses registros e relacionamentos para tentar descobrir pistas úteis para futuras investigações.

Outra aplicação de LA, chamada de *Link Discovery Tool* (HORN; BIRDWELL; LEEDY, 1997) foi desenvolvida para descobrir grupos de indivíduos e organizações que são fortemente relacionados por associações em redes criminais. Essa ferramenta também usa algoritmos de caminho mais curto[3] para tentar descobrir associações entre pessoas que aparentemente não estão relacionadas.

Goldberg e Senator (1998) mostraram uma aplicação de LA chamada de *FinCEN Artificial System* (FAIS), que é um exemplo de um sistema de descoberta de conhecimento que busca irregularidades em banco de dados, as quais podem indicar fraude. O objetivo do FAIS é identificar indicativos de lavagem de dinheiro em um banco de dados de transações financeiras. No FAIS, consolidação é importante para identificar, de forma única, as entidades do mundo real (nesse caso, uma pessoa, negócio ou conta). A formação de links é realizada para tentar relacionar as entidades. Isso é necessário porque lavagem de dinheiro raramente é manifestada por uma única transação ou por uma única entidade, mas preferencialmente por um padrão de transações ocorrendo sobre o tempo e envolvendo um conjunto de entidades relacionadas.

Baldwin e Bagga (1998) apresentaram uma arquitetura de um sistema para relacionar entidades (pessoa, lugar, evento ou conceito) que aparecem em diferentes documentos de texto. Esse método, que utiliza técnicas de processamento de linguagem natural, pode ser usado para aplicação de LA sobre repositórios de informações em formato texto.

---

[3] Ver o capítulo 3 deste trabalho.

Lee (1998) descreve alguns sistemas que foram projetados para ajudar analistas de inteligência na descoberta e investigação de indivíduos e organizações envolvidos em atividades ilegais, tais como tráfico de drogas, terrorismo, etc. É apresentada a possibilidade de se construir uma ferramenta que utiliza técnicas de processamento de linguagem natural para popular bases de dados relacionais com informação detalhada a partir de mensagens de texto. Essa informação inclui não apenas entidades, mas também eventos e associações. A informação é estruturada de um modo que permite análises úteis realizadas por meio de LA.

Lyons e Tseytin (1998) estudaram a possibilidade de se aplicar LA no estudo de padrões de comportamento de passageiros de ônibus. Nesse conjunto de dados, as estações podem ser consideradas como nós, e as jornadas dos passageiros, como links entre eles. O objetivo desse tipo de aplicação é conhecer melhor o fluxo de passageiros, saber se é necessário disponibilizar novas linhas de ônibus, novas estações, etc.

O software PolyAnalyst (KISELEV, 1994; KISELEV; ANANYAN; ARSENIEV, 1998) possui um conjunto de ferramentas de Mineração de Dados. Entre essas ferramentas há um módulo para se trabalhar com LA, o qual revela padrões complexos de correlações entre valores dos atributos representando-os visualmente. Resultados da análise são apresentados como um grafo de objetos conectados que suporta vários tipos de manipulação e operações *drill-down*[4]. A saída visual de LA facilita um melhor entendimento de estruturas escondidas dos dados investigados e ajuda rapidamente a isolar padrões de interesse para uma posterior investigação. Na Figura 2.5, apresenta-se um exemplo de uso do software PolyAnalyst.

[4] Capacidade de passar rapidamente entre menus extraídos e visualizar as hierarquias e seus níveis.

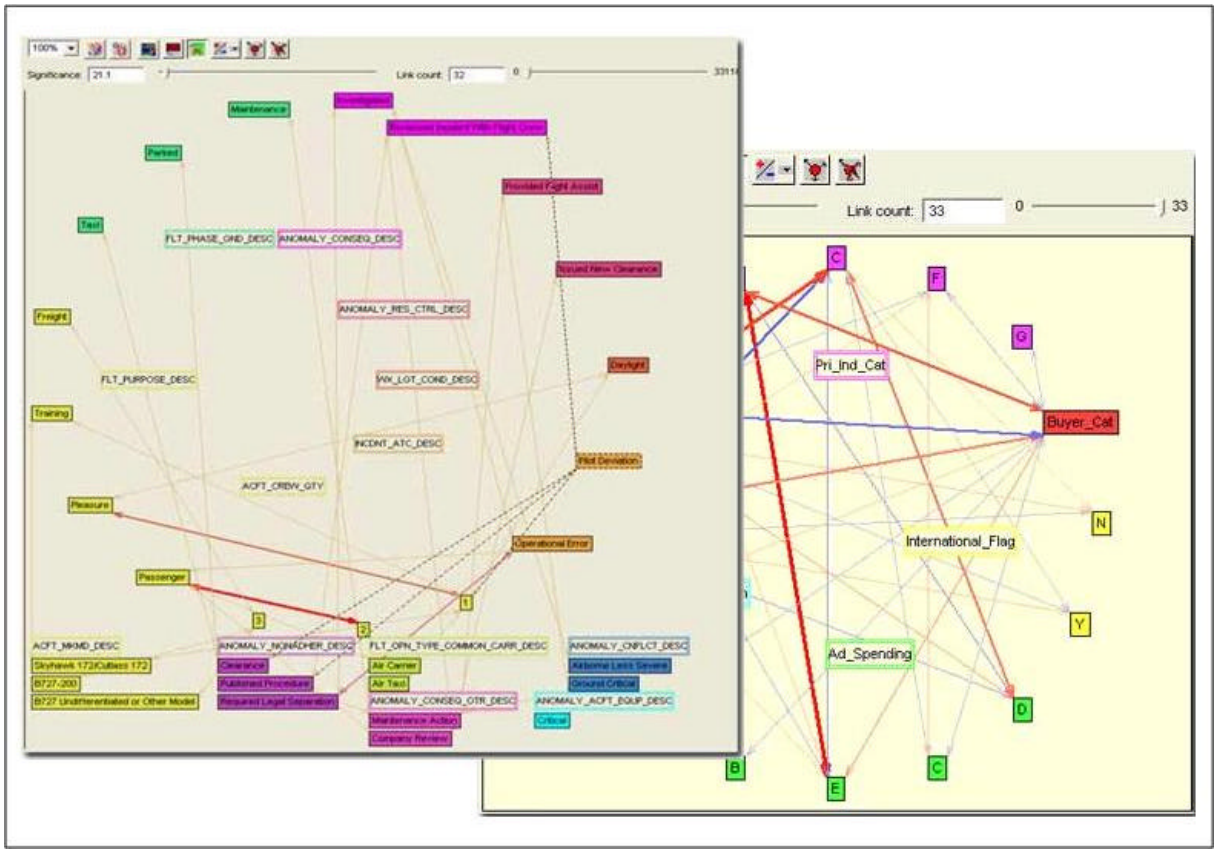

Figura 2.5 - Exemplo de *Link Analysis* no software PolyAnalyst

O PolyAnalyst pode ser usado em *database marketing*, detecção de fraudes, análise de comunicações etc. Esse software também pode ser utilizado para analisar textos. A habilidade de mostrar visualmente e posteriormente investigar e agrupar termos extraídos de notas textuais provê um outro componente interessante na análise de dados não estruturados.

Outra aplicação comercial de LA é o VisuaLinks® (VISUALINKS, 2004), que é uma ferramenta de análise gráfica usada para descobrir padrões, tendências, associações e redes ocultas em qualquer número e fonte de dados. O VisuaLinks® (Figura 2.6) apresenta os dados visualmente, mostrando padrões e relacionamentos escondidos.

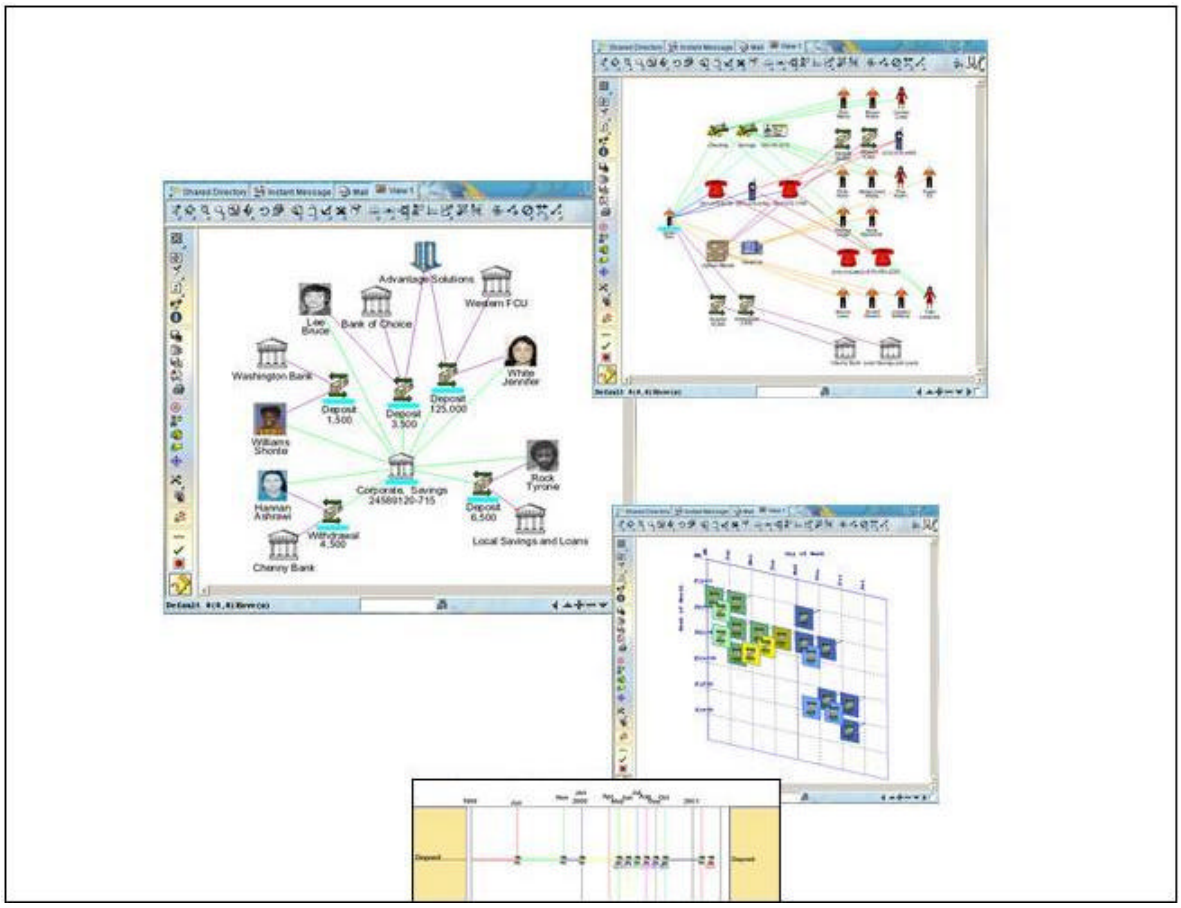

Figura 2.6 - Exemplos de *Link Analysis* na ferramenta VisuaLinks.

Um dos módulos do VisuaLinks é o Network Miner, que foi projetado para revelar grupos ou redes de informações relatadas. O Network Miner realiza buscas nos dados procurando tipos particulares de dados que são relacionados por tipos específicos de associações (links). Essa ferramenta mostra entidades conectadas e revela redes ocultas de associações para descobrir objetos de dados fortemente conectados que podem ser alvos de análise e investigação posteriores.

Hauck, Chau e Chen (2002) apresenta um protótipo de aplicação chamado de COPLINK para utilização em investigações criminais. Uma das ferramentas dessa aplicação, o COPYLINK Detect, usa algumas técnicas estatísticas para identificar relacionamentos entre objetos (termos ou conceitos) de interesse (LESK, 1997). Esse protótipo apresenta uma rede de termos e associações ponderadas que representam os conceitos e suas associações dentro de um espaço de informações. Além disso, a análise de co-ocorrência utiliza funções de *clustering* e similaridade (CHEN; LYNCH, 1992) para ponderar relacionamentos entre todos os possíveis

pares de conceitos. A rede resultante possui todas as possíveis associações entre objetos, o que significa que todos os links existentes entre cada par de conceitos são armazenados. O COPYLINK Detect é uma ferramenta que pode ser usada para detectar a presença ou a ausência de links entre pessoas, lugares, veículos e outros tipos de objetos importantes em investigações criminais.

Outro campo de aplicação de LA é a Web. Trata-se da maior, e que mais rapidamente cresce, coleção de documentos do mundo. Uma de suas características dominantes é a natureza interligada de seus documentos. Como a Web é formada basicamente por páginas (nós) e hiperlinks (relações), há uma relação direta com problemas de *Link Analysis*. As aplicações mais conhecidas são os sites de buscas na Web. Os algoritmos PAGERANK (BRIN; PAGE, 1998) e HITS (KLEINBERG, 1998) bem como algumas de suas extensões (NG; ZHENG, JORDAN, 2001; MILLER et al., 2001; KAMVAR et al., 2003) são métodos de LA que usam a estrutura de links da Web para calcular um indicador de importância para cada página. Esses indicadores são usados para classificar páginas sobre um determinado assunto. Assim, os sites de buscas na Web utilizam esse indicador para responder a consultas de buscas feitas pelos usuários.

Um outro exemplo de aplicação de LA na Web é apresentado por Craven et al. (2000), que dizem que há uma grande quantidade de informações disponíveis na Web, mas que essas informações são compreensíveis somente para humanos. Os autores apresentam um protótipo de uma aplicação que permite a criação automática de uma base de conhecimento que seja compreensível por computadores e cujo conteúdo reflita as informações contidas na Web. Uma vez montada, tal base permitiria a recuperação mais eficiente de informações da Web e promoveria inferências baseadas em conhecimento e em resolução de problemas. O método proposto consiste em desenvolver um sistema de extração de conhecimento treinável que toma duas entradas: (1) uma ontologia que define as classes (e.g., companhia, pessoa, empregado, produto) e as relações (e.g., empregado_por, produzido_por) de interesse quando se está criando as bases de conhecimento; (2) um conjunto de dados de treinamento que consiste de documentos (*labeled regions*) de hipertexto, os quais representam instâncias dessas classes e relações. Dadas essas entradas, o sistema aprende como extrair informação de outras páginas e hiperlinks na Web.

2.6 Pontos fortes e fracos da LA

Segundo Berry e Linoff (1997), a técnica de *Link Analysis* possui alguns pontos fortes, entre os quais se destacam os relacionados a seguir.

- **É apropriada para dados relacionais**. Como mostrado nos exemplos anteriores, muito problemas decorrentes da Mineração de Dados naturalmente envolvem links, o que torna interessante o uso de LA
- **É útil para visualização**. A visualização direta de links poder contribuir muito para a descoberta de conhecimento. LA oferece um modo alternativo de ver os dados, diferente do formato dos bancos de dados relacionais e das ferramentas OLAP (*On-Line Analytical Processing*).

Segundo esses autores, *Link Analysis* também possui alguns pontos fracos, como os que se seguem.

- **Não é aplicável a muitos tipos de dados**. Apesar de a LA ser poderosa quando aplicável, o seu uso não é apropriado para determinados tipos de problemas. Muitos tipos de dados simplesmente não são apropriados para LA
- **É implementada por poucas ferramentas**. Não há muitas ferramentas desenvolvidas para LA. A maioria delas é utilizada para propósitos específicos, tais como para as ferramentas existentes na área de investigação criminal.
- **Uso de banco de dados relacionais pode ser ineficiente**. As estruturas de dados em geral demandam um processamento elevado, uma vez que as implementações que utilizam seleções encadeadas em bancos de dados relacionais podem ser ineficientes.

2.7 Considerações finais

Este capítulo abordou os processos referentes a KDD e Mineração de Dados, com ênfase para a técnica de *Link Analysis*. Notou-se que ferramentas inteligentes de exploração de dados são interessantes paraauxiliar especialistas na visualização

da conectividade entre objetos à medida que a quantidade de dados aumenta. No próximo capítulo discorrer-se-á sobre a Teoria dos Grafos, que pode ser combinada com LA no processo de extração de conhecimento a partir do relacionamento entre unidades de informação.

3 TEORIA DOS GRAFOS

3.1 Introdução

A Teoria dos Grafos (TG) é um ramo da matemática que teve início no século XVIII com o matemático Leonard Euler (1707-1783). O desenvolvimento da TG ficou esquecido por muitos anos e teve maior impulso com as aplicações voltadas a problemas de otimização já na segunda metade do século XX. Na atualidade ela está dentro do conjunto de técnicas que compõem a pesquisa operacional. Tal desenvolvimento ocorreu devido ao surgimento dos computadores, sem os quais muitas das aplicações da TG seriam impossíveis.

Neste capítulo serão apresentadas algumas definições, noções básicas e formas de representação de grafos. Serão mostradas também aplicações com vistas a identificar métodos aplicáveis à extração de conhecimento a partir de informações representadas na forma de relacionamentos entre os elementos de um domínio.

3.2 Definições

Há várias definições sobre o que é um grafo. Existem algumas diferenças nas definições e terminologias apresentadas na literatura. Apesar dessas variações, um grafo G pode ser visto como um par (V, A), onde V é um conjunto não vazio $\{v_1, v_2, ..., v_n\}$ e seus elementos são denominados vértices, e A é uma família $(a_1, a_2, ..., a_m)$ de elementos pertencentes ao produto cartesiano V × V, chamados de arestas.

Na Figura 3.1, por exemplo, pode-se ver uma representação de um grafo G = (V, A) onde V = {a, b, c, d} e A = {(a, b), (a, c), (a, d), (c, d), (b, d)}.

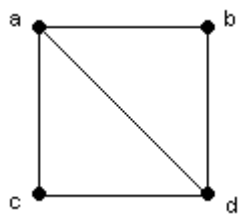

Figura 3.1 - Exemplo de grafo

Nesse caso considera-se que as relações são simétricas. Por exemplo, a aresta (a, b) poderia ser designada por (b, a), pois as arestas não possuem orientação. Portanto, esse é um grafo não orientado.

Observe, agora, o grafo G = (V, A) mostrado na Figura 3.2, onde V = {a, b, c, d} e A = {(a, b), (c, a), (d, c), (d, b)}. Como pode ser visto, as relações desse grafo não são simétricas. Por exemplo, a relação (a, b) não poderia ser representada por (b, a), pois estamos considerando que existe orientação nas suas relações. Portanto, G é chamado de grafo orientado (também chamado de digrafo). Em um grafo orientado, as relações entre os vértices são chamadas de arcos.

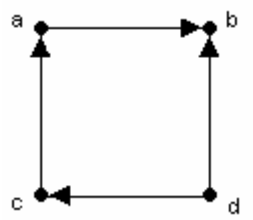

Figura 3.2 - Exemplo de grafo orientado

De acordo com Boaventura Netto (2001), existe uma equivalência entre grafos não orientados e orientados. Isso quer dizer que todo grafo não orientado G possui um grafo orientado G' e que cada aresta de G estará associada a um par de arcos opostos em G'. Portanto, é preferível não distinguir os grafos em orientados e não orientados quanto às suas aplicações. O que ocorre é que há conceitos que exigem orientação, e outros não. Existem, também, os denominados grafos mistos, que são aqueles que possuem ligações orientadas e não orientadas (veja Figura 3.3).

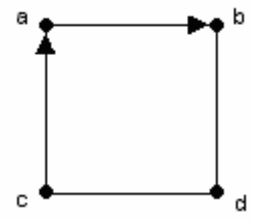

Figura 3.3 - Exemplo de grafo misto

Para facilitar o entendimento dos conceitos sobre grafos, neste trabalho não se fará distinção entre os termos *arco* e *aresta*. Para a palavra *vértice* também é comum o uso dos termos *nós*, *nodos* ou *pontos*.

3.3 Noções básicas

A seguir é apresentada uma descrição das noções básicas sobre TG. Definições mais formais podem ser encontradas em Gondran e Minoux (1985), Berge (1985), West (1996) e Boaventura Netto (2001).

- **Ordem**. A ordem de um grafo é determinada pelo número de vértices que ele possui. O grafo da Figura 3.1, por exemplo, é de ordem 4.
- **Adjacência**. Dois vértices x e y de um grafo orientado G são adjacentes (ou vizinhos) se existir um arco ou uma aresta ligando x a y. Para grafos orientados esse conceito pode ser dividido em sucessor e antecessor. Um grafo orientado G x será sucessor de y ser houver um arco iniciando y e atingindo x. Da mesma forma, x será antecessor de y se o grafo possuir um arco que parte de x e chega em y.
- **Grau**. O grau de um vértice v de um grafo G é fornecido pelo número de arestas que lhe são incidentes. Se o grafo for orientado, pode-se dividir o conceito de grau de um vértice v em dois: (1) grau de emissão e (2) grau de recepção. O primeiro corresponde ao número de arcos que saem de v. E, por conseguinte, o grau de recepção corresponde ao número de arcos que chegam em v. Se todos os vértices de um grafo G têm o mesmo grau, então se diz que G é um grafo regular.
- **Fonte**. Um vértice v de um grafo orientado G é uma fonte se o grau de recepção desse vértice for igual a zero. Por exemplo, o vértice d do grafo mostrado na Figura 3.2 é uma fonte.
- **Sumidouro**. Um vértice v de um grafo orientado G é um sumidouro se o grau de emissão desse vértice for igual a zero. Por exemplo, o vértice b do grafo mostrado na Figura 3.2 é um sumidouro.
- **Laço**. Um laço é uma ligação do tipo a = (v, v). Essa ligação conecta um vértice a ele mesmo.
- **Grafo regular**. Um grafo é chamado de regular quando todos os seus vértices possuem o mesmo grau. O grafo da Figura 3.4, por exemplo, é um grafo regular.

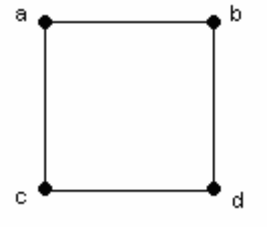

Figura 3.4 - Grafo regular

- **Grafo completo**. Um grafo completo possui uma aresta para cada par de vértices. Se esse grafo é de ordem n, então ele pode ser chamado de K_n. É também chamado de grafo regular-(n-1), pois todos os seus vértices têm grau n-1. A Figura 3.5 mostra um exemplo de grafo completo.

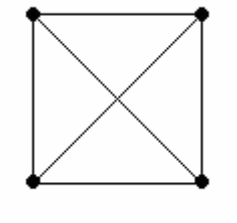

Figura 3.5 - Grafo completo

- **Grafo rotulado**. Como foi apresentado anteriormente, o V de G = (V, A) é um conjunto. Isso implica que os elementos de V, que são os vértices do grafo, sejam identificados. Portanto, os grafos aqui mencionados são chamados de rotulados. Esse tipo de grafo possui uma palavra ou um valor numérico associado a cada vértice. Os grafos das Figuras 3.1, 3.2 e 3.3 são exemplos desse tipo de grafo.
- **Grafo valorado**. Um grafo G = (V, A) é valorado se existe uma ou mais funções associando V ou A a um conjunto de números. Isso ocorre, por exemplo, em um grafo representando as populações de uma cidade (vértices) e a distância entre elas (ligações). A Figura 36 apresenta o exemplo de um grafo valorado sobre suas ligações.

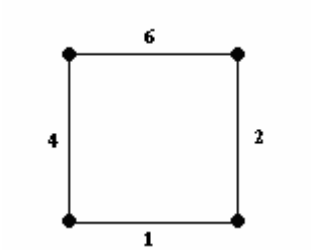

Figura 3.6 - Grafo valorado

- **Partição de grafos**. Um grafo G = (V, A) é dito k-partido quando o seu conjunto de vértices V puder ser dividido em V_1, V_2, ..., V_k, de tal maneira que não haja vértices adjacentes no mesmo conjunto. A Figura 3.7 apresenta um grafo 2-partido (ou bipartido). O grafo dessa figura também é bipartido completo, pois todos o vértices da partição V_1 estão ligados a todos os vértices da partição V_2.

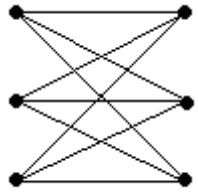

Figura 3.7 - Grafo 2-partido (bipartido) completo

- **Multigrafo**. Um grafo G = (V, A) pode ser chamado de multigrafo quando existem múltiplas arestas entre pares de seus vértices. O grafo apresentado na Figura 3.8 é um exemplo de multigrafo, pois há duas arestas ligando os vértices a e b.

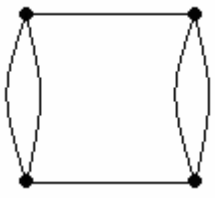

Figura 3.8 - Exemplo de multigrafo

- **Subgrafo**. Um grafo $G_s=(V_s, A_s)$ é dito subgrafo de um grafo G=(V, A) quando $V_s \subseteq V$ e $A_s \subseteq A$. Por exemplo, o grafo da Figura 3.9 é subgrafo do grafo da Figura 3.8.

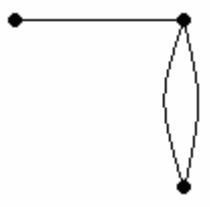

Figura 3.9 - Exemplo de subgrafo

A seguir serão apresentadas algumas possíveis formas de representação de grafos.

3.4 Representação de grafos

As figuras anteriores (3.1 a 3.5) mostram grafos representados de forma esquemática. Tal forma é muito útil para humanos entenderem o grafo, mas é necessário um modelo matemático que possa ser armazenado e processado por computadores. Três maneiras comuns de se representar numericamente um grafo são apresentadas a seguir.

3.4.1 Lista de adjacência

É uma forma simples e econômica (do ponto de vista computacional) de se representar um grafo. É constituída por listas de vértices, e cada lista é formada por um vértice inicial e pelo conjunto de vértices com os quais ele possui uma aresta em comum. No caso dos grafos orientados, a lista de vértices pode ser formada de duas maneiras: (1) pelo conjunto de vértices que recebem um arco do primeiro vértice ou (2) pelo conjunto de vértices dos quais sai um arco em direção ao primeiro. Portanto,

um grafo orientado possui duas listas de adjacência equivalentes. Isso pode ser visto na Figura 3.10.

Vértices	
Origem	Destino
1	2 4 5
2	3 6
3	2 5 7
4	4 6 7
5	3 8
6	1 4 5 8
7	2 3 4
8	

Vértices	
Destino	Origem
1	6
2	1 3 7
3	2 5 7
4	1 4 6 7
5	1 3 6
6	2 4
7	3 4
8	5 6

Figura 3.10 - Lista de adjacência

Na primeira lista de adjacência mostrada acima tem-se o vértice 7 associado a uma lista vazia. Isso ocorre porque não há arcos partindo dele. Então, esse vértice é um sumidouro do grafo.

3.4.2 Matriz de adjacência

É muito comum a representação de grafos através de matrizes. A mais utilizada é a matriz de adjacência. Trata-se de um matriz n x n, sendo n a ordem do grafo. Nessa matriz, os vértices do grafo são distribuídos em suas linhas e colunas. Os valores a_{ij} de uma matriz A serão 0 ou 1 (no caso de grafos não valorados), de acordo com a regra:

a_{ij} = 1, se existe algum vértice (i, j) em V;

a_{ij} = 0, caso não exista algum vértice (i, j) em V.

Na Figura 3.11 pode ser visto um exemplo de um grafo e de sua matriz de adjacência.

	1	2	3	4
1	0	1	0	1
2	0	1	1	1
3	1	0	0	0
4	0	1	1	0

Figura 3.11 - Matriz de adjacência

No caso dos grafos valorados sobre as suas ligações, os próprios valores das ligações substituiriam os “1” da matriz, a qual pode ser chamada de matriz de valores. Se o grafo for não orientado, têm-se uma matriz triangular, pois não há distinção no sentido de suas arestas.

3.4.3 Matriz de incidência

Este tipo de matriz representa um grafo a partir de suas arestas. Como exige muitas vezes a alocação de uma matriz maior do que no método da matriz de adjacência, não é tão utilizada quanto aquela. A matriz alocada deverá ter dimensões C(V) x C(A).

Para grafos não orientados, o princípio dessa representação está na seguinte regra:

m[i,j] = 1 se o vértice i incide com a aresta j, 0 caso contrário.

Para grafos orientados a regra é:

m[i,j] = 1 se existe uma aresta (i,j), -1 se existe uma aresta (j,i) e 0 caso não exista uma ligação entre i e j.

3.5 Conceitos, métodos e áreas de aplicação

A Teoria dos Grafos surgiu na busca por solução de um problema prático. Tal problema, chamado de Pontes de Konigsberg (Kaliningrad), apresenta a seguinte questão: “será possível a um cidadão sair de sua casa, passar por cada uma das 7 (sete) pontes apenas uma só vez e retornar a sua casa?”. Esse problema foi codificado como um grafo no qual as áreas terrestres eram representadas como vértices, e as pontes, como as ligações entre eles.

A era dos computadores possibilitou o desenvolvimento de várias aplicações na teoria dos grafos. Entre estas, tem-se em química orgânica a enumeração dos isômeros dos hidrocarbonetos alifáticos saturados. Em outras áreas têm-se a análise de redes sociais, a teoria das árvores nas aplicações à computação e o uso da teoria

dos fluxos em redes nas aplicações de pesquisa operacional em transportes e comunicações. Em pesquisas mais recentes, tem sido feitas aplicações à síntese orgânica e à interpretação da estrutura do DNA.

3.5.1 Distância

A idéia de distância está relacionada ao grau de afastamento (ou custo, capacidade de transmissão, fluxo, etc.) de um vértice em relação aos outros vértices do grafo. Os grafos aqui considerados são valorados sobre as ligações, sendo os valores normalmente associados a custos ou comprimentos. Em um grafo G=(V, A), a distância entre dois vértices (v_1, v_2) será o menor caminho entre eles no grafo. Essa distância pode ser calculada com algoritmos de caminho mais curto (GONDRAN; MINOUX, 1985; BERGE, 1985; WEST, 1996; BOAVENTURA NETTO, 2001; XU, 2000). Esse conceito pode ser aplicado, por exemplo, em transportes, quando se procura o menor caminho entre duas localizações. Outra aplicação interessante está em redes sociais, em que é possível estudar distanciamento entre indivíduos tendo-se um grafo que represente relações sociais de um grupo de pessoas, pode-se estudar distanciamento entre indivíduos. Devido ao fato de que grande parte dos indivíduos não são conectados diretamente à maioria dos outros indivíduos em uma população, pode ser muito importante ir além de simplesmente examinar as conexões imediatas entre os atores. O conceito de distância poderia ser aplicado para se descobrir a distância entre dois pesquisadores em rede de co-autoria científica, como realizado por Newman (2000). Outro exemplo desse tipo de aplicação pode ser encontrado em Erdos (2004). Usando-se a noção de distância, pode-se também calcular uma medida chamada *betweeness*, que mede o grau de intervenção de um ator sobre os outros atores da rede. A medida de *betweeness* de um vértice (ou ator) v é definida como sendo o número total de caminhos mínimos entre pares de vértices os passam por v (NEWMAN, 2000).

3.5.2 Centros, medianas e anticentros

Antes de se tratar de centros, anticentros e medianas, é necessário que dois conceitos sejam estudados: *afastamento* e *raio*. O afastamento de um vértice x ? V em um grafo G=(V, A) é a maior distância de x a algum y ? V. Já o raio de um grafo G= (V, A) é o menor dos afastamentos existentes no grafo. Assim, o centro de um grafo G=(V, A) é um vértice de afastamento igual ao raio. A idéia de centro ajuda a encontrar os melhores locais para a instalação de serviços de emergência (bombeiros, polícia, hospital, etc.), visto que se minimiza a maior distância. Outra aplicação do conceito de centro está na análise de redes sociais Propriedades formais de centralidade foram inicialmente investigadas por Bavelas (1950). Nesse caso, estuda-se a localização do ator em relação à rede total, identificando-se indivíduos considerados "importantes" na rede.

A mediana de um grafo G = (V, A) é um vértice para o qual a soma das distâncias aos demais vértices é mínima em relação a V. Uma mediana é a solução para o problema de localização de um serviço comercial de entregas no qual apenas um local seja atendido de cada vez, como, por exemplo, a localização de um serviço interurbano em que cada cidade seja atendida por uma entrega separada.

O anticentro de um grafo G = (V, A) é um vértice cuja menor distância em relação a algum outro vértice é máxima. Um anticentro pode ser usado para localizar um serviço cuja proximidade seja incômoda, como um depósito de lixo, por exemplo. Para um grafo G, os valores para esses três conceitos – centro, mediana e anticentro – podem ser obtidos a partir da matriz de caminhos mínimos entre todos os vértices de G. Essa matriz pode ser calculada usando-se o algoritmo de Floyd, que possui boa performance e grande simplicidade (BOAVENTURA NETTO, 2001).

3.5.3 Densidade

É a proporção de laços efetivos entre laços possíveis (HANNEMAN, 2000; SCOTT, 2000). Trata-se de uma medida do grau de inserção dos atores na rede. É uma tentativa de sumarizar a distribuição geral de ligações com o objetivo de

medir o quão distante o grafo está de ser completo (com todas as ligações possíveis). A fórmula para se calcular a densidade é:

$$\frac{l}{n(n-1)/2}$$

onde *l* é o número de ligações existentes no grafo e *n* é o número de vértices. Essa medida pode variar de 0 a 1, sendo 1 a densidade de um grafo completo. Tal conceito poderia, por exemplo, ser usado para verificar as densidades de grupos de pesquisa.

3.5.4 Fecho transitivo

Em grafos orientados tem-se a noção de vértices sucessores e antecessores. Se essa noção for aplicada iterativamente, leva-se a determinação de conjuntos que traduzem a ligação direta e indireta entre esses vértices. Tais conjuntos são denominados fechos transitivos. Essa noção está conceitualmente ligada às idéias de comunicação e controle. Se tivermos um grafo G = (V, A) sendo que V = {*habitantes de uma cidade*} e A = {(x, y) | <x conhece y>}, o fecho transitivo direto de uma pessoa k será o conjunto de pessoas que poderão tomar conhecimento de uma informação de que k disponha, através de comunicação pessoa a pessoa. Assim, esse conceito poderia ser usado parase estudar o efeito da comunicação informal bem como poderia ser aplicado para se encontrar uma medida chamada de *reachability,* a qual mede a extensão do contato que um ator tem com outros na rede (HANNEMAN, 2000). Um ator é “alcançável” por outro se existe algum conjunto de conexões pelo qual se pode traçar um caminho da fonte até o ator-alvo, sem considerar quantos outros atores estão entre eles.Kaufmann (1968) e Roy (1969) apresentam um algoritmo simples para a determinação dos fechos transitivos de um grafo.

3.5.5 Ponto de articulação

Um vértice é dito um ponto de articulação se sua remoção (juntamente com as arestas a ele conectadas) provoca uma redução na conexidade do grafo. Em um grafo orientado, as principais aplicações da teoria dos pontos de articulação estão na análise de redes sociais e, como conseqüência, na administração de recursos humanos. A verificação da existência de um ponto de articulação pode levar a um melhor direcionamento de esforços ou de recursos em relação ao aperfeiçoamento e à promoção de determinadas pessoas. Os algoritmos de Ford e Fulkerson (LOPES, 1980) podem ser usados para a determinação dos pontos de articulação de um grafo, caso existam. Aplicações e redes sociais podem ser encontradas em Frujuelle (1990).

3.5.6 Problema do labirinto

Segundo Boaventura Netto (2001), há certas situações na quais se procura um caminho entre dois vértices de um grafo sem qualquer consideração quantitativa. Nesses casos, o problema estará resolvido quando o caminho for encontrado. Esse grafo normalmente é sem orientação e representa um conjunto de decisões possíveis a serem tomadas em uma certa seqüência cuja determinação corresponde à solução do problema. Essa é a situação de quem se encontra em um labirinto, daí o nome “problema do labirinto”. O algoritmo mais comum para resolver esse tipo de problema é o algoritmo de Trémaux (BERGE, 1973).

3.5.7 Caminho de valor máximo

A busca pelo caminho de valor máximo envolve a seqüenciação de atividades em algum tipo de trabalho que possa ser dividido em etapas. Essa técnica permite a determinação do tempo total mínimo e também a realocação de recursos no sentido de melhor aproveitá-los (BOAVENTURA NETTO, 2001). Para resolver esse tipo de problema, tem-se o algoritmo de Pert (GONDRAN; MINOUX, 1985).

3.5.8 Árvore parcial mínima

A árvore parcial mínima trata de um problema de interligação ótima em grafos não orientados que representam modelos de redes nas quais algum tipo de serviço é distribuído e o custo de cada elemento da rede não depende da maior ou menor distância até algum ponto-chave (BOAVENTURA NETTO, 2001). Como exemplo têm-se as ligações elétricas em redes de pequena dimensão, redes telefônicas rurais e redes de microondas. Esse problema pode ser resolvido usando-se os algoritmos de Prim (ROY, 1969) e de Kruskal (KRUSKAL, 1956).

3.5.9 Estabilidade interna

Segundo Boaventura Netto (2001), um subconjunto S ? V em um grafo G = (V, A) é dito internamente estável (SCIE) ou independente se, para todo par {i, j} ? S se tem (i, j) ? A. E um SCIE S é maximal se não existir outro SCIE S' que verifique S' ? S. A teoria da estabilidade interna pode ser aplicada ao estudo de modelos cuja solução dependa da não-adjacência, ou mesmo da adjacência total, caso se passe ao complementar. Um exemplo disso – o problema de rotação de tripulações em uma companhia área – é discutido por Roy (1969). Para resolver esses tipos de problemas, podem ser usados o método de Maghout (KAUFMANN, 1968; IVANESCU; RUDEANU, 1968), o algoritmo de Read (WILSON; BEINEKE, 1979), o algoritmo de Demoucron e Hertz (ROY, 1969) e o algoritmo de Bron e Kersbosch (BRON; KERSBOSCH, 1973).

3.5.10 Estabilidade externa

Este conceito se refere à propriedade de determinados subconjuntos de vértices de um grafo, correspondente à existência de relações de adjacência com os todos os vértices externos a eles (BOAVENTURA NETTO, 2001). Esses subconjuntos (chamados de SCEE) serão minimais se não contiverem outro de menor cardinalidade. As aplicações mais comuns deste conceito se referem a questões de

vigilância, de controle ou de supervisão. O problema clássico envolvendo tal conceito é o chamado problema dos radares, que corresponde à localização de um certo número de pontos de vigilância que devem cobrir uma área determinada. Tendo-se um grafo que represente essa situação, cada SCEE minimal desse grafo é uma solução para o problema. Para determinar os SCEE minimais de um grafo pode-se usar o método de Maghout (KAUFMANN, 1968; IVANESCU; RUDEANU, 1968).

A estabilidade externa, assim como a estabilidade interna, são conceitos relacionados a problemas de subconjuntos de vértices. Tais conceitos têm relação com problemas de subgrupos em redes sociais (HANNEMAN, 2000; SCOTT, 2000).

3.5.11 Fluxos em grafos

Referem-se a um grafo valorado sobre as ligações, e tal valoração indicará quanto de um determinado recurso está sendo transferido através de cada ligação do grafo, possivelmente com base em alguma escala de tempo (BOAVENTURA NETTO, 2001). Esse recurso relaciona-se normalmente a aplicações em transportes, comunicações ou administração. Em transportes, é habitual que se fale em veículos, unidades de massa ou de volume; em comunicações, pode-se pensar em número de ligações telefônicas; e em administração, pode-se lidar com fluxos financeiros e/ou de documentos. Quanto aos objetivos desse conceito, pode-se desejar obter a maximização do fluxo do grafo que representa o problema e/ou minimizar o custo associado. Para resolver o problema do fluxo máximo pode-se utilizar o algoritmo de Ford e Fulkerson (FORD; FULKERSON, 1962), o algoritmo de Dinic (SYSLO; DEO; KOWALIK, 1983) e (ROSEAUX, 1991) e o algoritmo de DMKM (SYSLO; DEO; KOWALIK, 1983). Quanto ao problema do fluxo de custo mínimo pode-se utilizar o algoritmo de Roy, Busacker e Gowen (SYSLO; DEO; KOWALIK, 1983; ROSEAUX, 1991), o algoritmo de Bennington (ROSEAUX, 1991) e o algoritmo “out-of-kilter” (FULKERSON, 1961).

3.5.12 Percursos abrangentes

Os percursos abrangentes são aqueles que utilizam todas as ligações ou todos os vértices de um grafo. Os tipos específicos de percursos abrangentes mais interessantes são os que utilizam uma única vez uma ligação ou um vértice, os percursos eulerianos e hamiltonianos, respectivamente. Os problemas envolvendo os percursos eulerianos são denominados de forma genérica como Problemas do Carteiro Chinês (GOODMAN; HEDETNIEMI, 1973). As aplicações desse conceito se referem a problemas de atendimento seqüencial a um conjunto de usuários de um serviço oferecido no interior de uma malha urbana, tais como entrega de correio e coleta de lixo. A busca por percursos eulerianos em um grafo pode ser resolvida através do algoritmo de Fleury (BERGE, 1973). Já os problemas hamiltonianos são denominados genericamente como Problemas do Caixeiro-Viajante e envolvem questões de otimização, como, por exemplo, o percurso de perfuratrizes automáticas em trabalhos pré-programados (BOAVENTURA NETTO, 2001). Para resolver esses tipos de problemas pode-se usar a técnica apresentada por Behzad, Chartrand e Lesniak-Foster (1979).

O Quadro 3.1 mostra de forma resumida os conceitos, os métodos e as áreas de aplicação da Teoria dos Grafos apresentados anteriormente.

Conceito	Método	Áreas de aplicação
Distância	Algoritmos de caminho mais curto (GONDRAN; MINOUX, 1985; BERGE, 1985; WEST, 1996; BOAVENTURA NETTO, 2001; XU, 2000).	Redes sociais e transportes
Centros, medianas e anticentros	Calculados através da matriz de caminhos mínimos (BOAVENTURA NETTO, 2001)	Redes sociais e localização de recursos
Densidade	Através de uma fórmula simples (SCOTT, 2000)	Redes sociais
Fecho transitivo	Algoritmo apresentado por Kaufmann (1968) e Roy (1969)	Redes sociais e comunicação
Ponto de articulação	Algoritmo de Ford e Fulkerson (LOPES, 1980)	Redes sociais
Problema do labirinto	Algoritmo de Trémaux (BERGE, 1973)	Problemas que envolvem apenas decisões baseadas em propriedades locais
Caminho de valor máximo	Algoritmo de PERT (GONDRAN; MINOUX, 1985)	Seqüenciação de atividades em algum tipo de trabalho que possa ser dividido em etapas
Árvore parcial mínima	Algoritmo de Prim (ROY, 1969) e algoritmo de Kruskal (KRUSKAL, 1956)	Ligações elétricas, redes telefônicas rurais; e redes de microondas

Estabilidade interna	Método de Maghout (KAUFMANN, 1968; IVANESCU; RUDEANU, 1968), o algoritmo de Read (WILSON; BEINEKE, 1979), o algoritmo de Demoucron e Hertz (ROY, 1969) e o algoritmo de Bron e Kersbosch (BRON; KERSBOSCH, 1973)	Redes sociais e problemas de rotação de tripulações em uma companhia área
Estabilidade externa	Método de Maghout (KAUFMANN, 1968; IVANESCU; RUDEANU, 1968)	Redes sociais e problemas de vigilância, de controle ou de supervisão
Fluxos em grafos	Algoritmo de Ford e Fulkerson (FORD; FULKERSON, 1962); o algoritmo de Dinic (SYSLO; DEO; KOWALIK, 1983; ROSEAUX, 1991); o algoritmo DMKM (SYSLO; DEO; KOWALIK, 1983), algoritmo de Roy, Busacker e Gowen (STSLO et al., 1983; ROSEAUX ,1991); o algoritmo de Bennington (ROSEAUX, 1991) e o algoritmo "out-of-kilter" (FULKERSON, 1961)	Transportes, comunicações e administração
Percursos abrangentes	Algoritmo de Fleury (BERGE, 1973) e a técnica apresentada por Behzad, Chartrand e Lesniak-Foster (1979)	Problemas de atendimento seqüencial a um conjunto de usuários de um serviço e problemas de otimização

Quadro 3.1 - Resumo dos conceitos, métodos e áreas de aplicação da Teoria dos Grafos

3.6 Considerações finais

Este capítulo apresentou definições sobre grafos bem como algumas noções básicas e formas de representá-los. Também abordaram-se alguns dos principais conceitos da Teoria dos Grafos, seus métodos de resolução e algumas possíveis áreas de aplicação. O próximo capítulo apresenta o método proposto nesta dissertação.

4 MÉTODO PROPOSTO

4.1 Introdução

Este capítulo descreve o método proposto neste trabalho. Trata-se de um método para extração de informações que viabilizem a análise de relacionamentos por meio de Teoria dos Grafos e *Link Analysis.* Na próxima seção tem-se uma descrição geral sobre o método. Na seqüência, apresentam-se as cinco fases que o compõem, seguidas pelas considerações finais.

4.2 Descrição do método

A Figura 4.1 apresenta o método proposto para a transferência de informações para o domínio de análise de redes de relacionamentos. No primeiro passo do método está a representação da FASE I, ou seja, o estabelecimento das diretrizes com que serão traduzidas as fontes de informação em uma ontologia voltada à descrição de redes de relacionamentos de informações. Para tal, deve haver um analista do domínio do problema, isto é, uma pessoa capaz de descrever os tipos de relações que gostaria de ver analisadas para o domínio do problema para o qual se dispõe de informação.

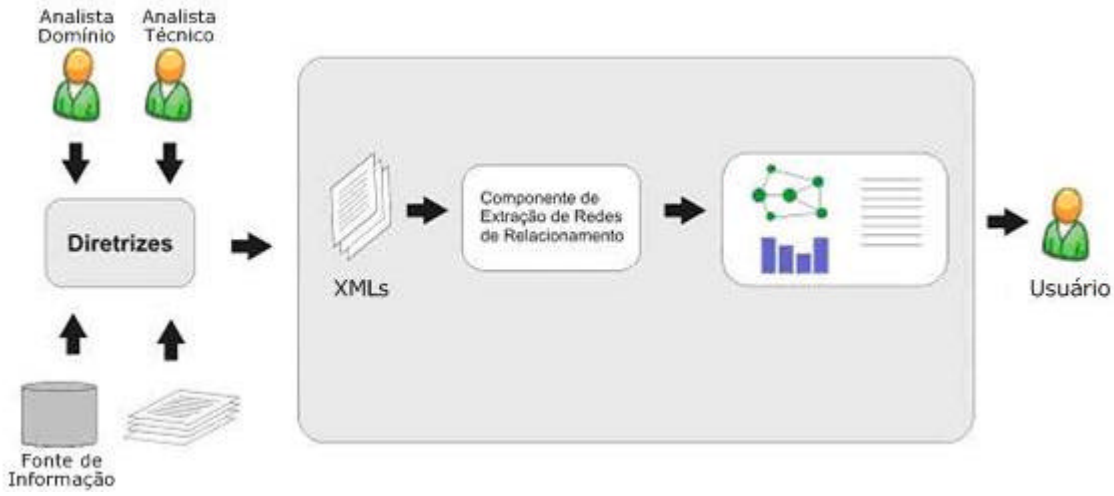

Figura 4.1 - Método proposto

Além do analista de domínio, é necessário que o analista técnico na estrutura da fonte de informação possa identificar quais elementos, estados de registros e regras devem ser aplicados para que se possa efetuar a tradução das fontes na ontologia de relacionamentos. O analista técnico deve também analisar a fonte de informação para verificar se há necessidade de aplicação de alguma técnica de *Link Analysis* para a busca de relacionamentos entre as unidades de informação apresentadas pelo analista de domínio.

A geração do arquivo que descreve a rede de relacionamentos, por sua vez, constitui a FASE II do método proposto. O arquivo XML que descreve os relacionamentos é composto de elementos e atributos que representam as ligações entre as unidades envolvidas no domínio. É a partir desse arquivo que se inicia a aplicação da FASE III, ou seja, a extensão do arquivo XML para possibilitar aplicações de visualização de relacionamentos existentes. Essa extensão pode, por exemplo, permitir que o sistema de conhecimento apresente, a partir de arestas do grafo, o detalhamento de informações já existentes nos sistemas que tratam das fontes de informação originais (ex.: uma aresta com o total de compras efetuadas em uma determinada região do País pode conter um link para a relação dessas compras). Outra extensão possível é a que permite projetar o vértice com determinado ícone.

A FASE IV consiste na aplicação de algoritmos de grafos ou de *Link Analysis* para geração de conhecimento sobre os relacionamentos mapeados. Essa geração pode ser realizada a partir de pacotes de *Link Analysis* ou Grafos, desde que se traduza o arquivo gerado na FASE II para os formatos esperados pelos pacotes. Uma alternativa é o desenvolvimento de componentes que sejam compatíveis com o formato estabelecido na FASE II e, ainda, compatíveis com o formato de visualização descrito na FASE III. Na FASE V são construídos os sistemas de visualização de redes que utilizarão como entrada os arquivos gerados na FASE III.

FASE I – DIRETRIZES PARA TRADUÇÃO DAS FONTES DE INFORMAÇÃO NA ONTOLOGIA DE DESCRIÇÃO DE REDES DE RELACIONAMENTOS

Depois de escolhidas as fontes de informações para as análises de relacionamento, deve-se identificar suas unidades de informação. Para isso, o primeiro passo é definir o escopo do projeto juntamente com o analista de domínio. É nesse momento que se verifica o propósito da análise e a disponibilidade das informações. Definem-se quais serão os elementos que farão parte das análises que o sistema irá realizar, o que implica em determinar as unidades de informação disponíveis e desejáveis. Deve-se efetuar a análise no âmbito de domínio da fonte de informação, como, por exemplo, saúde, financiamento e transporte. Nessa etapa devem-se identificar as principais unidades de informação do domínio e/ou aquelas que estão relacionadas com os objetivos do projeto. É importante entrevistar os usuários do sistema, os especialistas no domínio, os gestores, etc.

Para todos os tipos de fontes de informação, o analista no domínio do problema deve explicar ao analista técnico na fonte de informação o tipo de análise relacional que gostaria de efetivar. Essa explicação inclui a definição dos elementos das relações envolvidas. Por exemplo, um analista de domínio que trabalhe com análise de fornecedores pode explicar para o analista técnico na fonte de informação que gostaria de saber qual tem sido o fluxo de fornecimento e de vendas de uma empresa, dado que entre seus fornecedores estão também clientes.

O analista técnico pode avaliar duas bases de dados relacionais: (1) uma de compras, com códigos de fornecedor, e (2) outra de vendas, com códigos de empresas-cliente. O grafo de compras/vendas solicitado pelo analista de domínio poderá ser resultante da seleção de informações sobre volume financeiro de compras (retirado da base de dados de compras), volume financeiro de vendas (retirado da base de vendas), ambos como relações de diferentes direções entre a empresa e os seus fornecedores/clientes.

Para o analista de domínio as relações identificarão relações comerciais de equilíbrio, de vantagens para a empresa (quando as vendas são maiores que a compra) ou de desvantagens (no caso inverso), o que pode implicar novas políticas

comerciais (ex.: permutas). Para o analista técnico de bancos de dados trata-se apenas de registros inseridos em bases de dados distintas, porém conectáveis.

Portanto, para que as análises de redes de relacionamentos possam ser extraídas de fontes de informação, o analista do domínio deve descrever as entidades envolvidas nas relações, os significados e os sentidos dos relacionamentos. Ao analista técnico na fonte de informação cabe o trabalho de identificar quais tabelas devem ser acessadas, com que estado de valor e com que regra de geração para produzir um link na forma solicitada pelo analista no domínio. Ele deve também analisar a fonte de informação para verificar se há necessidade de aplicação de alguma técnica de *Link Analysis* para a busca de relacionamentos entre as unidades de informação apresentadas pelo analista de domínio. Por exemplo, pode haver a necessidade de se relacionarem duas unidades de informação que são representadas por tabelas em bases de dados distintas sem que haja uma chave para relacioná-las. Ferramentas e/ou técnicas de *Link Analysis* podem também ajudar a encontrar unidades de informação que possuem um tipo de relacionamento que não era previamente conhecido pelos analistas responsáveis pelo projeto.

Esse trabalho é distinto para cada tipo de representação de informação. Para bancos de dados relacionais as tabelas do tipo “entidade” e do tipo “relacionamento” já podem dar pistas de possíveis ligações analisáveis sob a abordagem de redes. Por outro lado, entidades não relacionadas diretamente também podem, quando combinadas, produzir novos domínios de análise. Em bancos do tipo *data warehouse*, o próprio modelo de combinação de domínios operacionais, para efeitos de relatórios e indicadores dinâmicos, permite estabelecer um novo conjunto de análises na forma de redes de relacionamentos, por exemplo, pela verificação do esquema adotado na modelagem e por sua comparação com os metadados do problema. Em bases compostas de arquivos XML a relação de entidades pode ser evidenciada pela aplicação, por exemplo, de regras de combinação de elementos (ex.: programação XSLT[5]). A seguir, discutem-se essas situações.

[5] *Extensible Stylesheet Language Transformation*

- **Diretrizes para banco de dados relacionais**

  - Identificar as tabelas do tipo “entidade” e “relacionamento”. Para isso, pode-se analisar as chaves primárias. Normalmente tabelas do tipo “entidade” possuem somente um atributo na chave primária, e as tabelas de relacionamento possuem mais de um atributo nessa chave. Isso pode ser visto no exemplo apresentado na Figura 4.1, que representa um modelo Entidade-Relacionamento. Nesse modelo, as duas entidades, EN_PESSOA e EN_IDIOMA, possuem apenas um atributo cada como chave primária, que são ID_PESSOA e ID_IDIOMA, respectivamente. Já o relacionamento entre as duas entidades (tabela RE_IDIOMA_PESSOA) possui dois atributos na chave primária, ID_PESSOA e ID_IDIOMA.

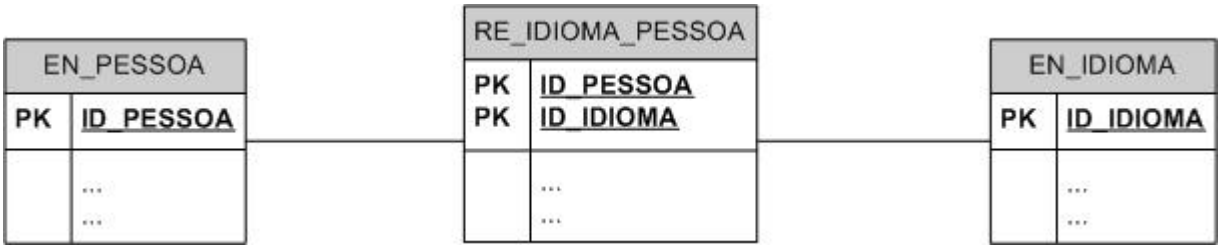

Figura 4.2 - Duas entidades e um relacionamento

  - Comparar as entidades relacionadas com os elementos do domínio do problema para os quais se deseja realizar as análises de rede, identificando aqueles mapeados em tabelas do tipo “entidade” e os que eventualmente estão em tabelas de relacionamento. Nesse caso, o analista técnico na fonte de informação deve procurar combinar essas tabelas (através de chaves primárias e estrangeiras) na tentativa de buscar unidades de informação e relacionamento desejados pelo analista de domínio, ou mesmo para descobrir unidades e elucidar relacionamentos não previstos pelo analista de domínio. Um relacionamento entre entidades, representado pela tabela RE_IDIOMA_PESSOA, pode ser visto no modelo apresentado na Figura 4.2

  - Verificar se o valor de atributos das tabelas devem estar em determinados estados para identificarem elementos da análise de redes que se deseja efetuar. Por exemplo, para que uma pessoa entre na análise de egressos, o

atributo “ano de conclusão” do curso deve estar preenchido. Isso pode ser visto no na Figura 4.3.

EN_FORMACAO

ID_FORMACAO	ID_PESSOA	.....	ANO_CONCLUSAO	.....
001	921	.....	2000	.....
002	921	.....		.....
003	922	.....	1990	.....
004	923	.....	1994	.....
005	924	.....		.....
.....	.....	.....	.....	.....

Selecionar registros com o campo ANO_CONCLUSAO preenchido.

ID_FORMACAO	ID_PESSOA	.....	ANO_CONCLUSAO	.....
001	921	.....	2000	.....
003	922	.....	1990	.....
004	923	.....	1994	.....
.....	.....	.....	.....	.....

Figura 4.3 - Seleção de registros com o atributo ANO_CONCLUSAO preenchido.

- Buscar também unidades de informação em campos de uma tabela. Algumas unidades de informação descritas pelo analista de domínio podem ser representadas por um atributo, e não necessariamente pela entidade em seu conjunto. Por exemplo, o analista de domínio pode querer apresentar um grafo no qual alguns dos vértices sejam definidos pelo sexo da pessoa. Nesse caso, essa unidade de informação poderia ser gerada a partir do campo SEXO de uma tabela EN_PESSOA, como pode ser visto na Figura 4.4.

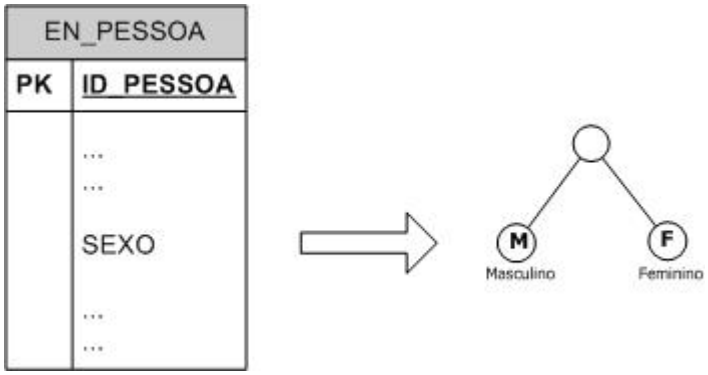

Figura 4.4 - Atributo Sexo sendo usado como unidade de informação no grafo

- Averiguar se há a necessidade de se criarem classes para que determinados campos representem as unidades de informação solicitadas pelo analista de domínio. Por exemplo, o analista de domínio poder desejar formar uma rede na qual um tipo de vértice será “período de formação”, em que cada vértice corresponde a um período de cinco anos. Na tabela que representa as formações de uma pessoa, poderia haver um atributo “ano de formação”. Portanto, os valores desse atributo devem ser classificados em períodos de cinco anos para atender à necessidade do analista de domínio.
- Analisar os metadados e a documentação do banco de dados com o intuito de melhor compreender a semântica das tabelas e seus relacionamentos.

- **Diretrizes para Data Warehouse**

- Identificar as dimensões do modelo de dados. Normalmente esses tipos de tabelas representam entidades importantes no domínio do problema.
- Verificar cada um dos campos das dimensões identificadas. Muitos desses campos possuem informações derivadas que podem ser interessantes para análise. Por exemplo, uma dimensão "DIM_PESSOA" poderia ter um campo “TITULACAO_MAXIMA” com a descrição do maior nível de formação acadêmica da pessoa, como mostra a Figura 4.5. Esse campo pode facilmente ser usado como unidade de informação em uma rede. Mas se a fonte de informação utilizada fosse uma base relacional, provavelmente não existiria esse campo, e sim uma tabela com várias formações. Nesse caso,

seria necessário identificar qual formação representa a titulação máxima do indivíduo.

DIM_PESSOA

ID_PESSOA	NOME_PESSOA	.....	TITULACAO_MAXIMA	.....
921	Joaquim Neves	.....	Doutorado	.....
922	João de Souza	.....	Mestrado	.....
923	José da Silva	.....	Doutorado	.....
.....	.....	.....	.....	.....

Figura 4.5 - Dimensão Pessoa com o atributo “TITULACAO_MAXIMA”

- Verificar os campos das tabelas de fato. Isso porque os campos dessas tabelas normalmente possuem informações sumarizadas e prontas para serem analisadas, portanto, podem ser unidades de informação interessantes
- Analisar os metadados e a documentação do banco de dados com o intuito de melhor compreender a semântica das dimensões e tabelas de fato bem como dos campos previamente identificados.

- **Diretrizes para bases compostas de arquivos XML**

- Procurar primeiramente as entidades presentes nos elementos de primeiro nível, pois, provavelmente, os conteúdos mais distantes da raiz da árvore serão menos significativos do que aqueles que estão próximos à raiz
- Para cada um dos elementos identificados, verificar se os seus atributos podem representar unidades de informação interessantes ao analista de domínio.
- Quando houver dificuldade de encontrar no arquivo XML os elementos relacionados aos conceitos apresentados pelo analista de domínio, utilizar algum sistema de recuperação de informação para documentos XML. Com

sistemas desse tipo, o analista técnico pode fazer consultas usando palavras-chave informadas pelo analista de domínio. Como exemplos desses sistemas têm-se o *XYZFind* (EGNOR, 2000), o *Xyleme* (AGUILERA, 2000) e o XMLFS (AZAGURY, 2000).

- ✍ Analisar os metadados e a documentação do projeto para compreender a semântica dos seus elementos e atributos.

FASE II – GERAÇÃO DE UM ARQUIVO XML QUE DESCREVE A REDE DE RELACIONAMENTOS

Depois de estabelecidas as unidades de informação a serem utilizadas bem como seus relacionamentos e as regras de busca na fonte de informação, o próximo passo é implementar o sistema que fará a busca dos dados. Quando a fonte de informação for do tipo base relacional ou *data warehouse*, uma sugestão é o uso de *flat files*[6] combinados com XSLT para a geração do arquivo XML no formato proposto. Esse procedimento está esquematizado na Figura 4.6.

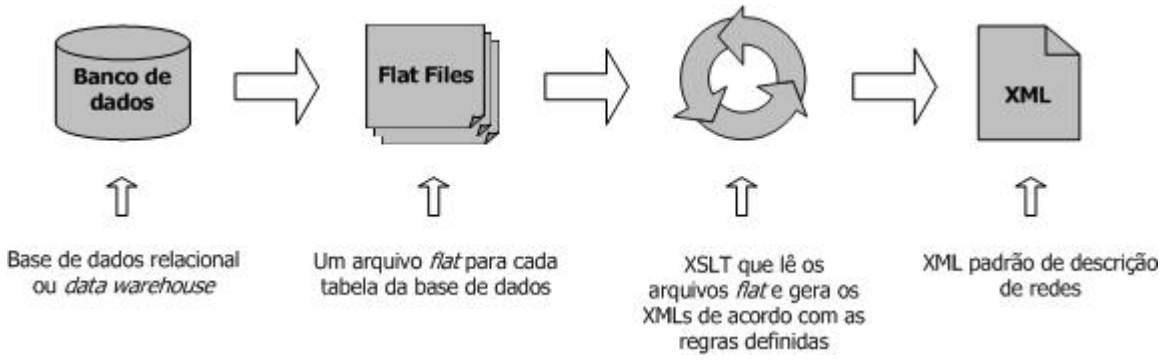

Figura 4.6 - Geração de XML no formato proposto a partir do Banco de Dados

O primeiro passo consiste em implementar um módulo que faz as consultas ao BD e cria um *flat file* para cada tabela da base necessária ao sistema. Esses arquivos são apenas "espelhos" das tabelas. A seguir, implementa-se um programa em XSLT que lê tais arquivos e gera o arquivo XML de descrição de redes de

[6] Arquivos com texto separado por vírgulas, tabulação ou outro formato similar.

relacionamento, de acordo com as regras estabelecidas na FASE I. A vantagem de se usar esse procedimento é que ele reduz a dependência de plataformas. Isso se deve ao fato de que, caso fosse necessário desenvolver esse sistema em outra plataforma ou simplesmente em outra linguagem, bastaria implementar novamente o módulo que gera os *flat files*. Esse módulo apenas executa consultas simples em SQL (*Structured Query Language*) sobre as tabelas envolvidas no sistema. As regras definidas na FASE I, as quais podem ser complexas, estão presentes no XSLT, que, por ser portável, não necessita ser implementado novamente.

Caso a fonte de informação seja do tipo XML, a sugestão é que se utilize apenas programação XSLT. Esse programa XSLT, utilizando as regras definidas na FASE I, lê o(s) arquivo(s) com os dados e, então, gera o arquivo XML no formato padrão proposto.

Para qualquer um dos três tipos de fontes de informação, a solução mais simples é implementar todas as consultas e regras em uma linguagem de programação qualquer que gera o arquivo no formato padrão. Porém, essa solução é mais dependente de plataforma. A escolha de uma ou outra solução depende das particularidades de cada projeto.

Independentemente da fonte de informação ou do tipo de solução utilizada, o arquivo XML deve seguir a ontologia de descrição de redes de relacionamentos apresentada na Figura 4.7.

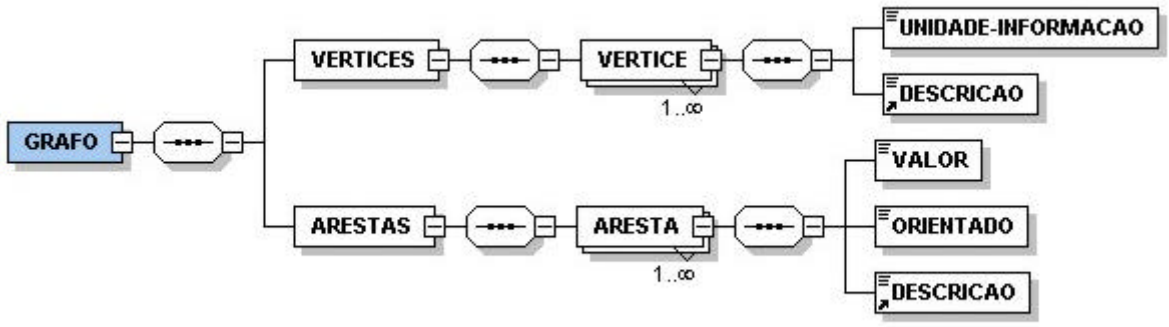

Figura 4.7 - Schema do modelo proposto

A Figura 4.7 apresenta de forma esquemática o XML Schema utilizado para validação do arquivo gerado nesta fase. Esse schema representa um grafo (ou

rede), que é composto de um conjunto de vértices e de um conjunto de arestas. O Quadro 4.1 apresenta uma descrição dos elementos contidos em cada um dos elementos vértice e aresta.

VERTICE	
ID	Identificador único do vértice
UNIDADE-INFORMACAO	Usado para indicar a qual tipo de unidade de informação o vértice pertence
DESCRICAO	Fornece uma descrição textual do item contido no vértice
ARESTA	
ID	Identificador único da aresta
ID-ORIGEM	Identifica o vértice de origem da aresta, somente é usado se o atributo ORIENTADO for igual a True
ID-DESTINO	Identifica o vértice de destino da aresta, somente é usado se o atributo ORIENTADO for igual a True
VALOR	Indica um número que será o valor do relacionamento
ORIENTADO	Diz se a ligação é orientada ou não
DESCRICAO	Fornece uma descrição textual sobre a aresta

Quadro 4.1 - Descrição dos elementos de um vértice e de uma aresta

Assim, o componente de extração de redes de relacionamento utiliza esse XML Schema para validar o arquivo de entrada. Um exemplo desse arquivo é apresentado na Figura 48.

```
<grafo>
  <vertices>
      <vertice id="1">
        <unidade-informacao>Cidade</unidade-informacao>
        <descricao>Florianópolis</descricao>
      </vertice >
      <vertice id="2">
        <unidade-informacao>Cidade</unidade-informacao>
        <descricao>Curitiba</ descricao>
      </vertice >
  </vertices>
  <arestas>
      <aresta id="1" id-origem="1" id-destino="2">
        <valor>300</valor>
        <orientado>false</orientado>
        <descricao>Distância em Km</descricao >
      </aresta>
  </arestas>
</grafo>
```

Figura 4.8 - Exemplo do XML segundo a ontologia de descrição de redes

Esse arquivo representa um grafo com dois vértices e uma aresta. Um vértice representa a cidade de Florianópolis, e o outro representa Curitiba. A aresta entre esses dois vértices representa a distância (elemento <valor>) entre as duas cidades. O elemento <orientado> igual a *false* diz que o grafo não é orientado. Isso se deve ao fato de que a distância de Florianópolis até Curitiba é a mesma de Curitiba até Florianópolis. Assim, esse arquivo poderia representar uma base de dados sobre um mapa, o qual conteria informações sobre cidades (vértices) e a distância (arestas) entre elas.

FASE III – EXTENSÃO DO ARQUIVO XML PARA APLICAÇÕES DE VISUALIZAÇÃO DOS RELACIONAMENTOS EXISTENTES

A partir do arquivo gerado na FASE II se inicia a aplicação da FASE III, ou seja, a extensão do arquivo XML para aplicações de visualização de relacionamentos. Essa extensão pode, por exemplo, permitir que o sistema de conhecimento retorne, a partir de cliques nas arestas do grafo, detalhes de informações já existentes nos sistemas que tratam das fontes de informação originais (ex: o usuário pode partir da aresta com o volume de compras efetuadas em uma determinada região do País e ir para a relação dessas compras).

Outra extensão possível é a que permite projetar o vértice com determinado ícone. Esse arquivo estendido deve seguir a ontologia apresentada na Figura 49, que apresenta de forma esquemática o XML Schema utilizado para validação do arquivo que permite a visualização dos relacionamentos. Esse schema representa um grafo que é composto de um conjunto de vértices e de um conjunto de arestas. A diferença desse arquivo para o da fase anterior está nas informações adicionais contidas nos vértices e nas arestas que são utilizadas para a visualização.

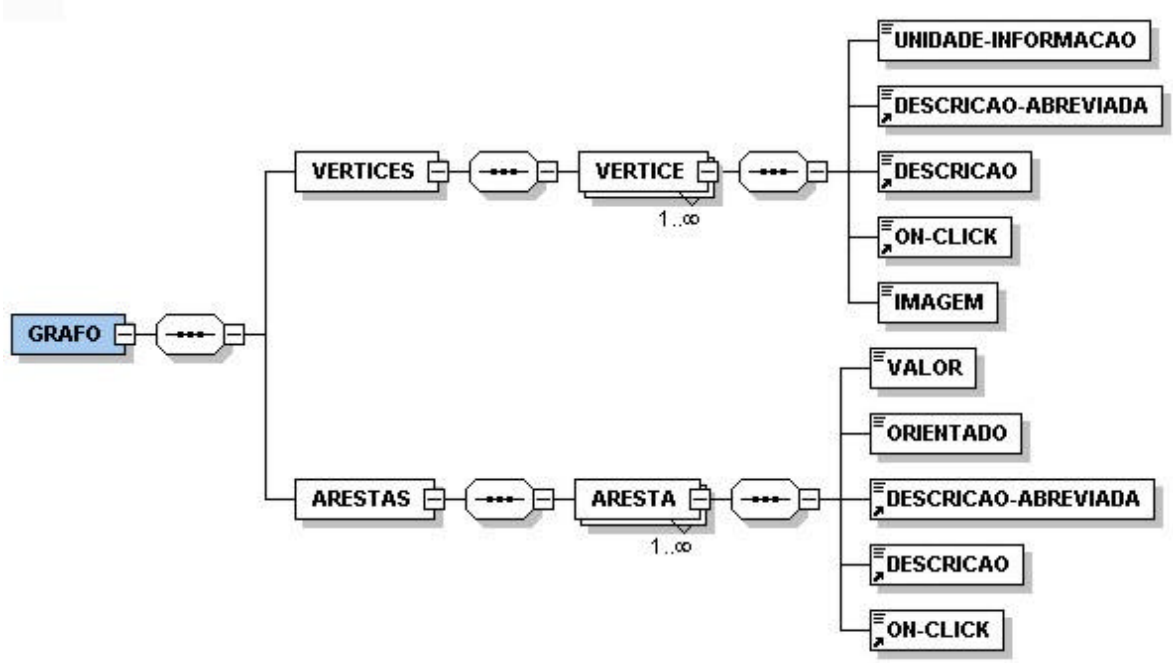

Figura 4.9 - Schema estendido do modelo proposto

A descrição desses elementos pode ser vista no Quadro 4.2.

VERTICE	
ID	Identificador único do vértice
UNIDADE-INFORMACAO	Usado para indicar a qual tipo de unidade de informação o vértice pertence
DESCRICAO-ABREVIADA	Fornece uma descrição textual abreviada do item contido no vértice
DESCRICAO	Fornece uma descrição textual do item contido no vértice
ON-CLICK	Indica para o componente de visualização que evento disparar quando o usuário clicar no vértice
IMAGEM	Indica para o componente de visualização que imagem utilizar para representar o vértice
ARESTA	
ID	Identificador único da aresta
ID-ORIGEM	Identifica o vértice de origem da aresta, somente é usado se o atributo ORIENTADO for igual a True
ID-DESTINO	Identifica o vértice de destino da aresta, somente é usado se o atributo ORIENTADO for igual a True
VALOR	Indica um número que será o valor do relacionamento
ORIENTADO	Diz se a ligação é orientada ou não
DESCRICAO-ABREVIADA	Fornece uma descrição textual abreviada sobre a aresta
DESCRICAO	Fornece uma descrição textual sobre a aresta.
ON-CLICK	Diz qual evento deverá ser disparado quando o usuário clicar na aresta

Quadro 4.2 - Descrição dos elementos de um vértice e de uma aresta do arquivo estendido

O elemento <imagem> de um vértice pode indicar o nome e a localização do arquivo que identifica o ícone usado para representar o significado da unidade de informação contida em tal vértice. Por exemplo, se um determinado vértice representa uma instituição, o logotipo dessa instituição pode ser usado como imagem no vértice. Além disso, o elemento <imagem> também pode ser um parâmetro que indica qual figura geométrica usar para representar o vértice entre as opções disponíveis no componente de visualização.

As descrições textuais dos vértices e das arestas podem ser textos simples que descrevem o seu significado. Essa descrição também pode ser gerada através de técnicas de geração automática de textos a partir de banco de dados (MARTINS et al., 2004).

Um exemplo de arquivo XML estendido que está de acordo com a ontologia proposta é apresentado na Figura 410.

```
<grafo>
  <vertices>
     <vertice id="1">
        <unidade-informacao>Cidade</unidade-informacao>
        <descricao-abreviada>FLN</descricao-abreviada>
        <descricao>Florianópolis</descricao>
        <on-click>enviarVertice('1')</on-click>
        <imagem>circulo</imagem>
     </vertice>
     <vertice id="2">
        <unidade-informacao>Cidade</unidade-informacao>
        <descricao-abreviada>CTB</descricao-abreviada>
        <descricao>Curitiba</ descricao>
        <on-click>enviarVertice('2')</on-click>
        <imagem>circulo</imagem>
     </vertice>
  </vertices>
  <arestas>
     <aresta id="1" id-origem="1" id-destino="2">
         <valor>300</valor>
         <orientado>false</orientado>
         <descricao-abreviada>Distância em Km</descricao-abreviada>
         <descricao>Distância em Km</descricao >
         <on-click>enviarAresta('1')</on-click>
     </aresta>
  </arestas>
</grafo>
```

Figura 4.10 - Exemplo de extensão do XML proposto no método

Além das informações contidas no arquivo gerado na FASE II, esse arquivo contém informações adicionais, a saber:

- uma descrição abreviada sobre cada vértice que será usada para gerar um rótulo para vértice;
- um nome de função para cada vértice, função esta que deve ser chamada quando o usuário clicar sobre o vértice;
- um parâmetro indicando ao componente que figura geométrica usar para representar um vértice;
- uma descrição abreviada sobre cada aresta que será mostrada quando o usuário apontar o mouse sobre a aresta;e
- um nome de função para cada aresta, que deve ser chamada quando o usuário clicar sobre a aresta.

FASE IV – APLICAÇÃO DE ALGORITMOS DE TEORIA DOS GRAFOS E LINK ANALYSIS

Esta etapa consiste na aplicação de algoritmos de grafos ou de *Link Analysis* para geração de conhecimento sobre os relacionamentos mapeados. Essa geração pode ser realizada a partir de pacotes de *Link Analysis* ou grafos, desde que se traduza o arquivo gerado na FASE II para os formatos esperados pelos pacotes. Uma alternativa é o desenvolvimento de componentes nessas metodologias, já compatíveis com o formato estabelecido na FASE II e, ainda, compatíveis com o formato de visualização descrito na FASE III. Com relação a *Link Analysis*, deve-se considerar as questões propostas por Lyons (1998), as quais foram mostradas no capítulo 2 desta dissertação. Na aplicação de métodos da Teoria dos Grafos, deve-se analisar o Quadro 3.1 para identificar quais métodos podem ser interessantes para serem aplicados no grafo em estudo.

Agora voltemos ao exemplo da base de dados sobre um mapa que tem informações sobre cidades (vértices) bem como a distância (arestas) entre elas. Tendo-se um grafo que represente essa base, um usuário pode querer saber a distância entre duas cidades que não possuem uma ligação direta entra elas. Assim, um algoritmo de caminho mínimo pode ser utilizado para resolver o problema e, então, apresentar essa informação nova ao usuário.

FASE V – VISUALIZAÇÃO DOS RELACIONAMENTOS EXISTENTES A PARTIR DO ARQUIVO XML OBTIDO NA FASE III

Nesta fase são desenvolvidos os sistemas de visualização dos relacionamentos. Esses sistemas devem utilizar o arquivo XML estendido que foi obtido na FASE III, no qual estão presentes as informações necessárias para a visualização. Assim, utilizando-se o exemplo de arquivo de descrição de redes estendido apresentado na Figura 4.10, uma possível visualização é ilustrada na Figura 4.11.

Figura 4.11 - Exemplo de visualização da rede

Existem várias maneiras de se construírem aplicações para visualização das redes. O responsável pelo sistema deve identificar a técnica, o componente ou o sistema de visualização que melhor satisfaça as necessidades do projeto. A única exigência é que a tecnologia escolhida ou desenvolvida consiga utilizar como parâmetro de entrada o arquivo XML produzido na FASE III.

Uma opção para se construir o mecanismo de visualização é implementar um componente, em uma linguagem de programação qualquer, que receba esse arquivo como entrada, faça sua validação usando o XML Schema e gere uma imagem, como, por exemplo, é ilustrado na Figura 4.11.

Outra opção é usar o SVG (*Scalable Vector Graphics*) combinado com XSLT. O SVG é uma linguagem para descrever objetos bidimensionais em XML (SVG, 2004). O SVG permite três tipos de objetos gráficos: linhas, imagens e texto. Além disso, possui um rico conjunto de manipuladores de eventos. Esses são os recursos necessários para apresentar a rede de forma gráfica. A Figura 4.12 mostra o uso de XSLT com SVG para a visualização da rede.

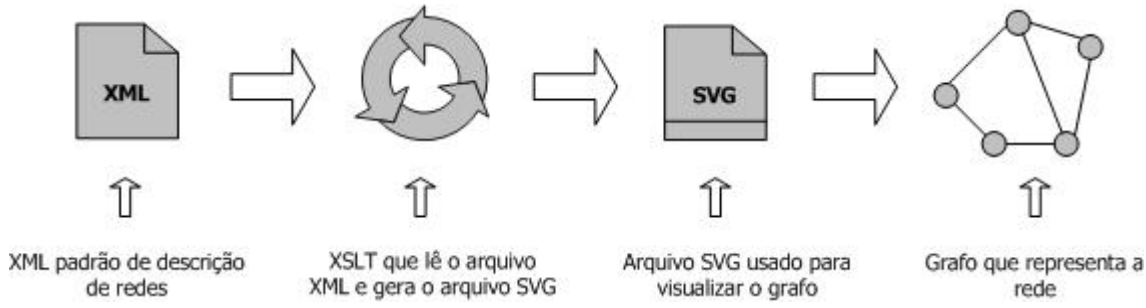

Figura 4.12 - Visualização da rede através de XSLT combinado com SVG

O arquivo XML com a descrição da rede (ver exemplo na Figura 4.10) é usado como entrada para um programa XSLT, o qual, por sua vez, valida o XML e gera o SVG de acordo com os dados do XML. Depois disso, o arquivo SVG pode ser usado para gerar automaticamente a imagem através de bibliotecas de software específicas para tal. Caso a imagem seja gerada na Web, os navegadores conseguem gerá-la através do uso de *plug-in*.

4.3 Considerações finais

Este capítulo mostrou o método proposto para a transferência de informações para o domínio de análise de redes de relacionamentos. Apresentaram-se também uma descrição geral do método e suas cinco fases, sendo que para cada fase foram relacionadas algumas diretrizes e sugeridas soluções de desenvolvimento. No próximo capítulo, apresentam-se duas aplicações desse método na Plataforma Lattes de CT&I.

5 APLICAÇÃO DO MÉTODO PROPOSTO

5.1 Introdução

Serão apresentados a seguir dois sistemas de conhecimento que foram desenvolvidos no âmbito de uma plataforma de governo eletrônico – a Plataforma Lattes[7] – segundo o método proposto neste trabalho. Então, primeiramente será mostrada a arquitetura de tal plataforma, indicando de que forma os dois sistemas estão presentes nessa arquitetura. Na seqüência, apresenta-se o primeiro sistema, o Lattes Egressos, e em seguida, o Lattes Colaboradores. Posteriormente, são descritas as considerações finais.

5.2 Plataforma Lattes

A Plataforma Lattes (PL) é um projeto do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) desenvolvida para apoiar as atividades de Ciência, Tecnologia e Inovação (CT&I) do Brasil. Foi inicialmente concebida para integrar os sistemas de informação das agências federais e promover a racionalização do processo de gestão de CT&I. Lançada em 1999, com a divulgação de seu primeiro componente (Sistema CV-Lattes), a PL é atualmente formada de bases de dados, sistemas de informação, diretórios de serviços e portais Web. Trata-se de um recurso que forneceu uma nova maneira de ver as informações em C&T no Brasil (SABBATINI, 2003). É utilizada por gestores, por técnicos de governo, pela comunidade científica e pela sociedade em geral.

Contudo, a PL não corresponde a um conjunto independente de sistemas. Sua construção foi realizada seguindo a metodologia integrada para desenvolvimento de plataformas de governo proposta por Pacheco (2003). Essa arquitetura, no caso da PL, é apresentada na Figura 5.1. Sua representação é em forma de pirâmide, sendo que a base é composta de unidades de informação da plataforma. Na camada seguinte representa-se a padronização, sistematização e publicação de informações

[7] http://lattes.cnpq.br

e serviços. E, por último, no topo estão os instrumentos de gestão, produção e publicação de conhecimento.

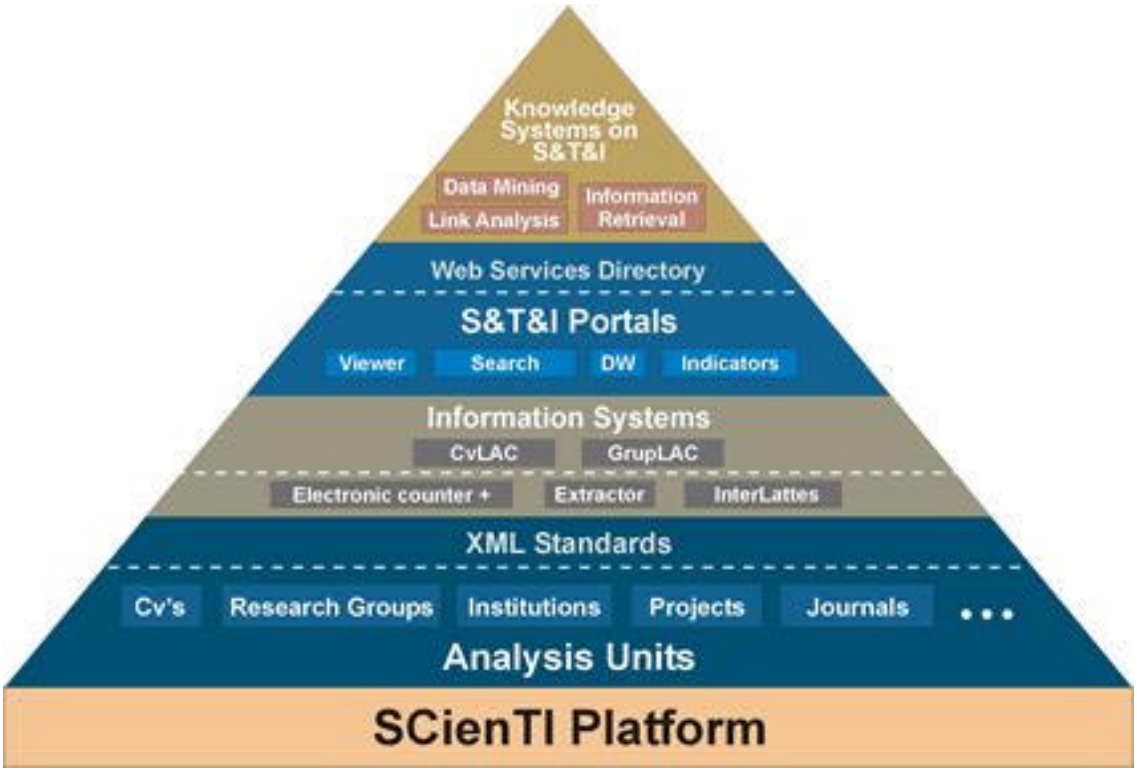

Figura 5.1 - Arquitetura conceitual da Plataforma Lattes
Fonte: Pacheco (2003)

5.2.1 Unidades de informação

Na primeira camada dessa arquitetura estão situadas as Unidades de Informação que descrevem os subdomínios da área-fim à qual se destina a plataforma. Tais unidades estão associadas a conteúdos, processos e serviços específicos. Segundo Pacheco (2003),

> [...] uma unidade de informação não pode ser genérica a ponto de não especificar conteúdo e processos independentes nem ser específica na descrição ou funcionamento (casos em que provavelmente seja um elemento da unidade de informação).

Ainda de acordo com Pacheco (2003), uma unidade possui identificação unívoca para cada registro da plataforma de informação e possui relação direta com outras

unidades de informação do domínio da plataforma. No caso da Plataforma Lattes, as unidades de informação são chamadas de “Unidades de Análise”, e as principais são descritas a seguir.

- **Currículo**. Essa unidade representa as informações que descrevem a identificação, o endereço, a atuação, a experiência profissional e a produção das pessoas relacionados a CT&I, tais como pesquisadores, estudantes, docentes, gestores de CT&I, técnicos de governo e administradores (PACHECO, 2003).
- **Grupos de Pesquisa**. Essa unidade descreve os grupos de pesquisa, que podem ser definidos como conjuntos de indivíduos organizados hierarquicamente, de acordo com hierarquia de experiência e liderança no terreno científico ou tecnológico em que atuam. Os grupos reúnem profissionais envolvidos permanentemente com atividades de pesquisa, cujo trabalho é organizado em torno de linhas comuns de pesquisa, compartilhando instalações e equipamentos (GUIMARÃES et al., 1999).
- **Projetos de Pesquisa**. Essa unidade contempla a descrição das atividades de pesquisa, desenvolvimento ou extensão, realizadas por um pesquisador ou por uma equipe de pesquisa, tendo como base um tema ou objeto específico, com objetivos, metodologia e duração definidos (*i.e.*, projetos de pesquisa) e os pedidos de auxílios enviados ao CNPq (PACHECO, 2003).
- **Instituições**. Essa unidade representa as informações de organizações, institutos, empresas, universidades e demais organismos ligados a CT&I, referenciados nas demais unidades como local de lotação profissional ou de pesquisa, ou representando atores institucionais que interagem com o CNPq (PACHECO, 2003).

A Figura 5.2 ilustra de forma esquemática a relação entre as unidades de análise da Plataforma Lattes (currículos e grupos de pesquisa), as diferentes formas de classificá-las e as suas principais comunidades usuárias.

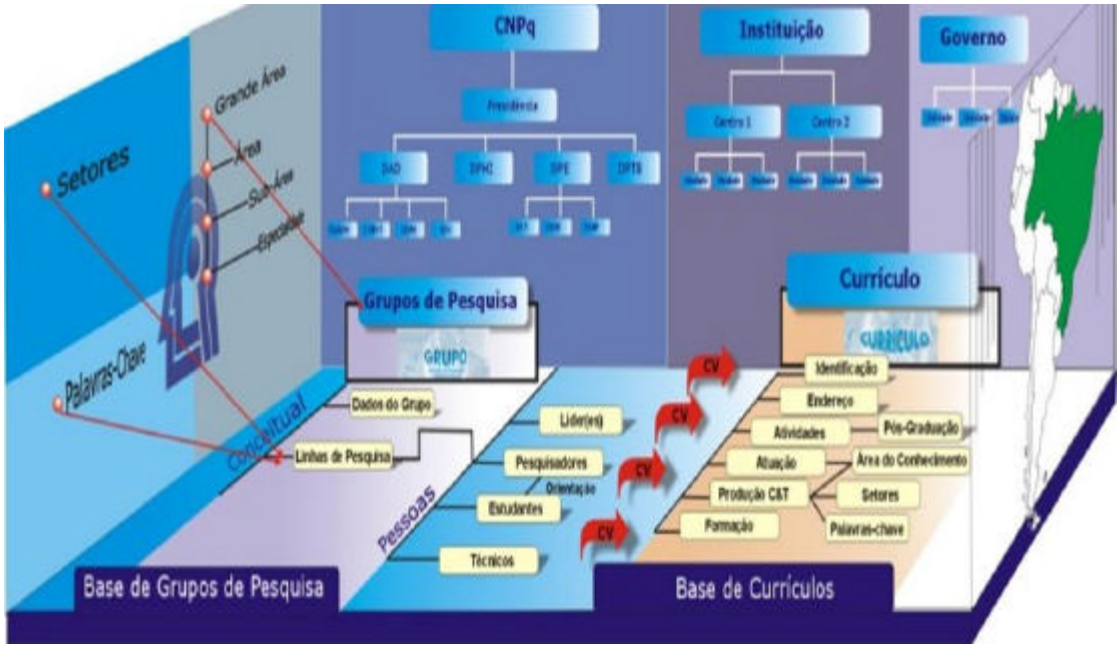

Figura 5.2 - Unidades de Análise da Plataforma Lattes
Fonte: STELA (2002)

Está prevista na camada de unidades de informação a padronização do conteúdo da cada unidade de análise. Para realizar essa padronização, é necessário definir a ontologia das unidades de informação e sua padronização XML (PACHECO, 2003).

As ontologias estabelecem compreensão compartilhada e comum de um domínio e podem ser trocadas entre pessoas e computadores (STUDER et al., 2000). Com a definição de ontologias obtêm-se gramáticas e vocabulários comuns a usuários interessados no domínio correspondente (PACHECO; KERN, 2001). O resultado é a uniformização de referências e conseqüente facilitação do processo de descoberta e geração de conhecimento (PACHECO, 2003). Portanto, a realização de um trabalho adequado de padronização facilita o desenvolvimento dos instrumentos propostos neste trabalho.

Segundo Pacheco (2003), "a tarefa de estabelecer ontologias para as unidades de informação da plataforma deve produzir como resultado padrões compartilháveis e intercambiáveis entre os interessados". Para isso, é necessária a produção explícita desses padrões, que se constitui na formação de metadados específicos para cada unidade de informação da plataforma. Assim, para realizar essa padronização, Pacheco (2003) sugere o uso da linguagem XML (*eXtensible Markup Language*), em virtude de ser o padrão de maior aceitação, de acordo com Khare e Rifkin (1998).

Na Plataforma Lattes, o trabalho de padronização das unidades de análise vem sendo realizado pela Comunidade para Ontologias em Ciência, Tecnologia e Informações de Aperfeiçoamento de Nível Superior (CONSCIENTIAS)[8]. A CONSCIENTIAS foi criada para desenvolver ontologias, em linguagem XML, que têm por finalidade principal o estabelecimento de uma forma comum de troca de informações entre agências de fomento e suas instituições usuárias.

A definição de padrões Lattes e, principalmente, sua disseminação e inclusão nos sistemas de informação, têm promovido pesquisas específicas por parte da comunidade científica nacional na proposição de ontologias para a Plataforma Lattes (BONIFÁCIO, 2002).

5.2.2 Sistemas e fontes de informação

A segunda camada da arquitetura corresponde aos sistemas e às fontes de informação de cada uma das unidades (PACHECO, 2003). Os sistemas de informação são responsáveis pela captura, pelo tratamento e pela armazenagem dos dados de cada unidade de informação da plataforma. As fontes de informação são repositórios de dados relativos a cada unidade. Tanto os sistemas como as fontes de informação devem seguir as padronizações definidas na base da pirâmide.

A Plataforma Lattes possui um conjunto de sistemas de informação *off-line* e *on-line* para captura, tratamento, armazenagem e recuperação das informações referentes às suas unidades de análise. Por exemplo, o CV-Lattes (unidade Currículo), Grupo (unidade Grupo), Propostas (unidade Projetos), entre outros.

Quanto às fontes de informação, a Plataforma Lattes possui bases operacionais e *warehouses* para suas unidades de análise. Por exemplo, SolicFomento (base relacional da unidade de análise Currículo), DMCurric (base *warehouse* da unidade Currículo), SIGEF (base relacional da unidade Projeto), DMFomento (base *warehouse* da unidade Projeto), entre outras.

[8] http://www.cnpq.br/lmpl

5.2.3 Portais e serviços Web

A terceira camada compreende os instrumentos de apresentação de informações na Web, tais como websites e portais. Também disponibiliza serviços de informação governamentais na Web através de *Web Services*.

Nessa camada, a Plataforma Lattes possui sistemas de busca, sites específicos a cada unidade de análise e sites temáticos para subdomínios de Ciência e Tecnologia. Por exemplo, o site da Plataforma Lattes (site principal da Plataforma), o Diretório de Grupos (site do projeto "Diretório de Grupos de Pesquisa noBrasil"), entre outros. Com relação aos *Web Services*, o seu uso possibilitou ao CNPq a integração da sua base de currículos com outras agências de fomento.

5.2.4 Sistemas de conhecimento

No topo da pirâmide estão os sistemas desenvolvidos para a gestão egeração do conhecimento a partir das fontes de informação da plataforma (PACHECO, 2003).

Os instrumentos projetados na camada de sistemas de conhecimento utilizam métodos de descoberta de conhecimento, como, por exemplo, *Link Analysis* e/ou Teoria dos Grafos, apresentados anteriormente neste trabalho Também podem usar técnicas de recuperação de informação, estatística, etc. Além disso, os desenvolvimentos realizados nessa camada guardam relação com todas as outras camadas da arquitetura.

Da primeira camada os sistemas de conhecimento valem-se das especificações e padrões de cada unidade de informação da plataforma.

Na segunda camada estão localizados os repositórios de dados relativos a cada unidade de informação, que são as principais fontes de informação para os sistemas de conhecimento.

Já na terceira camada estão disponíveis os portais que podem ser usados para apresentar o conhecimento gerado e também usar os *Web Services* para fornecer "serviços Web inteligentes".

No caso da Plataforma Lattes, há alguns trabalhos relativos à camada de sistemas de conhecimento, publicados pela comunidade científica e tecnológica (ROMÃO, 2002; NIEDERAUER, 2002; GONÇALVES, 2000). Existem também os projetos desenvolvidos dentro da Plataforma Lattes, como, por exemplo, Estratificação, Demografia Curricular, entre outros.

Assim, os dois sistemas desenvolvidos – Lattes Egressos e Lattes Colaboradores – estão relacionados diretamente com o topo da pirâmide da arquitetura de plataforma de governo, onde estão localizados os sistemas de conhecimento.

5.3 Lattes Egressos

O Diretório Lattes de Egressos[9] é um recurso da Plataforma Lattes de CT&I que permite a realização de análises sobre a distribuição de egressos de cursos de Instituições de Ensino Superior (IES) do País, com base nas informações constantes no currículo Lattes.

A análise de egressos poder ser um mecanismo importante, por exemplo, na busca por conhecimento institucional, com o qual as instituições podem desenvolver meios para a avaliação e adequação dos currículos dos seus cursos, reorientar suas atividades acadêmicas e captar as demandas do mercado de trabalho. Assim, conhecendo no que e onde estão trabalhando os seus egressos, uma instituição pode definir melhor as políticas de administração de seus cursos.

O estudo de egressos pode também ser interessante em casos mais específicos, como, por exemplo, na avaliação dos planos nacionais de desenvolvimento do País, segundo a análise da efetivação de metas de formação de recursos humanos desses planos, como no trabalho realizado por Queirós (2001). Portanto, o uso de instrumentos que facilitem a análise de informações sobre formados de uma instituição/curso é uma forma de gestão do conhecimento. A seguir, apresentam-se

[9] http://lattes.cnpq.br/lattesegressos

as fases de elaboração do Sistema Lattes Egressos de acordo com o método proposto.

FASE I – DIRETRIZES PARA TRADUÇÃO DAS FONTES DE INFORMAÇÃO NA ONTOLOGIA DE DESCRIÇÃO DE REDES DE RELACIONAMENTOS

Nessa aplicação, o tipo de fonte de informação utilizado foi o *Data Warehouse*. Mais especificamente, um *datamart* de currículos que representa os currículos dos usuários da Plataforma Lattes. O desenvolvimento teve início com uma solicitação feita pelo analista de domínio: "queremos fazer análises sobre os egressos de uma instituição distribuídos segundo os diversos atributos de análise disponíveis em seus currículos". O próximo passo foi definir qual seria o conceito de egresso neste sistema: "um egresso de uma instituição é uma pessoa que tenha uma formação concluída nessa instituição em determinado curso e nível". Assim, segundo a regra definida pelo analista de domínio, o conjunto de egressos de uma instituição qualquer é gerado pela regra:

SELECIONAR TODOS OS CURRÍCULOS CUJA FORMAÇÃO NO NÍVEL 'N' OCORREU NA INSTITUIÇÃO 'INST' NO CURSO DA ÁREA 'AR'; onde:

N' – graduação, especialização, mestrado, doutorado, MBA (segundo curso)
INST – Instituição de Origem dos Egressos
AR – Área do Curso de Formação

O analista de domínio, a partir de sua experiência e conhecimento sobre o domínio da aplicação, e sabendo dos interesses dos possíveis usuários dela, especificou as análises que deveriam inicialmente estar disponíveis no sistema. O analista técnico, por sua vez, de posse dessas informações e valendo-se de seu conhecimento sobre a base de dados, tendo acesso aos metadados, propôs pequenas modificações em algumas análises e acrescentou outras que não tinham sido previamente solicitadas pelo analista, porém que eram interessantes para o sistema.

Portanto, a partir da interação entre o analista de domínio e o analista técnico na fonte de informação foram identificados os possíveis contextos de análise e os indicadores que podem ser gerados com o estudo de egressos.Cada um desses

contextos de análise foi dividido em unidades de informação utilizadas na geração da rede de relacionamentos. Os contextos, as análises possíveis e as unidades de informação são apresentados no Quadro 5.1.

CONTEXTOS	UNIDADES DE INFORMAÇÃO	ANÁLISES POSSÍVEIS
Atividade profissional	UF Instituição Atividade-fim Atividade-meio	Distribuição de egressos de um curso segundo a Unidade da Federação em que eles atuam profissionalmente ou segundo o tipo de atividade que exercem (classificadas em atividades-meio e atividades-fim para IES).
Área de atuação	Grande área Área	Distribuição de egressos de um curso segundo a grande área e a área do conhecimento em que atuam.
Endereço profissional	País Região UF Instituição	Distribuição de egressos de um curso segundo o local onde eles exercem sua atividade profissional (país, região ou UF da instituição ou empresa em que atuam).
Formação acadêmica	UF Instituição Nível Grande área do curso Área do Curso Período de início Período de término	Distribuição de egressos de um curso segundo a sua formação acadêmica. Para tal, o usuário pode verificar em que UF os egressos do curso realizaram sua formação de maior título, com que níveis de formação estão atualmente, qual a distribuição por grande área de formação desses egressos do curso bem como os períodos (início e fim) em que os seus cursos de formação foram realizados.
Formação complementar	UF Instituição Nível Grande área do curso Área do curso Período de início Período de término	Distribuição de egressos de um curso segundo a sua formação complementar (quanto a cursos adicionais à formação acadêmica). Para tal, o usuário pode verificar em que UF os egressos do curso realizaram sua formação de maior título, com que níveis de formação estão atualmente, qual a distribuição por grande área de formação desses egressos do curso bem como os períodos (início e fim) em que os seus cursos de formação foram realizados.
Identificação	País de nascimento UF de nascimento Faixa etária Sexo	Distribuição de egressos de um curso segundo critérios relacionados à identificação desses ex-alunos, possibilitando conhecer a sua distribuição por nacionalidade, faixa etária, UF de nascimento e sexo.
Produção C&T	Tipo Subtipo Idioma Período Grande área Área	Distribuição de egressos de um curso segundo critérios relacionados à produção científica desses ex-alunos, possibilitando conhecer a sua distribuição segundo o tipo, idioma, período ou a grande área do conhecimento da produção realizada.

Quadro 5.1 - Análises sobre egressos

Como a aplicação solicitada demonstra, a cada análise os egressos de uma determinada instituição, distribuídos segundo uma das unidades de informação apresentadas no Quadro 5.1, a rede de relacionamentos será formada por uma instância da unidade de informação *InstituicaoFormacao* (que será informada pelo

usuário) e por seus relacionamentos com as instâncias da outra unidade de informação. Assim, por exemplo, suponhamos que o usuário queira ver os egressos da UFSC, curso de Matemática, nível mestrado, distribuídos de acordo com a faixa etária. Nessa rede, um vértice representa a UFSC (unidade de informação *InstituicaoFormacao*) e há um vértice para cada faixa etária (unidade de informação *FaixaEtaria*) que possui um relacionamento com a UFSC.

Compete ao analista técnico na fonte de informação verificar quais dimensões, tabelas de fato, atributos ou a combinação desses, bem como as regras, são necessários para representar cada uma das unidades de informação e seus relacionamentos com a unidade de informação *InstituicaoFormacao*.

Para ilustrar isso, apresentam-se abaixo as dimensões e regras para as unidades *InstituicaoFormacao* e *FaixaEtaria*.

a) ***InstituicaoFormacao*.** Observando o modelo de dados, o analista técnico identificou que a dimensão DI_FORMACAO (conforme o próprio nome já indica) possui ligação direta com o estudo de egressos de uma instituição. Uma das regras de negócio define que o campo STS_FORMACAO_CONCLUIDA seja igual a 'S'. Isso se faz necessário para o sistema selecionar somente as pessoas com formações concluídas. Como é o usuário que escolhe a instituição, o curso e o nível que serão usados nas análises, devem-se fazer filtros para os campos que representam esses valores: COD_INST, para a instituição; COD_AREA_CONHEC, para a área do curso; e COD_NIVEL_FORM para o nível de formação.

b) ***FaixaEtaria*.** Observando o modelo de dados, o analista técnico identificou que a dimensão DI_PESSOA possui um campo que identifica em qual faixa etária o pesquisador se localiza. Nesse caso, a consulta deve buscar o total de pessoas em cada uma das faixas etárias presentes no banco de dados.

Em a) e b) estão as dimensões e regras para as unidades de informação *InstituicaoFormacao* e *FaixaEtaria*. Para saber qual o valor da relação entre o vértice da instituição escolhida pelo usuário e cada um dos vértices representando uma das faixas etárias, basta realizar a interseção entre o conjunto de pessoas retornado da consulta de instituição e o conjunto de pessoas retornado para cada uma das faixas

etárias. A quantidade de elementos de cada conjunto resultante será o valor da relação.

O exemplo apresentado anteriormente resolve o problema para as análises entre as unidades de informação *InstituicaoFormacao* e *FaixaEtaria*. Esse procedimento foi repetido para *InstituicaoFormacao* e para cada uma das unidades de informação presentes no Quadro 5.1.

FASE II – GERAÇÃO DE UM ARQUIVO XML QUE DESCREVE A REDE DE RELACIONAMENTOS

Para geração do arquivo XML que descreve a rede de relacionamentos foi implementado um software que faz consultas em SQL (*Structured Query Language*) para realizar as buscas de acordo comas dimensões e regras definidas na FASE I.

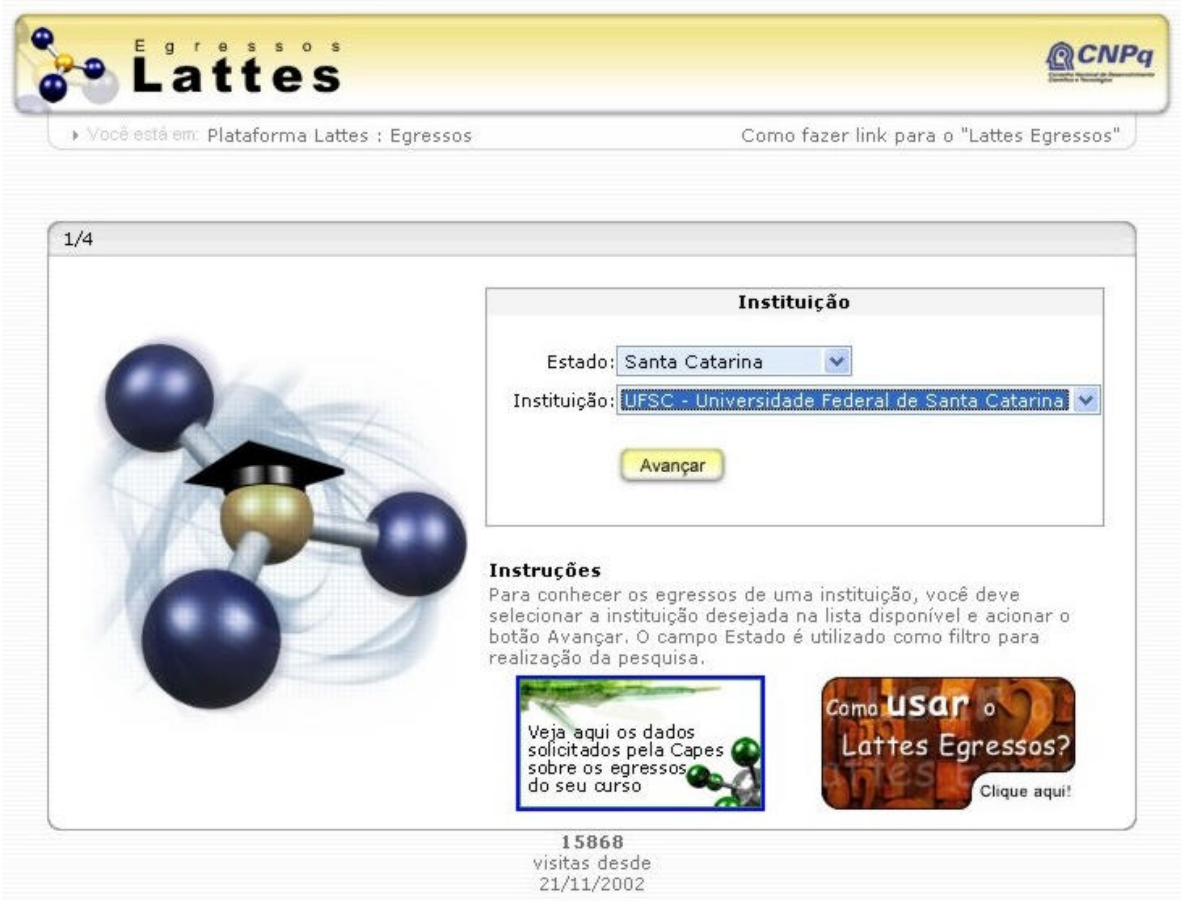

Figura 5.3 - Tela de escolha da Instituição

Para ver um exemplo do arquivo XML no Lattes Egressos, supõe-se que essa rede deva representar os egressos da UFSC, do curso de Ciência da Computação, de nível graduação, distribuídos segundo a faixa etária. Para isso o usuário da aplicação deve escolher uma instituição na tela inicial do site do Lattes Egressos, conforme pode ser visto na Figura 5.3.

Depois de escolhida a instituição, o usuário deve selecionar a área do curso e o nível de formação e, na tela seguinte, o atributo de análise que corresponde às unidades de informação (Figura 5.4).

Figura 5.4 - Telas de escolha da área e nível do curso e atributo de análise

Usando esses parâmetros (UFSC, Ciência da Computação, Graduação e Faixa Etária), o sistema gera a consulta SQL para as unidades de informação *InstituicaoFormacao* e *FaixaEtaria*. Os dados são carregados e escritos no arquivo XML de acordo com a ontologia proposta. Esse arquivo pode ser visto na Figura 5.5.

O componente de grafos implementado recebe como entrada o arquivo nesse formato. Para garantir a consistência dos dados, um XML Schema é usado para validar o arquivo. Depois que o arquivo é validado, o componente gera um grafo G=(V, A). Nesse caso, V = {1, 2, 3, 4, 5, 7, 8} e A = {(1, 2), (1, 3), (1, 4), (1, 5), (1, 6), (1, 7), (1, 8)}.

```
<grafo>
 <vertices>
  <vertice id="1">
   <unidade-informacao>InstituicaoFormacao</unidade-informacao>
   <descricao>UFSC</descricao>
  </vertice>
  <vertice id="2">
   <unidade-informacao>FaixaEtaria</unidade-informacao>
   <descricao>20-24</descricao>
  </vertice>
 <vertice id="3">
   <unidade-informacao>FaixaEtaria</unidade-informacao>
   <descricao>25-29</descricao>
 </vertice>
<vertice id="4">
   <unidade-informacao>FaixaEtaria</unidade-informacao>
   <descricao>30-34</descricao>
 </vertice>
<vertice id="5">
   <unidade-informacao>FaixaEtaria</unidade-informacao>
   <descricao>35-39</descricao>
 </vertice>
<vertice id="6">
   <unidade-informacao>FaixaEtaria</unidade-informacao>
   <descricao>40-44</descricao>
 </vertice>
<vertice id="7">
   <unidade-informacao>FaixaEtaria</unidade-informacao>
   <descricao>45-49</descricao>
 </vertice>
<vertice id="8">
   <unidade-informacao>FaixaEtaria</unidade-informacao>
   <descricao>50-54</descricao>
 </vertice>
</vertices>
<arestas>
  <aresta id="1" id-origem="1" id-destino="2">
   <valor>44</valor>
   <orientado>false</orientado>
   <descricao>Total de egressos</descricao>
  </aresta>
  <aresta id="2" id-origem="1" id-destino="3">
   <valor>58</valor>
   <orientado>false</orientado>
   <descricao>Total de egressos</descricao>
  </aresta>
  <aresta id="3" id-origem="1" id-destino="4">
   <valor>49</valor>
   <orientado>false</orientado>
   <descricao>Total de egressos</descricao>
  </aresta>
  <aresta id="4" id-origem="1" id-destino="5">
   <valor>35</valor>
   <orientado>false</orientado>
   <descricao>Total de egressos</descricao>
  </aresta>
  <aresta id="5" id-origem="1" id-destino="6">
   <valor>23</valor>
   <orientado>false</orientado>
   <descricao>Total de egressos</descricao>
  </aresta>
  <aresta id="6" id-origem="1" id-destino="7">
   <valor>6</valor>
   <orientado>false</orientado>
   <descricao>Total de egressos</descricao>
  </aresta>
  <aresta id="7" id-origem="1" id-destino="8">
   <valor>1</valor>
   <orientado>false</orientado>
   <descricao>Total de egressos</descricao>
  </aresta>
 </arestas>
</grafo>
```

Figura 5.5 - Exemplo de arquivo XML do Lattes Egressos

FASE III – EXTENSÃO DO ARQUIVO XML PARA APLICAÇÕES DE VISUALIZAÇÃO DOS RELACIONAMENTOS EXISTENTES

Como todas as análises do Lattes Egressos possuem um vértice que está conectado ao demais, sendo que as possíveis conexões entre os demais vértices são ignoradas, o grafo é denominado de radial. NaFigura 5.6 abaixo são mostradas todas as possibilidades de visualização, considerandose todas as unidades de informação.

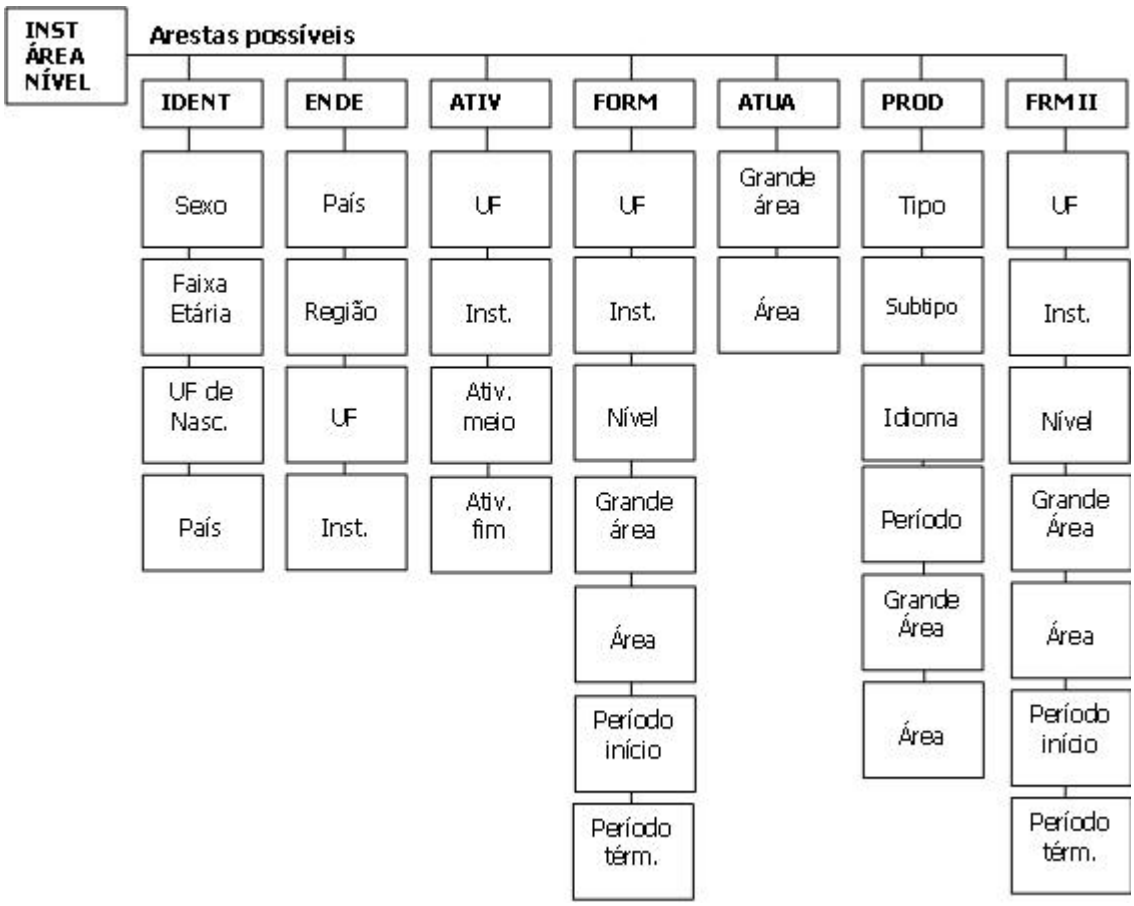

Figura 5.6 - Vértices e arestas possíveis

Para que a visualização do grafo (ou rede) que está representado no arquivo gerado na FASE II seja possível, esse arquivo deve ser estendido para incluir mais informações que são necessárias à visualização. Além das informações do arquivo apresentado na fase anterior, esse arquivo estendido deve possuir descrição e descrição abreviada sobre cada um dos vértices e arestas da rede. Essas

descrições são usadas como rótulo explicativo para o usuário. O arquivo também deve ter um elemento que aponta qual ação ocorrerá quando o usuário clicar sobre um vértice ou uma aresta. Além disso, há um parâmetro que indica ao componente como cada vértice deve ser representado visualmente. Esse arquivo pode ser visto na Figura 5.7.

```
<grafo>
 <vertices>
  <vertice id="1">
    <unidade-informacao>InstituicaoFormacao</unidade-informacao>
    <descricao-abreviada>UFSC</descricao-abreviada>
    <descricao>Universidade Federal de Santa Catarina</descricao>
    <on-click>enviarVertice(`1')</on-click>
    <imagem>CIRCULO</imagem>
  </vertice>
  <vertice id="2">
    <unidade-informacao>FaixaEtaria</unidade-informacao>
    <descricao-abreviada>20-24</descricao-abreviada>
    <descricao>De 20 a 24 anos</descricao>
    <on-click>enviarVertice (`2')</on-click>
    <imagem>CIRCULO</imagem>
  </vertice>
  <vertice id="3">
    <unidade-informacao>FaixaEtaria</unidade-informacao>
    <descricao-abreviada>25-29</descricao-abreviada>
    <descricao>De 25 a 29 anos</descricao>
    <on-click>enviarVertice (`3')</on-click>
    <imagem>CIRCULO</imagem>
  </vertice>

  ...
  ...

  <vertice id="8">
    <unidade-informacao>FaixaEtaria</unidade-informacao>
    <descricao-abreviada>50-54</descricao-abreviada>
    <descricao>De 50 a 54 anos</descricao>
    <on-click>enviarVertice (`8') </on-click>
    <imagem>CIRCULO</imagem>
  </vertice>
 </vertices>
 <arestas>
  <aresta id="1" id-origem="1" id-destino="2">
    <valor>44</valor>
    <orientado>false</orientado>
    <descricao-abreviada>Egressos</descricao-abreviada>
    <descricao>Total de Egressos</descricao>
    <on-click>enviarAresta(`1')</on-click>
  </aresta>
  <aresta id="2" id-origem="1" id-destino="3">
    <valor>58</valor>
    <orientado>false</orientado>
    <descricao-abreviada>Egressos</descricao-abreviada>
    <descricao>Total de Egressos</descricao>
    <on-click>enviarAresta(`2') </on-click>
  </aresta>

  ....
  ....

  <aresta id="7" id-origem="1" id-destino="8">
    <valor>1</valor>
    <orientado>false</orientado>
    <descricao-abreviada>Egressos</descricao-abreviada>
    <descricao>Total de Egressos</descricao>
    <on-click>enviarAresta(`7')</on-click>
  </aresta>
</arestas>
</grafo>
```

Figura 5.7 - Exemplo de extensão arquivo XML do Lattes Egressos

FASE IV – APLICAÇÃO DE ALGORITMOS DE TEORIA DOS GRAFOS E *LINK ANALYSIS*

A aplicação de algoritmos da Teoria dos Grafos e *Link Analysis* pode ser feita a partir do grafo obtido na FASE II. Uma opção é o cálculo de densidade do grafo, o que permite apresentar uma medida de comparação entre cada grafo gerado pelo sistema. Assim, é possível medir a centralidade local de um vértice através de seu grau, medida esta que, baseada em grau, corresponde à noção intuitiva de quão bem conectado um vértice está em seu ambiente local.

FASE V – VISUALIZAÇÃO DOS RELACIONAMENTOS EXISTENTES A PARTIR DO ARQUIVO XML OBTIDO NA FASE III

A partir do arquivo estendido, gerado na FASE III, o componente de visualização construído para o Lattes Egressos consegue gerar uma representação gráfica da rede. Para isso, o componente lê o arquivo, faz a validação com um XML Schema para verificar se o arquivo segue a ontologia proposta, extrai as informações e, então, gera a imagem com o grafo. Para o arquivo XML do exemplo mostrado na Figura 5.7, o componente de visualização gera a imagem apresentada na Figura 5.8.

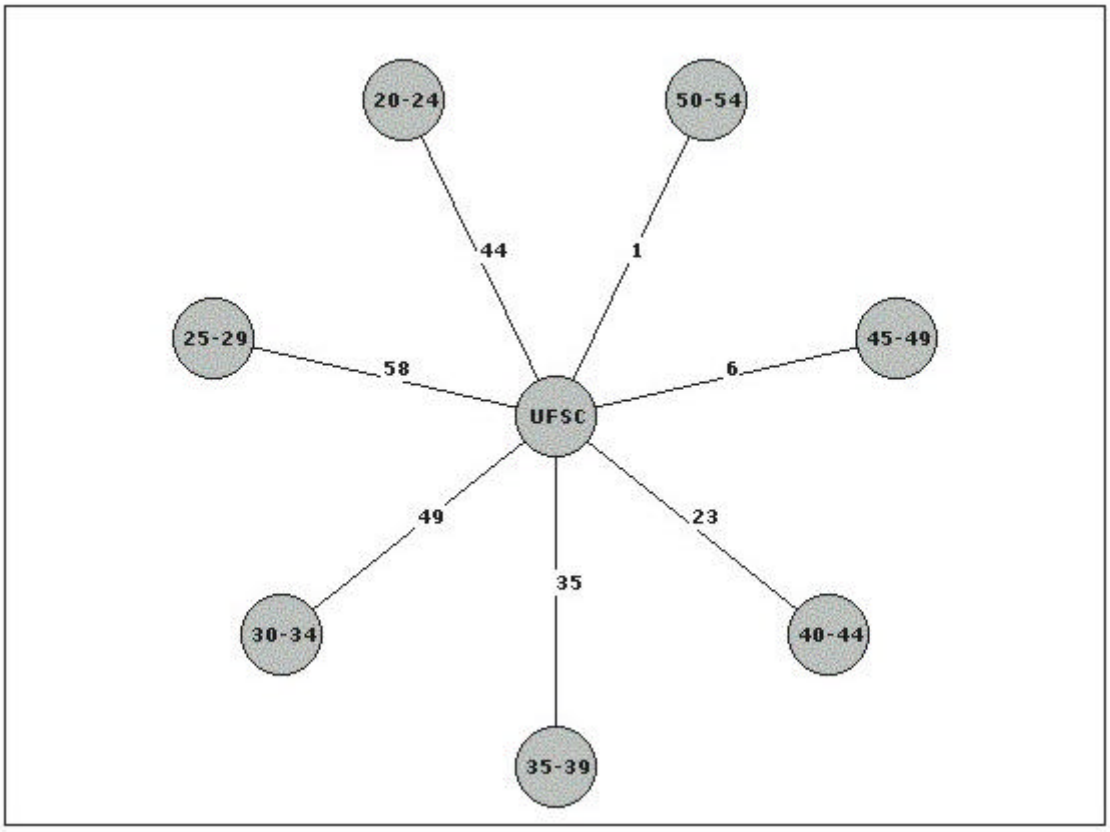

Figura 5.8 - Egressos da instituição UFSC, da área de Ciência da Computação, nível graduação, distribuídos por faixa etária

As possibilidades de análise a partir do Lattes Egressos são variadas. Em outro exemplo, os parâmetros utilizados na consulta foram: Instituição igual a UFSC (Universidade Federal de Santa Catarina), área igual a Engenharia de Produção, nível igual a Doutorado e atributo de análise igual a Grande Área de Atuação. O grafo resultante dessa consulta é apresentado na Figura 5.9.

Percebe-se que os egressos do PPGEP/UFSC em nível de doutorado estão atuando em todas as 8 grandes áreas cadastradas na tabela de áreas de conhecimento do CNPQ (2004b), evidenciando-se o caráter multidisciplinar do programa. Dessa forma, elucida-se uma informação existente porém indisponível pelos sistemas anteriores. Em conseqüência, isso aumenta o conhecimento em relação à rede de formação de pós-graduação brasileira, com possível impacto nos futuros processos de tomada de decisão sobre essa rede.

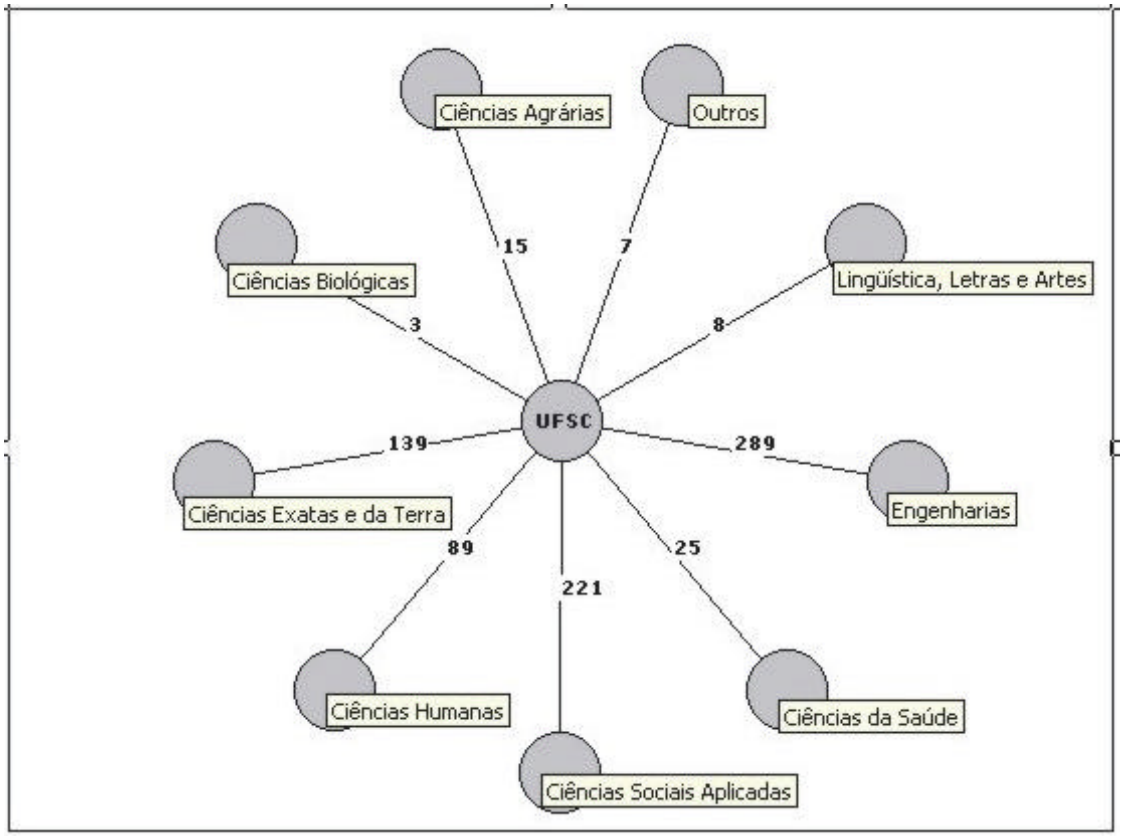

Figura 5.9 - Egressos da instituição UFSC, da área de Engenharia de Produção, nível doutorado, distribuídos por grande de atuação

5.4 Lattes Colaboradores

O Lattes Colaboradores é uma aplicação da Plataforma Lattes que tem por objetivo apresentar as colaborações entre pesquisadorescom base nas informações constantes no currículo Lattes, a saber: colaborações de produção C&T, orientação e participação em projetos. A seguir, apresenta-se o Lattes Colaboradores de acordo com as fases do método proposto.

FASE I – DIRETRIZES PARA TRADUÇÃO DAS FONTES DE INFORMAÇÃO NA ONTOLOGIA DE DESCRIÇÃO DE REDES DE RELACIONAMENTOS

O analista de domínio solicitou o desenvolvimento de um sistema que apresentasse de forma visual as colaborações científicas dos pesquisadores com currículo Lattes. O passo a seguir foi definir qual seria o conceito de Colaboração

neste sistema. Assim, "considerando as informações da base de currículos, dois pesquisadores colaboram se eles se enquadrarem em uma das seguintes situações: se possuem um ou mais artigos juntos; se um é orientador do outro; ou se eles participam de um ou mais projetos de pesquisa juntos". Logo, percebe-se que o Lattes Colaboradores utiliza apenas uma unidade de informação – o currículo dos pesquisadores.

O desenvolvimento desse sistema foi dividido em duas partes. Na primeira, utilizou-se como fonte de informação uma base relacional, a base operacional de currículos da Plataforma Lattes. Observando-se o modelo de dados dessa base, o analista técnico identificou que a tabela EN_COAUTOR armazena os colaboradores que cada pesquisador informou em seu currículo. Mais especificamente, cada pesquisador informa o nome e a citação bibliográfica de cada um de seus co-autores, alunos orientados e colegas de projetos. Contudo, a identificação desses colaboradores que estão informados nos currículos não pode ser feita automaticamente devido ao fato de que os pesquisadores registram os nomes de seus colaboradores de forma incompleta. Portanto, uma técnica de comparação de textos foi utilizada para fazer o *matching* entre o colaborador informado no currículo e o seu próprio currículo na Plataforma Lattes, utilizando o nome e a citação bibliográfica. Desse modo, depois de identificados os colaboradores, uma base *warehouse* foi carregada com as informações sobre os colaboradores de cada pesquisador. Essa base, que é o *datamart* de currículos, é usada na segunda parte do desenvolvimento.

As consultas para localizar os colaboradores de um pesquisador devem ser feitas sobre uma dimensão apenas: DI_COAUTOR. Logo, para cada pesquisador deve-se fazer uma contagem nessa dimensão, apenas fazendo a classificação por tipo de colaborador: co-autor, orientado ou co-participante em projetos.

FASE II – GERAÇÃO DE UM ARQUIVO XML QUE DESCREVE A REDE DE RELACIONAMENTOS

Para geração do arquivo XML que descreve a rede de relacionamentos foi implementado um software que faz a consulta em SQL para realizar contagem do

número de colaboradores de cada pesquisador classificandeos como co-autor, orientando e co-participante em projetos.

Para ver um exemplo do arquivo XML no Lattes Colaboradores, suponha que essa rede deva representar os co-autores de um determinado pesquisador. Para isso, o usuário da aplicação deve realizar uma busca por nome do pesquisador na tela inicial do site do sistema Lattes Colaboradores. Isso pode ser visto na Figura 5.10.

Figura 5.10 - Tela de busca por nome de pesquisadores

Depois de escolhido o pesquisador, o sistema gera a consulta SQL para realizar a busca. Os dados são carregados e escritos no arquivo XML de acordo com a ontologia proposta. Esse arquivo pode ser visto na Figura 5.11.

O componente de grafos implementado recebe como entrada o arquivo nesse formato. Para garantir a consistência dos dados, um XML Schema é usado para validar o arquivo. Depois dessa validação, o componente gera um grafo G=(V, A). Nesse caso, o V = {1, 2, 3, 4, 5, 7, 8, 9, 10, 11, 12, 13, 14} e A = {(1, 2), (1, 3), (1, 4), (1, 5), (1, 6), (1, 7), (1, 8), (1, 9), (1 , 10), (1, 11), (1, 12), (1, 13), (1, 14)}.

```
<grafo>
 <vertices>
  <vertice id="1">
    <unidade-informacao>Curriculo</unidade-informacao>
    <descricao>Denílson Sell</descricao>
  </vertice>
  <vertice id="2">
    <unidade-informacao>Curriculo</unidade-informacao>
    <descricao>Alexandre Leopoldo Gonçalves</descricao>
  </vertice>
  <vertice id="3">
    <unidade-informacao>Curriculo</unidade-informacao>
    <descricao>Ricardo Noronha Rieke</descricao>
  </vertice>

  ...
  ...

  <vertice id="14">
    <unidade-informacao>Curriculo</unidade-informacao>
    <descricao>João Paulo da Costa</descricao>
  </vertice>
 </vertices>
 <arestas>
  <aresta id="1" id-origem="1" id-destino="2">
     <valor>10</valor>
    <orientado>false</orientado>
    <descricao>Total de produções</descricao>
  </aresta>
  <aresta id="2" id-origem="1" id-destino="3">
     <valor>6</valor>
    <orientado>false</orientado>
    <descricao>Total de produções</descricao>
  </aresta>

  ...
  ...

  <aresta id="13" id-origem="1" id-destino="14">
     <valor>1</valor>
    <orientado>false</orientado>
    <descricao>Total de produções</descricao>
  </aresta>
 </arestas>
</grafo>
```

Figura 5.11 - Exemplo de arquivo XML do Lattes Colaboradores

FASE III – EXTENSÃO DO ARQUIVO XML PARA APLICAÇÕES DE VISUALIZAÇÃO DOS RELACIONAMENTOS EXISTENTES

De forma semelhante ao Lattes Egressos, no Lattes Colaboradores os grafos gerados possuem um vértice (pesquisador, da unidade Curículo) que está conectado ao demais (colaboradores, também da unidade Currículo), sendo essas possíveis conexões entre os demais vértices ignoradas Portanto, o grafo também é radial.

Para ser possível a visualização do grafo (ou rede) que está representado no arquivo gerado na FASE II, esse arquivo deve ser estendido para incluir mais informações que são necessárias à visualização. Além das informações do arquivo apresentado na fase anterior, esse arquivo estendido deve possuir descrição e

descrição abreviada sobre cada um dos vértices e arestas da rede. Essas descrições são usadas como rótulos explicativos para o usuário. O arquivo também deve ter um elemento que indica qual ação ocorrerá quando o usuário clicar sobre um vértice ou aresta. Além disso, há um parâmetro que indica ao componente como cada vértice deve ser representado visualmente. Esse arquivo pode ser visto na Figura 5.12.

```
<grafo>
 <vertices>
  <vertice id="1">
    <unidade-informacao>Curriculo</unidade-informacao>
    <descricao-abreviada>D. Sell</descricao-abreviada>
    <descricao>Denilson Sell</descricao>
    <on-click>enviarVertice('1')</on-click>
    <imagem>RETANGULO</imagem>
  </vertice>
  <vertice id="2">
    <unidade-informacao>Curriculo</unidade-informacao>
    <descricao-abreviada>A. L. Gonçalves</descricao-abreviada>
    <descricao>Alexandre Leopoldo Gonçalves</descricao>
    <on-click>enviarVertice('2')</on-click>
    <imagem>RETANGULO</imagem>
  </vertice>
  <vertice id="3">
    <unidade-informacao>Curriculo</unidade-informacao>
    <descricao-abreviada>R. N. Rieke</descricao-abreviada>
    <descricao>Ricardo Noronha Rieke</descricao>
    <on-click>enviarVertice('3')</on-click>
    <imagem>RETANGULO</imagem>
  </vertice>

  ...
  ...

  <vertice id="14">
    <unidade-informacao>Curriculo</unidade-informacao>
    <descricao-abreviada>J. P. da Costa</descricao-abreviada>
    <descricao>João Paulo da Costa</descricao>
    <on-click>enviarVertice('14')</on-click>
    <imagem>RETANGULO</imagem>
  </vertice>
 </vertices>
 <arestas>
  <aresta id="1" id-origem="1" id-destino="2">
    <valor>10</valor>
    <orientado>false</orientado>
    <descricao-abreviada>Produções</descricao-abreviada>
    <descricao>Total de produções</descricao>
    <on-click>enviarAresta('1')</on-click>
  </aresta>
  <aresta id="2" id-origem="1" id-destino="3">
    <valor>6</valor>
    <orientado>false</orientado>
    <descricao-abreviada>Produções</descricao-abreviada>
    <descricao>Total de produções</descricao>
    <on-click>enviarAresta('2')</on-click>
  </aresta>

  ...
  ...

  <aresta id="13" id-origem="1" id-destino="14">
    <valor>1</valor>
    <orientado>false</orientado>
    <descricao-abreviada>Produções</descricao-abreviada>
    <descricao>Total de produções</descricao>
    <on-click>enviarAresta('13')</on-click>
  </aresta>
 </arestas>
</grafo>
```

Figura 5.12 - Exemplo de extensão do XML do Lattes Colaboradores

FASE IV – APLICAÇÃO DE ALGORITMOS DE TEORIA DOS GRAFOS E *LINK ANALYSIS*

A aplicação de algoritmos da Teoria dos Grafos e *Link Analysis* pode ser feita a partir do grafo obtido na FASE II. Uma aplicação muito interessante consiste em utilizar algoritmos para determinar a distância entre dois pesquisadores, assim como é feito para o número de Erdös. O conceito de número de Erdös é assim chamado em homenagem ao prolífero matemático húngaro Paul Erdös, um dos pais da Teoria dos Grafos, que publicou mais de 1.500 artigos em colaboração com 507 co-autores (BARABASI, 2003). O número de Erdös é a menor distância entre um dado pesquisador e Paul Erdös em redes de co-autoria. Um pesquisador que é co-autor de Paul Erdös possui número de Erdös igual a 1; um pesquisador que não publicou junto com Paul Erdös, mas é co-autor de um co-autor dele, possui número de Erdös igual a 2; e assim por diante. O próprio Paul Erdös possui número de Erdös igual a 0, e uma pessoa que não pode ser conectada a ele através da rede de co-autoria possui o número de Erdös igual a infinito. Alguns estudos mostram que a média do número de Erdös de um pesquisador é aproximadamente 4.7 (NEWMAN, 2000).

Assim, tendo-se o grafo que representa as relações de colaboração entre os pesquisadores, algoritmos de caminho mínimo poderiam ser usados para extração desse tipo de conhecimento sobre uma rede de colaboração.

FASE V – VISUALIZAÇÃO DOS RELACIONAMENTOS EXISTENTES A PARTIR DO ARQUIVO XML OBTIDO NA FASE III

A partir do arquivo estendido, gerado na FASE III, o componente de visualização construído para o Lattes Colaboradores conseguecriar uma representação gráfica da rede. Para isso, o componente lê o arquivo, faz a validação com um XML Schema para verificar se esse arquivo segue a ontologia proposta, extrai as informações e, então, gera a imagem com o grafo. Para o arquivo XML do exemplo mostrado na Figura 5.12, o componente de visualização gera a imagem apresentada na Figura 5.13.

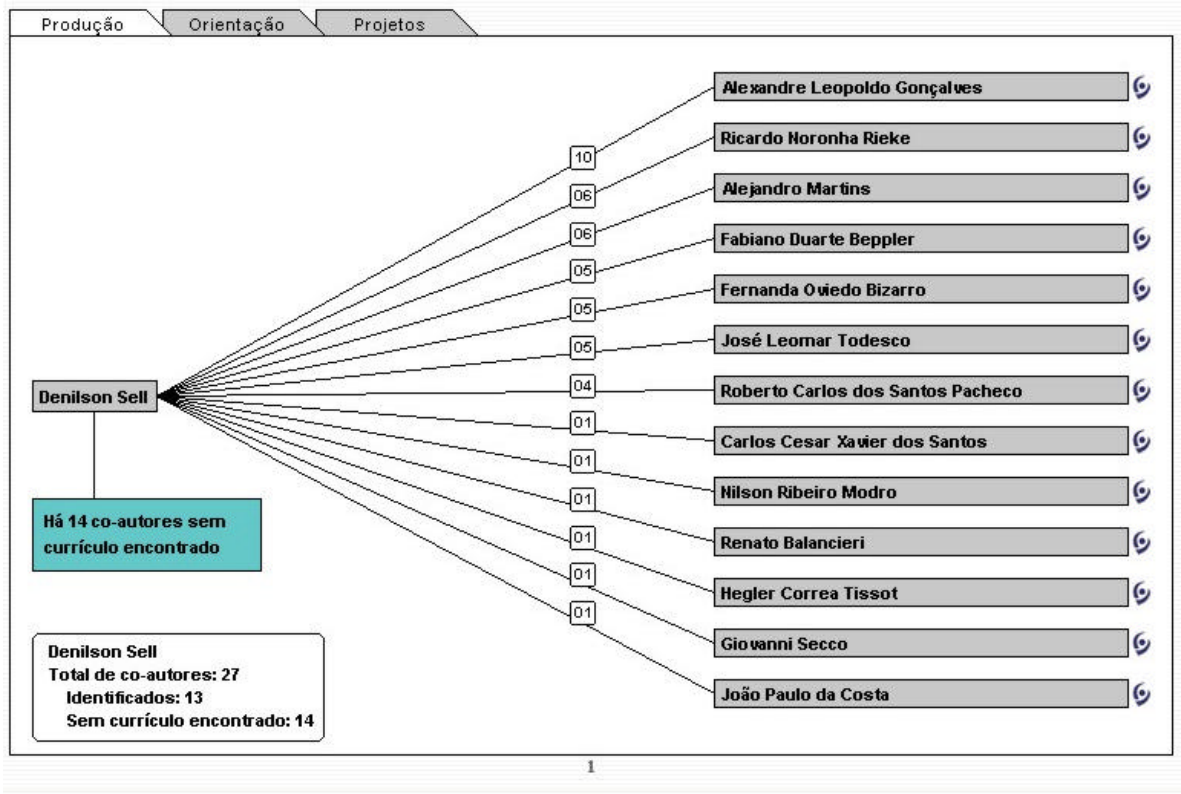

Figura 5.13 - Um pesquisador e seus co-autores

5.5 Considerações finais

Este capítulo apresentou duas aplicações do método proposto nesta dissertação. Para isso, primeiro apresentou-se a Plataforma Lattes, identificando-se em que parte de sua arquitetura estão inseridos os dois sistemas de conhecimento desenvolvidos. O primeiro sistema, o Lattes Egressos, é usado para apresentar informações sobre os egressos das IES brasileiras. O segundo, o Lattes Colaboradores, mostra as relações de colaboração entre pesquisadores com currículo Lattes.

6 CONCLUSÕES E TRABALHOS FUTUROS

Este trabalho apresentou um método que permite traduzir fontes de informação (banco de dados relacionais, *data warehouses* e documentos XML) em um formato padrão de representação de relacionamentos entre elementos do domínio do problema, de forma a viabilizar a extração de conhecimento por meio da aplicação de *Link Analysis* e Teoria dos Grafos.

Para o desenvolvimento do método, estudaram-se as área de KDD e MD, com ênfase em *Link Analysis*. Com isso, foi possível identificar métodos de extração de conhecimento baseado no estudo de relacionamentos entre elementos de um domínio, como é apresentado no Capítulo 2. A partir desse estudo, percebeu-se que ferramentas inteligentes de exploração de dados podem ser interessantes para auxiliar especialistas na visualização da conectividade entre objetos conforme aumenta o volume de informações. No Capítulo 3, apresentaram-se os estudos realizados na área da Teoria dos Grafos, que permitiram identificar métodos aplicáveis à extração de conhecimento a partir de informações representadas na forma de relacionamentos entre os elementos de um domínio.

Para possibilitar a aplicação dos métodos de TG e LA, no Capítulo 4 propôs-se um método para a transferência de informações para o domínio de análise de redes de relacionamentos, sendo apresentadas uma descrição geral do método e suas cinco fases. Para cada fase foram estabelecidas algumas diretrizes e sugeridas algumas soluções de desenvolvimento.

No Capítulo 6, a viabilidade do método foi mostrada a partir da sua aplicação no desenvolvimento de dois sistemas de conhecimentos no contexto de uma arquitetura de sistemas de governo eletrônico: o Lattes Egressos, usado para apresentar informações sobre os egressos das IES brasileiras, e o Lattes Colaboradores, que mostra as relações de colaboração entre pesquisadores com currículo Lattes.

Entre as características desejáveis de um determinado domínio para que se possa aplicar o método proposto neste trabalho, destaca-se a existência de uma fonte de informação sobre o domínio, que pode ser uma base relacional, *warehouse* ou documentos XML. Tal fonte deve conter relacionamentos, explícitos ou não, entre os seus principais elementos. Outro fator importante para a aplicação do método é a

interação entre o analista de domínio e o analista técnico na estrutura da fonte de informação.

A possibilidade de apresentação visual da rede é um ponto de diferenciação desse método em relação às demais técnicas de extração do conhecimento que apresentam apenas conhecimento textual. Isso pode ser visto, por exemplo, no Lattes Egressos. O potencial desse sistema foi mostrado em análises sobre os egressos do Programa de Pós-graduação em Engenharia de Produção (UFSC), nível doutorado, que foram apresentados segundo as grandes áreas de atuação. Foi possível observar que o PPGEP/UFSC tem uma grande influência sobre as outras áreas, visto que existem doutores oriundos desse curso atuando nas oito (8) grandes áreas do conhecimento.

A vantagem em serem utilizados arquivos XML para representar a rede de relacionamentos é que a XML constitui um formato aberto e extensível, sendo independente de sistema operacional, linguagem e fonte de informação. Essa linguagem permite enfatizar os relacionamentos, e os seus dados são legíveis por homens e máquinas. Além disso, os componentes usados para manipular esse arquivo podem ser construídos uma única vez e usados em vários domínios, desde que o arquivo seja escrito seguindo a ontologia proposta.

Quanto à busca e à análise de relacionamentos em fontes de informação, algumas áreas de IA podem contribuir para aplicações de *Link Analysis*. A seguir, discutem-se algumas dessas áreas.

a) Processamento de Linguagem Natural

Muitos domínios podem ser analisados com LA, mas obter os dados relevantes pode ser difícil. Descrições textuais freqüentemente contêm uma riqueza de relações entre pessoas, lugares e objetos. Nesse contexto, técnicas de processamento de linguagem natural são úteis para extrair de forma eficiente tais relacionamentos. Uma ferramenta poderia extrair de um texto objetos úteis (e.g. número de telefones, endereços, nomes pessoais, nome de empresas, etc.) e relações entre eles (e.g., chamadas de telefones realizadas, reuniões). Técnicas aperfeiçoadas para a

identificação de objetos e relações são necessárias para aproveitar a grande quantidade de documentos que possui potencial para o uso em LA.

b) Agentes Inteligentes

Algumas aplicações de LA provêem certo grau de automação, inteligência e suporte semi-autônomo para criação, busca e entendimento das redes. Isso se torna importante à medida que as redes crescem em tamanho e complexidade. Em particular, agentes podem ser úteis para filtrar novas informações e adicioná-las à rede ou para alertar usuários sobre novos links interessantes nessa rede com base em especificações de alto nível feitas pelo usuário. Davis e Bennett (1998) afirmam que, cada vez mais, os sistemas de conhecimento necessitarão de um contexto social e cultural para que realmente sejam inteligentes. Isso se deve ao fato da crescente necessidade de um sistema interagir com sistemas diferentes. Nesse aspecto, os agentes de colaboração (ou cooperação) poderiam ser utilizados considerando-se algumas de suas características, tais como habilidade social e autonomia (CUNHA, 2002).

c) Ontologias

Normalmente, aplicações de LA suportam raciocínio sobre tipos simples de links e nós. Por exemplo, um nó pode representar uma pessoa, e um link pode representar uma chamada telefônica. Uma aplicação pode ajudar usuários a localizarem “organizações virtuais” baseadas nos padrões de ligações. Contudo, à medida que os tipos de nós e links crescem em grandes redes, fazem-se necessários meios mais sofisticados de raciocínio. Ontologias que suportam raciocínio sobre classes de nós e os relacionamentos entre eles podem fornecer suporte de dedução muito mais sofisticado. Assim, ontologias podem ser úteis para sistemas de LA facilitando a identificação de entidades e seus relacionamentos.

d) Raciocínio Baseado em Casos

Muitas aplicações de LA são realizadas por meio da análise de casos. Por exemplo, analistas que estão procurando evidências de crime organizado podem analisar casos identificados com o sucesso no passado para tentar encontrar novos casos semelhantes. Assim, ferramentas que usem técnicas de raciocínio baseado em casos sobre estruturas gráficas complexas serão um componente interessante para futuros métodos de LA.

e) Técnicas de Busca (*Search*)

A construção e a busca de redes muitas vezes são um gargalo para aplicações de LA. Com o crescimento geométrico das redes, elas ultrapassaram a capacidade de alguns métodos de descoberta de conhecimento em dados relacionais etambém de alguns métodos de busca de sub-redes interessantes dentro de redes já formadas. Técnicas de busca de IA podem prover um salto na capacidade de busca em comparação com a busca exaustiva.

No âmbito da Plataforma Lattes, sugere-se o emprego do método neste trabalho no estudo de outros níveis de redes pesquisa, além de co-autoria, orientação e projetos de pesquisa. Os demais níveis, identificados por Balancieri (2004), são: colegas de formação, colegas de trabalho, co-participante em bancas, colegas de GP, colegas de pós-graduação, conterrâneos (UF, cidade), vizinhos (UF, cidade, CEP).

Quanto à representação dos relacionamentos, propõese utilizar arquivos RDF[10]/XML (DACONTA, 2003), em vez de um simples arquivo XML, para representar a rede de relacionamentos. Isso poderia dar mais possibilidades de raciocínio semântico sobre a rede.

[10] *Resource Description Framework*

REFERÊNCIAS BIBLIOGRÁFICAS

AGUILERA, Vicent et al. Querying XML documents in xyleme. In: ACM SIGIR WORKSHOP ON XML AND INFORMATION RETRIEVAL, 2000.

ANDERSON, T. et al. Security works. **Security Management**, v. 38, n. 17, p. 17-20, 1994.

AZAGURY, Alain; FACTOR, Michael; MANDLER, Benny. XMLFS: An XML-Aware File System. **IBM Research Lab**, Haifa, 2000.

BALANCIERI, R. **Análise de Redes de Pesquisa em uma Plataforma de Gestão em Ciência e Tecnologia:** Uma Aplicação à Plataforma Lattes. 2004. Dissertação (Mestrado em Engenharia de Produção) – Programa de Pós-Graduação em Engenharia de Produção, Universidade Federal de Santa Catarina, Florianópolis, 2004.

BALDWIN, B.; BAGGA, A. Coreference as the Foundations for Link Analysis Over Free Text Databases. In: AAAI FALL SYMPOSIUM ON ARTIFICIAL INTELLIGENCE AND LINK ANALYSIS, 1998. **Proceedings…** 1998. p. 8-13.

BARABASI, Albert-László. **Linked:** How Everything Is Connected to Everything Else and What It Means for Business, Science, and Everyday Life. USA: A Plume Book, 2003.

BEHZAD, M.; CHARTRAND, G.; LESNIAK-FOSTER, L. **Graphs and Digraphs.** Wadsworth, 1979.

BERGE, C. **Graphes et Hypergraphes.** 2. ed. Paris: Bordas, 1973.

_____. **Graphs.** 2. ed**.** Amsterdã: Elsevier Science Publishers B.V., 1985.

BERRY, M. J. A.; LINOFF, G. **Data mining techniques - for marketing, sales, and customer support.** New York: John Wiley & Sons, 1997.

BOAVENTURA NETTO, Paulo Oswaldo. **Grafos:** Teoria, Modelos e Algoritmos. 2. ed. São Paulo: Blücher, 2001.

BONIFACIO, A. S. **Ontologias e Consulta Semântica**: Uma Aplicação ao Caso Lattes. 2002. Dissertação (Mestrado em Computação) – Programa de Pós-Graduação em Computação, Universidade Federal do Rio Grande do Sul, 2002.

BRIN, S.; PAGE, L. The Anatomy of a large-scale hypertextual (Web) search engine In: THE SEVENTH INTERNATIONAL WORLD WIDE WEB CONFERENCE, 1998.

BRON, C.; KERSBOSCH, J. Algorithm 457: Finding all Cliques of an Undirected Graph. **Comm. ACM**, v. 16, p. 575-577, 1973.

CHEN, H.; LYNCH, K. J. Automatic construction of networks of concepts characterizing document databases. **IEEE Transactions on Systems, Man and Cybernetics**, v. 22, n. 5, p. 885-902, 1992.

CNPq. Conselho Nacional de Desenvolvimento Científico e Tecnológico. **Plataforma Lattes**. Disponível em: <http://lattes.cnpq.br>. Acesso em: 20 abr. 2004.

_____. **Tabela de Áreas do Conhecimento**. Disponível em: <http://www.cnpq.br/areas/tabconhecimento/index.htm>. Acesso em: 17 maio 2004b.

CRAVEN, M. et al. Learning to construct knowledge bases from the world wide web. **Artificial Inteligence**, v. 118, n. 1-2, p. 69-113, 2000.

CUNHA, L. M.; FUKS, H.; LUCENA, C. J. P. Sistemas Multiagentes e Instrução da Web. **Scientific Literature Digital Library,** 2002. [online] Disponível em: <http://citeseer.nj.nec.com/cs>. Acesso em: 20 nov. 2002.

DACONTA, Michael C.; OBRST, Leo J.; SMITH, Kevin T. **The Semantic Web**: A Guide to the Future of XML, Web Services, and Knowledge Management. Wiley Publishing, Inc., 2003.

DAVIS, T.; BENNETT, H. Towards a Theory for a Sociable Software Architecture. In: AAAI Fall SYMPOSIUM ON ARTIFICIAL INTELLIGENCE AND LINK ANALYSIS, 1998. **Proceedings…** AAAI Press, 1998. p. 63-67.

EGNOR, Daniel; LORD, Robert. **Structured Information Retrieval using XML**. Disponível em: <http//www.xyzfind.com/>. Seatle, Washington, USA, 2000.

ERDOS (2004). **The Erdös Number Project**. Disponível em: <http://personalwebs.oakland.edu/~grossman/erdoshp.html>. Acesso em: 29 abr. 2004.

FAYYAD, U.; PIATETSKY-SHAPIRO, G.; SMYTH, P. From data mining to knowledge discovery: an overview. **Advances in Knowledge Discovery and Data Mining,** Menlo Park, CA: AAAI Press, 1996a. p. 1-34.

_____. Knowledge Discovery and Data Mining: Towards a Unifying Framework. In: SECOND INTERNATIONAL CONFERENCE ON KD & DM, 1996b, Portland, Oregon.

FORD, L. K.; FULKERSON, D. R. **Flows in Networks.** Princeton, 1962.

FRUJUELLE, R. **Um Sistema Computacional para Análise Grafo-Teórica de Sociogramas**. 1990. Dissertação (Mestrado em Engenharia de Produção)– COPPE/UFRJ, 1990.

FULKERSON, D. R. An Out-of-kilter Method for Minimal Cost Flow Problems. **SIAM J. Appl. Maths**, v. 9, n. 1, p. 18-27, 1961.

GOEBEL, M.; GRUENWALD, L. A survey of data mining and knowledge discovery software tools. In: SIGKDD EXPLORATIONS, June 1999.

GOLDBERG, Henry G; SENATOR, Ted E. Restructuring Databases for Knowledge Discovery by Consolidation and Link Formation. In: AAAI FALL SYMPOSIUM ON ARTIFICIAL INTELLIGENCE AND LINK ANALYSIS, 1998.

GONÇALVES, A. L. **Utilização de técnicas de mineração de dados em bases de C&T: uma análise dos grupos de pesquisa no Brasil**. 2000. Dissertação (Mestrado em Engenharia de Produção) – Programa de Pós-Graduação em Engenharia de Produção, Universidade Federal de Santa Catarina, 2000.

GONDRAN, M.; MINOUX, M. **Graphes et Algorithmes.** 2. ed. Eyrolles, 1985.

GOODMAN, S.; HEDETNIEMI, S. Eulerian Walks in Graphs. **SIAM J. Comp**., v. 2, n. 1, p. 16-27, 1973.

GRADY, Nancy W.; TUFANO, Daniel R.; FLANERY JUNIOR, Raymond E. Immersive Visualization for Link Analysis. In: AAAI FALL SYMPOSIUM ON ARTIFICIAL INTELLIGENCE AND LINK ANALYSIS, 1998.

GUIMARÃES, R. et al. **A pesquisa no Brasil** – Perfil da pesquisa no Brasil e hierarquização dos grupos de pesquisa a partir dos dados do Diretório dos Grupos de Pesquisa no Brasil. 1999.

HANNEMAN, R. **Introduction to Social Network Methods** – Textbook. Universidade da Califórnia, Riverside, 2000.

HARRISON, T. H. **Intranet data warehouse**. Berkeley, 1998.

HAUCK, R. V.; CHAU, M.; CHEN, H. COPLINK – Arming Law Enforcement with New Knowledge Management Technologies. In: MCIVER, W.; ELMAGARMID, A. (Ed.). **Advances in Digital Government**: Technology, Human Factors, and Policy, Kluwer. Academic Publishers, April 2002.

HESSELBEIN, Frances et al. **A Organização do Futuro**: como preparar hoje as empresas de amanhã. São Paulo: Futura, 1997.

HORN, R. D.; BIRDWELL, J. D.; LEEDY, L. W. Link discovery tool. In: COUNTERDRUG TECHNOLOGY ASSESSMENT CENTER AND COUNTERDRUG TECHNOLOGY ASSESSMENT CENTER'S ONDCP/CTAC INTERNATIONAL SYMPOSIUM. **Proceedings…** 1997, Chicago, IL.

IVANESCU, P. L. Hammer; RUDEANU. **Boolean Methods in Operations Research and Related Areas.** Springer-Verlag, 1968.

JENSEN, David. Statistical Challenges of Inductive Inference in Linked Data In: AAAI FALL SYMPOSIUM ON ARTIFICIAL INTELLIGENCE AND LINK ANALYSIS, 1998.

KAMVAR, S. D. et al. **Exploiting the block structure for the computing PageRank**. Stanford University Technical Report, 2003.

KAUFMANN, A. **Introduction à la Combinatorique en Vue des Applications.** Dunod, 1968.

KHARE, R.; RIFKIN, A. The origin of (document) species. **Computer Networks and ISDN Systems**, v. 30, Issues 1-7, p. 389-397, April 1998.

KISELEV, M. V. PolyAnalyst - a machine discovery system inferring functional programs. In: AAAI WORKSHOP ON KNOWLEDGE DISCOVERY IN DATABASES'94, 1994, Seattle. **Proceedings…** Seattle, 1994. p. 237-249.

KISELEV, M. V.; ANANYAN, S. M.; ARSENIEV, S. B. PolyAnalyst Data Analysis Techniques. In: KDD'98, 1998, New York. **Proceedings…** USA, August 1998. p. 7-31.

KLEINBERG, J. Authoritative sources in a hyperlinked environment. In: 9th ACM-SIAM SYMPOSIUM ON DISCRETE ALGORITHMS, 1998. **Proceedings...** 1998.

KRUSKAL, J. B. **On the Shortest Spanning Tree of a Graph and the Traveling Salesman Problem**, Proc. Am. Math. Soc. 7, 1956. p. 48-50.

LEE, Richard. Automatic Information Extraction from Documents: A Tool for Intelligence and Law Enforcement Analysts. In: AAAI FALL SYMPOSIUM ON ARTIFICIAL INTELLIGENCE AND LINK ANALYSIS, 1998.

LESK, M. **Practical Digital Libraries**. Los Altos, CA: Morgan Kauffmann, 1997.

LOPES, J. M. **Determinação dos Subconjuntos de Articulação Minimais de um Grafo.** 1980. Dissertação (Mestrado em Engenharia de Produção)– COPPE/UFRJ, 1980.

LYONS, Donal; TSEYTIN, Gregory S. Phenomenal Data Mining and Link Analysis. In: AAAI FALL SYMPOSIUM ON ARTIFICIAL INTELLIGENCE AND LINK ANALYSIS, 1998.

MANNILA, H. Data mining: machine learning, statistics, and databases. In:EIGHT INTERNATIONAL CONFERENCE ON SCIENTIFIC AND DATABASE MANAGEMENT, 1996.

MARTIN, B.; SUBRAMANIAN, G.; YAVERBAUM, G. **Benefits from expert systems**: an exploratory investigation, Expert Systems With Applications. v. 11, n. 1, p. 53-58, 1996.

MARTINS, S. R. et al. Geração automática de texto para gestão de conhecimento em C&T a partir da Plataforma Lattes. In: ENCONTRO NACIONAL DE ENGENHARIA DE PRODUÇÃO (ENEGEP), 2004, Florianópolis.

MILLER, J. C. et al. Modifications of Kleinberg's HITS Algorithm using Matrix Exponentiation and Web Log Records In: 24th INT'L ACM SIGIR CONFERENCE ON RESEARCH AND DEVELOPMENT IN INFORMATION RETRIEVAL, 2001.

NEWMAN, M. E. J. **Who is The Best Connected Scientist?** A Study of Scientific Coauthorship Networks. 2000.

NG, Andrew Y.; ZHENG, Alice X.; JORDAN, Michel L. Stable algorithms for link analysis. In: 24th INTERNATIONAL CONFERENCE ON RESEARCH AND DEVELOPMENT IN INFORMATION RETRIEVAL (SIGIR), 2001. **Proceedings...** 2001.

NIEDERAUER, C. A. P. **Ethos:** um modelo para medir a produtividade relativa de pesquisadores baseado na Análise por Envoltória de Dados 2002. Tese (Doutorado em Engenharia de Produção) – Programa de Pós-Graduação em Engenharia de Produção, Universidade Federal de Santa Catarina, Florianópolis, 2002.

OLIVEIRA, João Batista Araujo e. **A empresa inteligente:** organizações competitivas são as que aprendem a aprender. Serão as únicas a sobreviver no século XXI. Porto Alegre: Ortiz, 1992.

PACHECO, Roberto C. S. **Uma Metodologia de Desenvolvimento de Plataformas de Governo para Geração e Divulgação de Informações e de Conhecimento.** Artigo apresentado em cumprimento a requisito parcial de concurso para professor no INE/UFSC. Florianópolis, 2003. 35 p.

PACHECO, R. C. S.; KERN, V. M. Uma ontologia comum para a integração de bases de informação e conhecimento sobre ciência e tecnologia **Ciência da Informação**, v. 30, n. 3, p. 56-63, set./dez. 2001.

PINHEIRO, José C.; SUN, Don X. Methods for Linking and Mining Massive Heterogeneous Databases. In: KDD98.

QUEIRÓS, M. L. **Avaliação de planos de governo:** os planos plurianuais analisados segundo a formação de egressos, no âmbito do CNPq 2001. Dissertação (Mestrado em Engenharia de Produção) – Programa de Pós-Graduação em Engenharia de Produção, Universidade Federal de Santa Catarina, Florianópolis, 2001.

ROMÃO, W. **Descoberta de Conhecimento Relevante em Bancos de Dados Sobre Ciência e Tecnologia**. 2002. Tese (Doutorado em Engenharia de Produção) – Programa de Pós-Graduação em Engenharia de Produção, Universidade Federal de Santa Catarina, Florianópolis, 2002.

ROSEAUX. **Exercices et Problemes Résolus de Recherche Opérationnelle** – Graphes, Leurs Usages, Leurs Algorithmes, Masson, 1991.

ROY, B. **Algèbre Moderne et Théorie des Graphes.** Dunod, Paris, 1969.

SCOTT, J. **Social Network Analysis:** A Handbook. 2. ed. London: Sage Publications, 2000.

SABBATINI, M. **Lattes, cómo gestionar la ciência brasileña en la red.** Disponível em: <http://www.galeon.com/divulcat/articu/141a.htm>. Acesso em: 22 set. 2003.

SCHREIBER, Guus et al. **Knowledge Engineering and Management:** the CommonKADS Technology. MIT Press. 2002. IBSN: 0262193000.

STELA, Grupo. Histórico SCienTI. **Revista SCienTI**, v. 1, n. 1, p. 6-7, dez. 2002.

STUDER, R. et al. Situation and Prospective of Knowledge Engineering In: CUENA, J. et al. (Ed.). **Knowledge Engineering and Agent Technology**. IOS Series on Frontiers InArtificial Intelligence and Applications IOS Press, 2000.

SYSLO, M. M.; DEO, N.; KOWALIK, J. S. **Discrete Optimization Algorithms.** Prentice-Hall, 1983.

SVG (2004). **Scalable Vector Graphics**. Disponível em <http://www.w3c.org/Graphics/SVG>. Acesso em: 23 jun. 2004.

WASSERMAN, S.; FAUST, K. **Social Network Analysis.** Cambridge: Cambridge University Press, 1994.

WEST, Douglas Brent. **Introduction to Graph Theory**. Prentice-Hall, Inc., 1996.

WILSON, R. J.; BEINEKE, L. W. (Ed.). **Applications of Graph Theory.** Academic Press, 1979.

UNIVERSIDADE FEDERAL DE SANTA CATARINA. Projeto do Programa de Pós-Graduação em Engenharia e Gestão do Conhecimento. Florianópolis, 2003.

VISUALINKS (2004). **Visual Analytics Inc**. Disponível em <http://www.visualanalytics.com/Products/index2.cfm?Template=Visualinks>. Acesso em: 23 jun. 2004.

XU, Jennifer J.; CHEN, Hsinchun. **Using Shortest Path Algorithms to Identify Criminal Associations**. 2002.

ZHOU, Zhi-Hua. Three Perspectives of Data Mining. **Artificial Intelligence**, n. 143, p. 139-146, 2003.