UNIVERSIDADE FEDERAL DE SANTA CATARINA

TECNOLOGIAS DA INFORMAÇÃO E COMUNICAÇÃO

THALES DO NASCIMENTO DA SILVA

UMA ARQUITETURA PARA DESCOBERTA DE CONHECIMENTO A PARTIR DE BASES TEXTUAIS

Araranguá, 10 de julho de 2012

THALES DO NASCIMENTO DA SILVA

UMA ARQUITETURA PARA DESCOBERTA DE CONHECIMENTO A PARTIR DE BASES TEXTUAIS

Trabalho de Curso submetido à Universidade Federal de Santa Catarina como parte dos requisitos necessários para a obtenção do Grau de Bacharel em Tecnologias da Informação e Comunicação. Sob a orientação do Professor Alexandre Leopoldo Gonçalves.

Araranguá, 2012

THALES DO NASCIMENTO DA SILVA

UMA ARQUITETURA PARA DESCOBERTA DE CONHECIMENTO A PARTIR DE BASES TEXTUAIS

Trabalho de Conclusão de Curso submetido à Universidade Federal de Santa Catarina, como parte dos requisitos necessários para a obtenção do Grau de Bacharel em Tecnologias da Informação e Comunicação.

Professor Alexandre Leopoldo Gonçalves, Dr.
Presidente da Banca - Orientador

Professora Luciana Bolan Frigo, Dr.a
Membro

Professor Anderson Luiz Fernandes Perez, Dr.
Membro

Professor Giovani Mendonça Lunardi, Dr.
Membro Suplente

Araranguá, SC, 10 de julho de 2012.

Dedico este trabalho a todos que contribuíram direta ou indiretamente em minha formação acadêmica.

AGRADECIMENTOS

Agradeço a todos que contribuíram no decorrer desta jornada, em especialmente:

A Deus, a quem devo minha vida.

A minha família que sempre me apoiou nos estudos e nas escolhas tomadas.

A Pâmela por sempre me incentivar e compreender nos momentos difíceis.

Ao orientador Prof. Alexandre Leopoldo Gonçalves que teve papel fundamental na elaboração deste trabalho.

Aos meus colegas pelo companheirismo e disponibilidade para me auxiliar em vários momentos.

RESUMO

Atualmente, o volume de informação gerado aumenta exponencialmente, sendo que uma parcela significativa das informações encontra-se em formato textual. A partir desse formato é possível extrair determinados conhecimentos. Entretanto, face ao grande volume de informações disponíveis, seja na web ou mesmo nas organizações, tal tarefa constitui-se como um desafio computacional. Superado os obstáculos, o conhecimento obtido através de informações textuais pode ser utilizado na tomada de decisão com o intuito de gerar vantagem competitiva. Um dos meios de se extrair conhecimento é através da utilização do processo de Descoberta de Conhecimento em Bases de Dados e, no caso de informações textuais, através do processo de Descoberta de Conhecimento em Textos. De maneira geral, os processos de descoberta de conhecimento tradicionais são custosos quando aplicados em grandes coleções de documentos, por exemplo, a web. Com este pressuposto é proposto neste trabalho uma arquitetura para descoberta de conhecimento a partir de bases textuais almejando sua utilização em grandes fontes de informação. Para atingir este objetivo, a proposta utiliza, além da computação distribuída visando o aumento de desempenho, um modelo com base no conceito de correlação rápida. A demonstração de viabilidade é realizada através de um protótipo que implementa a arquitetura proposta. O protótipo tem a capacidade de gerar informações que relacionam padrões textuais (termos) e de permitir uma visão da evolução temporal em determinado domínio de problema. A aplicação do protótipo em um cenário possibilitou demonstrar que a arquitetura proposta é capaz de obter resultados consistentes e satisfatórios, tanto para o entendimento de determinado domínio, quanto para a análise de grandes bases textuais.

Palavras-chave: Descoberta de Conhecimento; Bases Textuais; Correlação de Informação; Computação Distribuída.

ABSTRACT

Currently the amount of information increases exponentially in which a great portion of these information is in textual format. From this format is possible to extract knowledge. However, considering the huge volume of information available, either the web or even in organizations, this task can be seen as a computational challenge. The knowledge acquired through textual information, once overcome the obstacles, can be used in decision making process aiming to generate competitive advantage. This can be done through Knowledge Discovery in Text. In general, traditional knowledge discovery processes are expensive when applied to large corpus, for instance, the web. Taken it into account is proposed in this work an architecture for knowledge discovery from textual databases aiming its use in large sources of information. Aiming to achieve the main objective this work focus on distributed computing in order to increase performance and on a fast correlation based model. The feasibility is demonstrated through a prototype implemented using the proposed architecture. The prototype has proved the ability to extract information by linking textual patterns (terms) and by allowing a temporal view in a given domain. The application of the prototype in a scenario has demonstrated that the proposed architecture is able to obtain consistent and satisfactory results.

Keywords: Knowledge Discovery; Text Databases; Information Correlation; Distributed Computing.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

API – *Application Program Interface*

CPU – *Central Processing Unit*

DBL – Descoberta Baseada em Literatura

DW – *Data Warehouse*

HTTP – *Hypertext Transfer Protocol*

IP – *Internet Protocol*

JDBC – *Java Database Connectivity*

JSON–*JavaScript Object Notation*

KDD – *Knowledge Discovery in Databases*

KDT – *Knowledge Discovery in Texts*

MD – Mineração de Dados

MIMD – *Multiple Instruction Multiple Data*

MT – Mineração de Textos

OWL – *Web Ontology Language*

PLN – Processamento de Linguagem Natural

RAM – *Random Access Memory*

RDF – *Resource Description Framework*

SETI – *Search ExtraterresTrial Intelligence*

SPARQL – *SPARQL Protocol and RDF Query Language*

UCP – Unidade Central de Processamento

URL – *Uniform Resource Locator*

XML – *eXtensible Markup Language*.

SUMÁRIO

1. INTRODUÇÃO

A evolução das tecnologias da informação vem promovendo diversas mudanças na sociedade em geral. Entre elas está a disponibilização de uma quantidade cada vez mais crescente de informações, resultado principalmente do aumento da capacidade de processamento e armazenamento. Este fenômeno torna-se cada vez mais evidente e vem sendo observado por diversos estudiosos da área.

Em 2003 o mundo produzia entre um e dois exabytes de informação nova por ano, ou seja, algo em torno de 250 megabytes para cada habitante na Terra (LYMAN; VARIAN, 2003). Um exabyte equivale a pouco mais de um bilhão de gigabytes. Estima-se que documentos impressos, que eram o meio mais comum de informação textual há algumas décadas, hoje representem apenas 0,003% da informação gerada anualmente (LYMAN; VARIAN, 2003).

O suporte ao aumento de informação é possivel graças a evolução dos meios de armazenamento magnéticos. Segundo Hilbert (2011), em 2000 os meios de armazenamento magnéticos representavam 5% da capacidade mundial, saltando para 45% em 2007, e a capacidade de armazenamento per capita que era de 2.866 megabytes em 1993, passou a ser de 44.716 megabytes em 2007.

Parte considerável dessa informação encontra-se na forma de textos nos mais diversos formatos. Desde a década de noventa estudos como os de Wilks e Catizone (1999) já apontavam que 80% da informação encontrava-se na forma textual. A cada ano são produzidos aproximadamente 968 mil livros, 80 mil revistas, 40 mil periódicos, bilhões de documentos (LYMAN; VARIAN, 2003). Além das fontes já citadas, redes sociais, wikis, e blogs também podem, e dependendo do foco de análise, devem ser consideradas como

importantes fontes de informação textual devido principalmente a sua dinamicidade. Weiss (2005) afirma ainda que informações textuais em linguagem natural são importantes fontes de informação, e que na maioria das vezes são ignoradas pelas organizações. “Se por um lado essa situação propicia muitas oportunidades de uso dessa informação para a tomada de decisão, por outro, lança muitos desafios em como armazenar, recuperar e transformar essa informação em conhecimento” (BOVO, 2011).

Um dos objetivos da análise da informação é a possibilidade de gerar conhecimento. Segundo Tuomi (1999) o caminho para o conhecimento é hierárquico, isto é, primeiramente são produzidos os dados (simples fatos) e em seguida, quando estruturados, são transformados em informação. A informação se torna conhecimento quando é interpretada, aplicada em um contexto, ou quando se adiciona significado a mesma. O conhecimento nos possibilita direcionar ações, tomar decisões, agir em determinadas situações (SCHREIBER et al., 2002).

Autores como Wilson (2002) e Alavi et al. (2001) afirmam que o conhecimento envolve processos mentais, entendimento e aprendizagem, e como tal, reside somente na mente das pessoas. Afirmam ainda que tudo aquilo que se utilize para expressar algo é realizado por meio de mensagens, desse modo, não constitui conhecimento e sim informação. Entretanto, outros autores consideram que o conhecimento pode ser explicitado (NONAKA; TAKEUCHI, 1995; SCHREIBER et al., 2002). Para Gonçalves (2006), não o conhecimento em si, mas redes de relacionamento, regras, padrões, tendências, entre outros, podem ser explicitados e se constituem, portanto, em ativos de conhecimento. Esses ativos podem então, através de ferramental adequado, serem descobertos visando auxiliar em processos de tomada de decisão.

Entre as possibilidades de ferramental visando identificar tais ativos a partir das informações geradas em determinado domínio encontram-se os processos de Descoberta de Conhecimento em Bases de Dados (*Knowledge Discovery in Database* - KDD) e de Descoberta de Conhecimento em Texto (*Knowledge Discovery in Texts* - KDT).

O processo de “KDD é o termo utilizado para promover a descoberta de conhecimento em bases de dados, e assim identificar e descrever os relacionamentos implícitos entre as informações nos bancos de dados em sistemas de uma organização” (SILVA; ROVER, 2011).

Considerando o processo de KDT este é similar ao KDD, porém trabalha com um *corpus* (coleção de documentos) em linguagem natural, buscando padrões e tendências, classificando e comparando documentos (SILVA; ROVER, 2011). Apesar do objetivo em comum, a descoberta de conhecimento, o KDT e o KDD, possuem diferenças importantes. A principal delas refere-se ao tipo de informação uma vez que KDT trabalha com informações textuais (não estruturadas ou semiestruturadas), enquanto que o KDD trabalha com informações estruturadas, geralmente obtidas a partir de bancos de dados relacionais e/ou orientado a objetos.

Os processos de descoberta de conhecimento são considerados não triviais, pois possuem diversas etapas compostas por algoritmos complexos, e trabalham com grande quantidade de informação, estruturada ou não. Esse fato se constitui em desafio uma vez que tais processos, quando executados a partir de uma infraestrutura computacional inadequada, podem inviabilizar a obtenção de resultados satisfatórios. Neste cenário, a computação distribuída desempenha um importante papel, e se torna uma solução viável no processamento de grande volume de informação. Segundo (TANENBAUM; STEEN, 2007) com a computação distribuída é possível utilizar um conjunto de computadores independentes, que na visão do usuário comportam-se como um sistema único e coerente. A principal motivação para a utilização de sistemas é a possibilidade de compartilhar recursos, tais como: componentes de hardware, discos, arquivos e bancos de dados (COULOURIS; DOLLIMORE; KINDBERG, 2005).

Desse modo, a utilização da computação distribuída é uma solução plausível para tratar o crescente incremento no volume de informação, pois possibilita o desenvolvimento de sistemas capazes de analisar grandes fontes de informação com o objetivo de extrair ativos de conhecimento capazes de auxiliar no processo de tomada de decisão.

1.1 PROBLEMA

A popularização da internet e a evolução constante dos recursos computacionais podem ser consideradas como as grandes responsáveis pelo crescente volume de informação, tanto na Web quanto nas organizações. A maior parte desta informação encontra-se no

formato não estruturado. De modo geral, este tipo de informação refere-se a documentos textuais como: e-mails, sites, artigos, teses, relatórios, ou seja, textos em linguagem natural.

Utilizando o processo de Descoberta de Conhecimento em Texto (KDT) é possível gerar conhecimento a partir de bases textuais. Contudo, este processo é custoso e não trivial devido principalmente à natureza da informação, ou seja, não possui estrutura e está sujeita a ambiguidade.

A correlação de informação é um dos métodos que podem ser utilizado no processo de KDT. Porém, métodos tradicionais de correlação que inspecionam as relações entre padrões textuais em cada documento quando aplicada em grandes bases de informação podem inviabilizar a utilização dessa abordagem. Visto que, em muitos casos, o que se deseja é um indicativo de determinado comportamento para posterior análise mais apurada, torna-se útil, ainda que com um erro implícito, um método capaz de correlacionar padrões utilizando grandes fontes de informação como a Web.

Desse modo tem-se como pergunta de pesquisa “Como elaborar uma arquitetura voltada à descoberta de conhecimento que, em conjunto com a computação distribuída, torne viável o estabelecimento de relações temporais entre padrões textuais a partir de grandes fontes de informações?”.

1.2 OBJETIVOS

1.2.1 Objetivo Geral

Propor uma arquitetura computacional com base na computação distribuída capaz de lidar com grandes bases de informação textual visando auxiliar o processo de Descoberta de Conhecimento em Texto.

1.2.2 Objetivos Específicos

Visando atingir o objetivo principal, alguns objetivos específicos são requeridos, entre eles:

- Analisar o panorama atual dos processos de descoberta de conhecimento, detalhando as fases que os compõem, assim como, a área de computação distribuída.

- Modelar uma base de dados que suporte os dados gerados a partir do processo de descoberta de conhecimento e permita representar a característica temporal das bases textuais.

- Desenvolver um protótipo da arquitetura voltado à descoberta de conhecimento que, de maneira distribuída, consiga manipular grandes bases textuais. .

- Realizar uma discussão dos resultados oriundos do processo de descoberta de conhecimento.

1.3 METODOLOGIA

O trabalho será desenvolvido com base um uma pesquisa exploratória através do desenvolvimento de um protótipo de descoberta de conhecimento a partir de bases textuais. A metodologia de desenvolvimento deste trabalho é dividida em três etapas:

Etapa 1: análise da literatura focando nas seguintes áreas: descoberta de conhecimento e computação distribuída.

Etapa 2: proposição da arquitetura lógica e física com suporte à execução distribuída.

Etapa 3: modelagem da camada de persistência voltada ao processo descoberta de conhecimento em bases textuais

Etapa 3: desenvolvimento da camada de persistência e do protótipo de descoberta de conhecimento com base na arquitetura proposta anteriormente.

Etapa 4: testes do protótipo considerando um cenário de uso.

Etapa 5: avaliação dos resultados obtidos através da utilização da arquitetura de KDT proposta.

1.4 ORGANIZAÇÃO DO TEXTO

O documento esta dividido em 6 capítulos. No primeiro capítulo apresenta-se o projeto, expondo uma breve contextualização e apresentando a problemática vislumbrada, assim como os objetivos geral e específicos.

No segundo capítulo é realizada uma revisão sobre a área de Descoberta de Conhecimento promovendo um maior detalhamento do processo da Descoberta de Conhecimento em Texto.

O terceiro capítulo faz uma revisão da literatura relacionada à área de Computação Distribuída, abordando conceitos de sistemas com múltiplos processadores e computação de alto desempenho.

O quarto capítulo propõe uma arquitetura de descoberta de conhecimento a partir de bases textuais aliada à computação distribuída. Este capítulo divide-se em duas partes, sendo: (a) um detalhamento do modelo lógico da arquitetura; e (b) uma descrição dos componentes tecnológicos e serviços da arquitetura (modelo físico).O quinto capítulo apresenta e discute os resultados obtidos assim como as possibilidades de análise considerando a proposição do trabalho.

Por fim, o sexto capítulo contém as considerações finais e os trabalhos futuros.

2. PROCESSOS DE DESCOBERTA DE CONHECIMENTO

2.1 DESCOBERTA DE CONHECIMENTO

Atualmente o número de informação gerada vem aumentando exponencialmente, esta informação, se tratada corretamente pode ser uma grande aliada na tomada de decisão dentro das organizações. Tuomi (1999) afirma que dados podem ser considerados como simples fatos, que, quando estruturados tornam-se informação. A informação torna-se conhecimento quando é interpretada, inserida em um contexto, ou quando é acrescentado um significado a ela. Com este presuposto pode-se afirmar que dado é um pré-requisito para a informação, e a informação é necessária para a geração de conhecimento.

A informação pode ser encontrada em três estados: estruturada, onde cada campo possui a identificação da informação (Banco de Dados, Planilha de Textos); semiestruturada, possui tags que possibilitam a marcação das informações (XML, RDF); ou não estruturadas, que são textos em linguagem natural. “Apesar de um texto em linguagem natural ser estruturado no sentido de possuir uma estrutura sintática, a referência a “estrutura” é feita no âmbito da Ciência da Computação” (BOVO, 2011). As informações não estruturadas podem ser encontradas em artigos, atas, sites, e-mails, ou seja, qualquer documento escrito em linguagem natural. Uma organização gera diversos documentos não estruturados, que contêm informações importantes sobre a realidade da organização, estes documentos muitas vezes são ignorados, quando poderiam auxiliar no processo de tomada de decisão.

O aumento da quantidade de informação gerada está em evidencia uma vez que diversos estudiosos analisam este fenômeno e meios de extrair conhecimento de toda esta informação. “A velocidade e a amplitude com que o conhecimento gerado passou a ser

compartilhado provocaram o surgimento de uma dinâmica de reaproveitamento e produção de novos conhecimentos, bem como o aparecimento de novas necessidades de tratar a informação" (RAMOS; BRASCHER, 2009).

A partir da necessidade de uma análise mais apurada da informação gerada surgiu na década de 90 o conceito de descoberta de conhecimento e os processos que possam conduzir a isso. Estes processos evidenciam informações que provavelmente não seriam observadas sem a utilização dos mesmos.

A descoberta de conhecimento pode ser dividida em duas vertentes: KDD e KDT. Esta divisão tem como base o conteúdo que será analisado, em que, se o conteúdo foi previamente organizado e estruturado o processo de descoberta utilizado será o KDD. Caso o conteúdo encontre-se disperso em documentos textuais o processo utilizado será o KDT (RAMOS, BRASCHER; 2009).

Os processos de descoberta de conhecimento são compostos por várias fases, sendo que cada fase possui diversas tarefas a serem executadas. Uma tarefa é resolvida através da escolha de uma técnica de resolução. Por fim, as técnicas de resolução utilizam algoritmos, podendo haver mais de um algoritmo que possa ser aplicado a uma determinada técnica.

O processo de KDD, ou descoberta de conhecimento em banco de dados, pode ser definido como "... um processo, não trivial, de identificar novos, válidos e potencialmente úteis padrões nos dados" (FAYYAD et al., 1996). Em suma o principal objetivo do KDD é a tradução de dados brutos em informações relevantes (VIANNA et al., 2010). A Figura 1 ilustra o fluxo do processo de KDD.

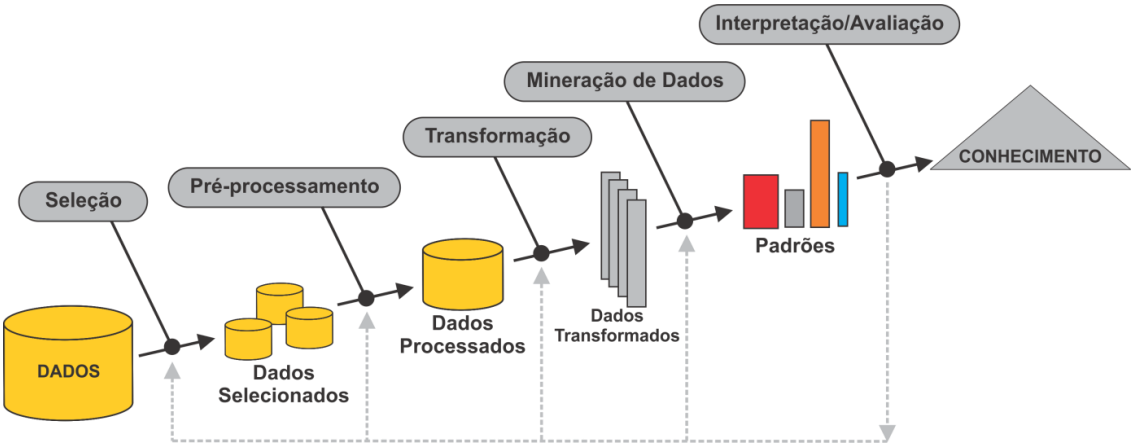

Figura 1 - Uma visão geral do processo de KDD.
Fonte: adaptado de (FAYYAD, 1996)

Como pode ser observado o KDD é um processo iterativo no qual todas as fases são importantes para se atingir o objetivo (SILVA, 2004). As fases tradicionais do processo de KDD são:

Seleção de Dados: Nesta fase são selecionados os dados pertinentes ao domínio do problema, em que fica evidente a necessidade da compreensão do domínio e dos objetivos (SILVA, 2004). Este processo de seleção é realizado utilizando-se de um banco de dados estruturado.

Pré-processamento: Esta etapa visa "eliminar os dados incompletos, problemas de definição de tipos, eliminação de tuplas repetidas, etc" (BARION; LAGO, 2008). De forma resumida esta etapa realiza pequenas correções e limpeza no banco de dados visando garantir a consistência e a exclusão de dados desnecessários.

Transformação: Após o pré-processamento a etapa de transformação é responsável por realizar a persistência dos dados tratados, deixando-os prontos para a mineração de dados. A transformação está diretamente ligada à técnica de mineração de dados. Segundo Barion e Lago (2008), o principal objetivo desta fase é "... facilitar a utilização das técnicas de mineração de dados".

Mineração de Dados: Segundo Fayyad et al., (1996) esta fase busca por padrões através de tarefas como: regras de classificação ou árvores, regressão, agrupamento, entre outros. Essas análises são geralmente executadas sobre uma grande quantidade de dados pré-processados e armazenados por *Data Warehouses*/*Data Marts* (GONÇALVES, 2006).

Interpretação/Avaliação: Esta fase apresenta o resultado da descoberta de conhecimento para o usuário por meio de visualização e representação do conhecimento obtido durante o processo. "Os resultados do processo de descoberta do conhecimento podem ser mostrados de diversas formas, porém devem ser apresentadas de forma que o usuário possa entender e interpretar os resultados obtidos" (BARION; LAGO, 2008).

O processo de KDD pode ser um grande aliado na busca por conhecimento, com o objetivo de obter alguma vantagem competitiva nas organizações. Porém, este processo não é

capaz de extrair conhecimento de informações não estruturadas. As informações não estruturadas possuem um grande potencial, pois é neste formato que se encontram a maioria das informações na atualidade.

Com objetivo de preencher esta lacuna surge o processo de descoberta de conhecimento em textos, ou KDT. O KDT é semelhante ao KDD, porém "baseia-se em técnicas específicas para tratamento de textos que devem ser utilizadas a fim de se obter conhecimentos implícitos em banco de dados textuais" (BARION; LAGO, 2008). A Figura 2 ilustra o fluxo do processo de KDT.

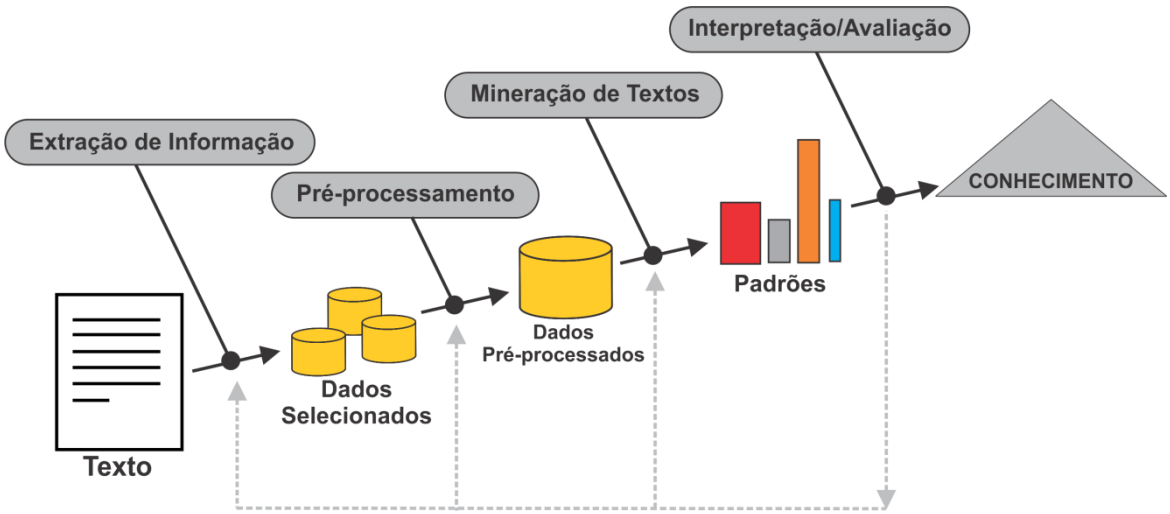

Figura 2 - Uma visão geral do processo de KDT.
FONTE: Adaptado de: MOONEY; NAHM, 2005.

O processo de KDT é semelhante ao KDD inclusive nas fases de mineração e interpretação/avaliação. Apesar disto, os passos do KDT possuem pequenas adaptações para que possa ser aplicado em informações não estruturadas. As principais diferenças ocorrem nos seguintes passos:

Extração de Informação: Nesta etapa são selecionados os textos em geral de acordo com o domínio do problema. "Considerando os objetivos que se deseja alcançar com o processo, o primeiro passo é eleger o conjunto de textos que será utilizado" (CECI et al., 2010).

Pré-processamento: "O objetivo desta fase é eliminação de termos não relevantes (stop-words), redução das palavras aos seus radicais (stemming), correções ortográficas e outros aspectos morfológicos e também sintáticos que as expressões textuais possuem."

(CECI et al., 2010). Nesse sentido, o Processamento de Linguagem Natural (PLN) é fundamental nesta fase, pois trata exatamente da descoberta destas estruturas implícitas, como por exemplo, a estrutura sintática (RAJMAN; BESANÇON, 1997). Percebe-se que o processo de KDD ilustrado na Figura 2 define a transformação de dados como uma etapa, já no processo de KDT a transformação dos dados esta implícita na etapa de pré-processamento segundo a proposta de Mooney e Nahm (2005). De acordo com Ceci et al., (2010), no pré-processamento ocorre à transformação do texto em vetores, tabelas, matrizes, etc. A partir deste ponto a informação encontra-se estruturada, e pronta para a fase de mineração.

Este processo de conversão de textos para informação estruturada pode promover custos adicionais ao processo de KDT em relação ao KDD. Esta conversão pode ser observada na Figura 3.

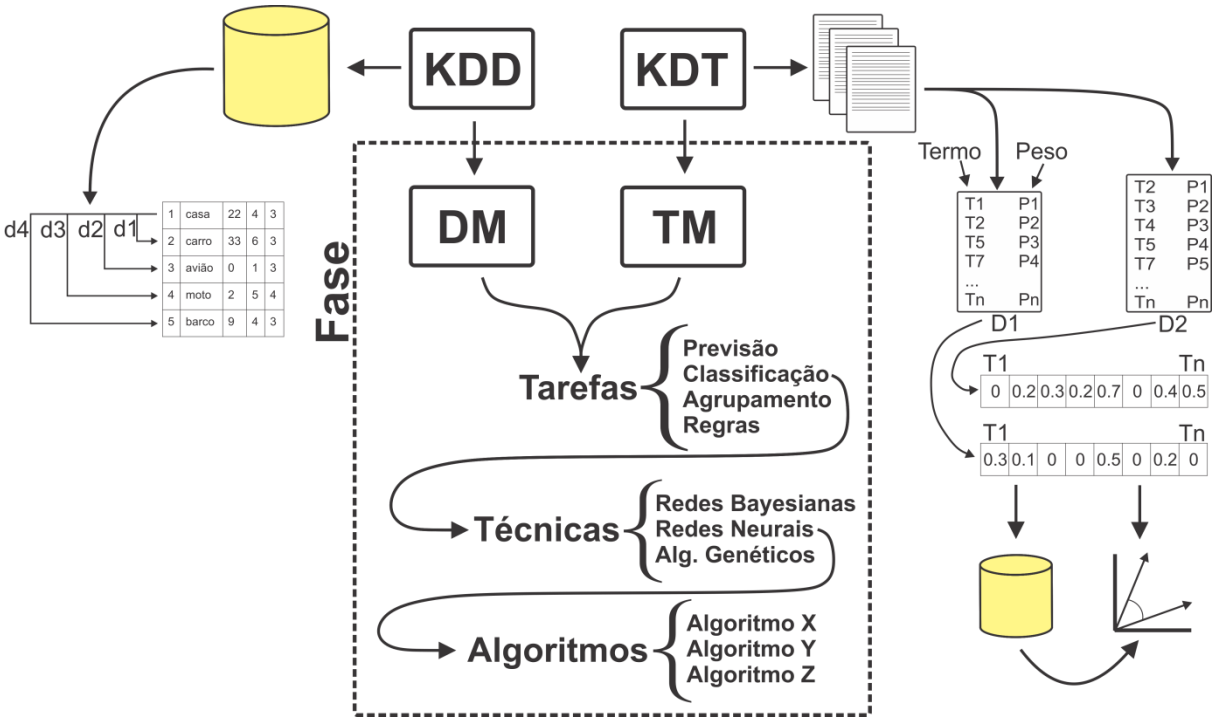

Figura 3 - Detalhamento e diferenciação dos processos de KDD e KDT.

Como pode ser observado na Figura 3 a fase de mineração de dados necessita de dados estruturados. No exemplo exposto, para cada documento é criado um vetor, sendo que o tamanho deste vetor é definido pelo documento com maior número de termos. Cada posição do vetor deve guardar um termo e o peso desse termo referente à sua importância no documento. Uma possível solução para evitar a padronização no tamanho do vetor (tamanho 1) o que aumenta o custo computacional é a utilização de tabelas *Hash*. Nesse sentido,

comparações vetoriais são otimizadas, uma vez que, considerando dois vetores quaisquer basta realizar uma varredura no menor vetor para descobrir os termos semelhantes em relação ao segundo vetor (vetor maior).

2.1.1 MODELOS BASEADOS EM COOCORRÊNCIA

Como citado anteriormente o processo de KDT necessita de um método para agregar a informação textual. A correlação pode ser utilizada para este fim. Através de cálculos oriundos da estatística é possível encontrar um grau de relação entre duas variáveis. Nesse sentido, o resultado da correlação é um conjunto de dados quantitativos que permitem estabelecer a força de conexão entre as variáveis (dois termos no contexto de KDT).

Segundo Stevenson (2001) "O objetivo do estudo correlacional é a determinação da força do relacionamento entre duas observações emparelhadas. O termo "correlação" significa literalmente "co-relacionamento", pois indica até que ponto os valores de uma variável estão relacionados com os de outra.". Ainda segundo Lira (2004) a partir da correlação é possível encontrar o grau de relacionamento entre duas variáveis.

Uma função de correlação gera como resultado o grau de correlação ou coeficiente de correlação. O conjunto de coeficientes de correlação forma uma matriz de correlação TERMOxTERMO, ou seja, uma matriz que contém em cada célula o valor da correlação entre dois termos quaisquer.

A Tabela 1 exemplifica uma matriz de correlação. A partir da matriz é possível localizar o peso de qualquer relação. Nesse sentido, a letra *w* representa o coeficiente de correlação entre dois termos, sendo que o coeficiente é calculado através de um método baseado em coocorrência.

TERMOS	t1	t2	t3	t4	t5	t6	t7
t1	-	$w_{(t1,t2)}$	$w_{(t1,t3)}$	$w_{(t1,t4)}$	$w_{(t1,t5)}$	$w_{(t1,t6)}$	$w_{(t1,t7)}$
t2	$w_{(t2,t1)}$	-	$w_{(t2,t3)}$	$w_{(t2,t4)}$	$w_{(t2,t4)}$	$w_{(t2,t6)}$	$w_{(t2,t7)}$
t3	$w_{(t3,t1)}$	$w_{(t3,t2)}$	-	$w_{(t3,t4)}$	$w_{(t3,t5)}$	$w_{(t3,t6)}$	$w_{(t3,t7)}$
t4	$w_{(t4,t1)}$	$w_{(t4,t2)}$	$w_{(t4,t3)}$	-	$w_{(t4,t5)}$	$w_{(t4,t6)}$	$w_{(t4,t7)}$
t5	$w_{(t5,t1)}$	$w_{(t5,t2)}$	$w_{(t5,t3)}$	$w_{(t5,t4)}$	-	$w_{(t5,t6)}$	$w_{(t5,t7)}$
t6	$w_{(t6,t1)}$	$w_{(t6,t2)}$	$w_{(t6,t3)}$	$w_{(t6,t4)}$	$w_{(t6,t5)}$	-	$w_{(t6,t7)}$
t7	$w_{(t7,t1)}$	$w_{(t7,t2)}$	$w_{(t7,t3)}$	$w_{(t7,t4)}$	$w_{(t7,t5)}$	$w_{(t7,t6)}$	-

Tabela 1 - Matriz de correlação TERMOxTERMO com 7 termos.

A seguir são apresentados os principais modelos baseados em coocorrência segundo a visão de Gonçalves (2006), assim como os principais conceitos utilizados por esses modelos. Neste trabalho “palavras” e “termos” são entendidos como conceitos similares.

2.1.1.1 Frequência

Uma maneira simples, ainda que imprecisa, de se estabelecer a relação entre dois termos é através da frequência conjunta. Entretanto, considerar pares de termos frequentes sem o devido tratamento não é adequado. Isto porque em uma língua, artigos e preposições, por exemplo, aparecem constantemente ligados entre si ou com substantivos e verbos (ex: é para, a partir, abaixo são, entre outros, tecnologias da). Uma possível solução é a utilização de uma lista de termos indesejáveis (*stop list*) para retirar tais termos da análise.

2.1.1.2 Média e Variância

Este modelo permite encontrar relações de maneira mais flexível, pois considera a relação de palavras mesmo havendo uma distância distinta no texto (utiliza janelas). O cálculo deste modelo é realizado da seguinte maneira: primeiramente é calculada a média das distâncias em que as palavras ocorrem no texto com a seguinte equação, $\bar{d} = \frac{s}{n}$, sendo que, ***s*** representa a soma das distâncias, e ***n*** o número de coocorências das palavras. Nesse sentido, a variância informa o grau de desvio das distâncias a partir da média, sendo estimada conforme a seguinte equação, $S^2 = \frac{\sum_{i=1}^{n}(di - \bar{d})^2}{n-1}$, onde ***n*** é o número de vezes que as duas palavras coocorrem, ***di*** é a distância da *i*th coocorrência, e $\bar{d}$ é a média das distâncias. Caso as distâncias sejam sempre as mesmas, a variância será zero. Do contrário, se as distâncias acontecem aleatoriamente, ou seja, não configuram um padrão de relacionamento, a variância será alta. Finalmente, é realizado o cálculo de desvio padrão, $S = \sqrt{s^2}$, sendo que, valores de desvios altos indicam relacionamentos pouco relevantes.

2.1.1.3 Teste de hipótese

Avaliar se algo é ou não um evento ao acaso é um problema clássico da estatística chamado de teste de hipótese (MANNING; SCHÜTZE, 1999). Neste modelo, formula-se a *hipótese nula H0* em que não existe associação entre duas palavras, apenas ocorrências ao acaso. Desse modo, calcula-se a probabilidade *p* que o evento ocorreria se *H0* fosse verdadeira, e então se rejeita *H0* se *p* é muito baixa (normalmente valores abaixo de um nível de significância de $p < 0.05$, 0.01, 0.005, ou 0.001) e, caso contrário, se aceita *H0* como sendo possível (BOVO, 2011). Sendo assim, quando se rejeita uma hipótese nula, existe um relacionamento entre duas palavras. Caso contrário, ou seja, quando se aceita uma hipótese nula não há um relacionamento entra as palavras.

2.1.1.4 Teste de Pearson - *Chi-square* (x^2)

É uma técnica estatística utilizada para determinar se a distribuição das frequências observadas difere das frequências esperadas. Se a diferença entre as frequências observadas e esperadas é alta, então a hipótese nula de independência pode ser rejeitada. Isso significa que há uma relação entre os dois termos, e não apenas algo aleatório (BOVO, 2011). A aplicação do Teste de Pearson utiliza uma tabela 2*2 (tabela de contingência), como pode ser observado na Tabela 2.

	w_2	$\overline{w}_2$
w_1	a	b
$\overline{w}_1$	c	d

Tabela 2 - Tabela de contingência de 2x2.

sendo que *a* representa a quantidade de vezes em que w_1 e w_2 ocorrem conjuntamente, *b* indica a quantidade de ocorrências de w_1 sem a presença de w_2, *c* indica a quantidade de ocorrências de w_2 sem a presença de w_1, e *d* é o tamanho da coleção de documentos menos o número de documentos que não contenham w_1 e/ou w_2, sendo $d=N-a-b-c$, onde *N* é o tamanho da base (GONÇALVES, 2006).

2.1.1.5 Phi-squared(ϕ^2)

O phi-squared também utiliza uma tabela de contingência, similar ao método anterior. Segundo Conrad e Utt (1994), o Phi-squared tende a favorecer associações com alta frequência. O Phi-squared (CHURCH; GALE, 1991) é definido como:

$$\phi^2 = \frac{(ad - bc)^2}{(a + b)(c + d)(a + c)(b + d)}, \text{ onde } 0 \leq \phi^2 \leq 1.$$

2.1.1.6 Informação Mútua

Segundo Churche e Hanks (1990), a Informação Mútua compara a probabilidade de um par de palavras (ou qualquer outra unidade linguística) aparecer mais frequentemente de maneira conjunta do que isoladamente. Essa medida cresce à proporção que a frequência conjunta também cresce. Se uma determinada palavra tende a ocorrer individualmente, então o índice apurado através da Informação Mútua será um valor negativo.

2.1.2 ASSOCIAÇÃO DE ELEMENTOS TEXTUAIS

O relacionamento indireto de padrões tendo como fonte de informação conteúdo não estruturado foi utilizado primeiramente por Swanson em 1986 (SWANSON, 1986). Ao realizar uma revisão da literatura o pesquisador descobriu uma intervenção médica para a Doença de Raynaud através da análise de relacionamentos indiretos. A análise evidenciou uma relação entre "Doença de Raynaud" e "Alta Viscosidade do Sangue". Ao revisar o termo "Alta Viscosidade do Sangue", Swanson encontrou uma conexão com "Óleo de Peixe". A partir deste pressuposto Swanson criou a hipótese de que o Óleo de peixe poderia ser utilizado para o tratamento da Doença de Raynaud (BOVO, 2011). A partir desta experiência surgiram diversos estudos sobre associação de elementos textuais, em sua maioria na área da medicina.

A parir dos estudos de Swanson surgiu a Descoberta Baseada em Literatura (DBL). A DBL tem como objetivo encontrar relacionamentos indiretos em fontes de informações textuais. A descoberta baseada em literatura (DBL) é basicamente uma atividade de criação de hipóteses científicas por meio da busca de conexões entre estruturas de conhecimento disponíveis publicamente, porém inadvertidamente desconhecidas, ou seja, jamais citadas conjuntamente (TARDELLI, 2002).

Segundo Bovo (2011) a DBL utiliza métodos de MT para a descoberta de novos conhecimentos através de relacionamentos indiretos entre elementos textuais. A associação de elementos textuais ocorre quando entidades se relacionam de forma indireta. A associação entre elementos textuais é obtida através da análise dos relacionamentos indiretos, com base nos contextos em que eles aparecem nos documentos textuais (BOVO, 2011). A Figura 4 expõe exemplos de relacionamentos diretos e indiretos.

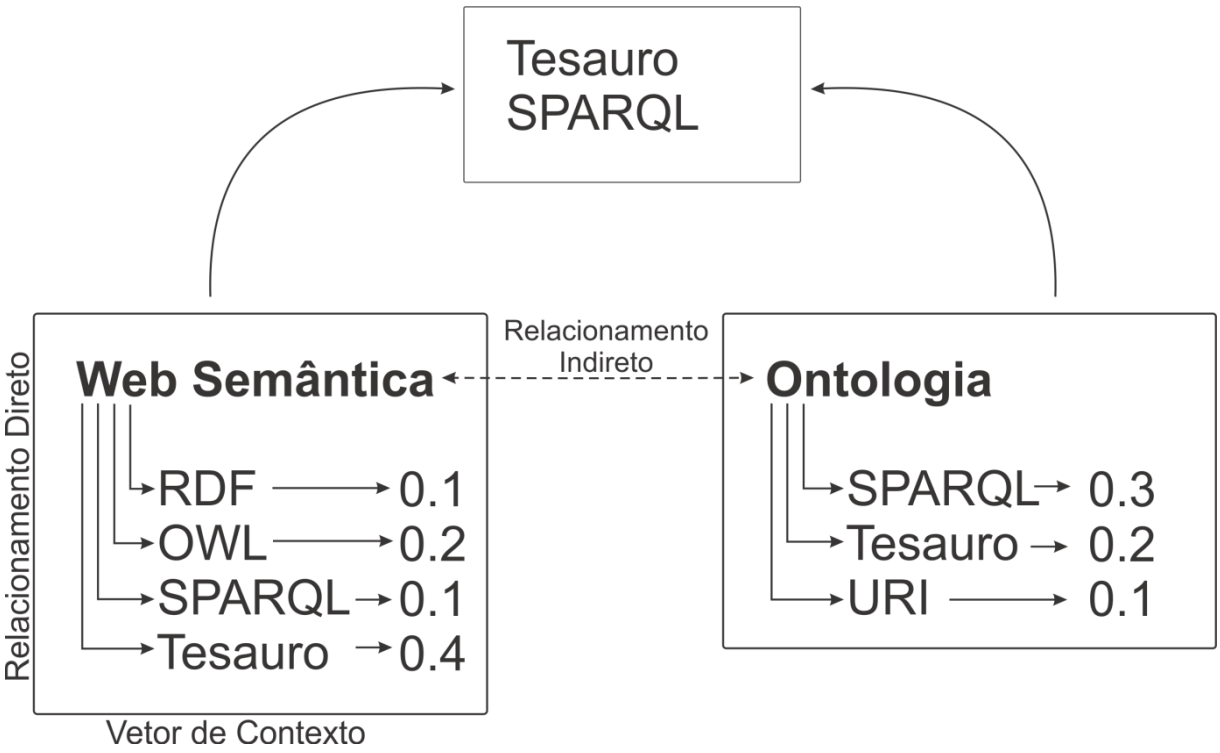

Figura 4 - Vetores de contexto de Web Semântica e Ontologia, relacionados indiretamente por Tesauro e SPARQL.

Os vetores baseados nos relacionamentos diretos são entendidos como vetores de contexto e são gerados através da correlação. Billhardt, Borrajo e Maojo (2002) afirmam que os vetores de contexto gerados através de um modelo baseado em coocorrência podem ser utilizados para estimar a força do relacionamento indireto entre termos, sendo geralmente aplicados a bases que tenho algum nível de semantica.

Primeiro, gerar vetores termcontext baseada na co-ocorrência de termos no mesmos documentos. Estes vectores são utilizados para calcular vectores de contexto para os documentos. Nós apresentamos diferentes técnicas para estimar as dependências entre os termos. Edição 3

A partir destes vetores é possível utilizar a DBL e evidenciar associações de elementos textuais. Segundo Gonçalves et al., (2005) a análise de relacionamentos indiretos podem revelar padrões mais complexos entre as entidades promovendo diferente perspectivas na análise de relações.

A Tabela 3 demonstra de forma resumida um exemplo de matriz TERMOxTERMO a partir da Figura 4 em que é possível extrair o vetor de contexto de cada termo. Os valores entre os termos são obtidos através de algum cálculo de coocorrência o que produzirá um peso (*w*) também chamado de coeficiente de correlação entre os termos.

		VETORES DE CONTEXTO						
		RDF	OWL	**Web Semântica**	SPARQL	**Ontologia**	Tesauro	URI
VETORES DE CONTEXTO	RDF	-	*w*	***0.1***	*w*	***0***	*w*	*w*
	OWL	*w*	-	***0.2***	*w*	***0***	*w*	*w*
	Web Semântica	***0.1***	***0.2***	***-***	***0.1***	***0***	***0.4***	***0***
	SPARQL	*w*	*w*	***0.1***	-	***0.3***	*w*	*w*
	Ontologia	***0***	***0***	***0***	***0.3***	***-***	***0.2***	***0.1***
	Tesauro	*w*	*w*	***0.4***	*w*	***0.2***	-	*w*
	URI	*w*	*w*	***0***	*w*	***0.1***	*w*	-

Tabela 3 - Vetores de Contexto.

O termo “Web Semântica” possui em seu vetor de contexto, outros termos que coocorrem com o mesmo, ou seja, os termos do vetor (RDF, OWL, Tesauro, SPARQL) possuem relacionamento direto com “Web Semântica”. Percebe-se que o termo “Ontologia” não coocorre com “Web Semântica”. Fica evidente que os termos “Web Semântica” e “Ontologia” não são citados em um mesmo documento. Com este cenário pode-se afirmar que não há relação direta entre os dois termos citados. Porém há uma relação indireta, que pode ser evidenciada através da associação de elementos textuais.

A busca pela associação analisa os dois vetores de contexto (Web Semântica e Ontologia), e busca por termos em comum. No caso do exemplo acima os dois vetores de contexto possuem os termos “Tesauro” e “SPARQL” em comum. Com este pressuposto pode-se afirmar que “Web Semântica” e “Ontologia” possuem relacionamento indireto.

2.1.2.1 Descoberta ABC

A descoberta ABC é uma técnica utilizada na DBL para descobrir associações entre elementos textuais. Segundo Bovo (2011) “esta técnica consiste em descobrir relacionamentos entre conceitos, que apesar de não coocorrerem estão conectados indiretamente por outros conceitos. É dividida em descoberta aberta e fechada”. Os dois modelos de descoberta ABC, serão apresentados a seguir.

2.1.2.1.1 Aberta

O modelo de descoberta aberta não tem um objetivo específico, ou seja, o modelo pode ser aplicado na tentativa de resolver um problema científico onde não há nenhuma ideia de onde a descoberta irá acabar. A descoberta aberta também pode ser utilizada para gerar novas hipóteses (WEEBER et al., 2001). A partir da Figura 5 é possível perceber que o processo de descoberta aberta não tem objetvo definido. Supondo que o A é uma doença, B é um sintoma, e o C é um tratamento, é possivel chegar ao tratamento para a doença sem nenhuma hipótese.

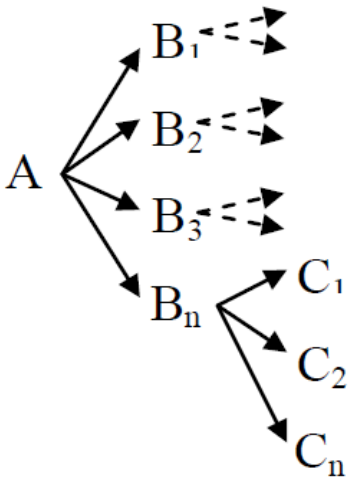

Figura 5 - Modelo de descoberta ABC aberta.
Fonte: GANIZ; POTTENGER; JANNECK, 2006**.**

O processo de descoberta aberta seleciona um termo origem (A) e partir dissso todos os documentos que contenham A são selecionados. Considerando os documentos selecionados, uma análise é realizada visando extrair termos considerados relevantes por especialistas de modo que se forme um conjunto com termos candidatos (B's). Em seguida os documentos que contenham B são analisados e os termos considerados relevantes por especialistas são extraidos, formandos os C's. A partir da relação direta entre A e B, e B e C é possivel afirmar que C e A possuem uma relação indireta, ou seja, não foram citadas conjuntamente em nenhum documento analisado. Ao final do processo é possivel gerar novas hipóteses, como observado no estudo inicial de Swanson.

2.1.2.1.2 **FECHADA**

Diferentemente do anterior, o modelo de descoberta fechada é iniciado com A e C conhecidos. Neste caso, a descoberta fechada limita-se em encontrar novos B's, a partir de hipóteses ou observações previamente concebidas. A Figura 6 exemplifica o modelo de descoberta fechada.

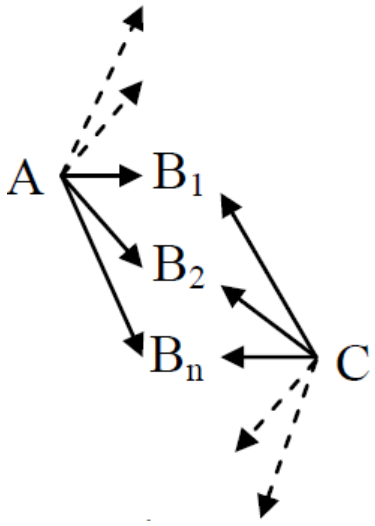

Figura 6 - Modelo de descoberta ABC fechada.
Fonte: GANIZ; POTTENGER; JANNECK, 2006**.**

O pressuposto deste modelo é o conhecimento do A e C. Pode-se afirmar que se uma comunidade científica sabe que B é um sintoma da doença C, e outra comunidade científica tem conhecimento que a substância A pode ser usada como tratamento para o sintoma B. É possivel chegar a uma ligação entre A e C através de B (WEEBER, 2003). Neste caso pode-se

afirmar que o modelo de descoberta fechada é mais adequado visto que este parte de uma hípotese que pode ou não ser validada.

2.1.3 MODELO VETORIAL

O modelo vetorial é utilizado para estabelecer a similaridade entre termos constantes em determinado vetor. No contexto do trabalho um vetor pode ser entendido com uma relação de termos relacionados a determinado termo de origem através da correlação entre eles. Sendo assim, a partir de determinado termo (origem) é obtido um vetor de contexto que contém os termos coocorrentes. Os diversos vetores de contexto formam uma matriz *m*x*m* que representa o peso entre os termos. Esta matriz pode ser observada na Tabela 3.

A partir de uma equação de similaridade é possível chegar ao grau de semelhança entre dois conjuntos, que é representado por um número positivo (EGGHE, MICHEL; 2002). Jones e Furnas (1987) afirmam que é possível aplicar a equação de similaridade a partir de um par de vetores. Com este pressuposto pode-se utilizar equações de similaridade a fim de encontrar o grau de semelhança entre os vetores de contexto. Dentre diversas equações de similaridade destacam-se: o índice Jaccard, índice Dice, medida *overlap* (máxima e mínima), medida do cosseno e medida do pseudo-cosseno (EGGHE; MICHEL, 2002; JONES; FURNAS, 1987).

Segundo Salton e Buckley (1988) "A equação do cosseno mede o ângulo entre dois vetores, variando de 1.0 (cos(0°) = 1.0) para vetores apontando na mesma direção, 0.0 (cos(90°) = 0.0) para vetores ortogonais e -1.0 (cos(180°) = -1.0) para vetores apontando em direções opostas". Esta equação pode ser definida como:

$$cos\theta = \frac{\sum_{i=1}^{n}(t1_i \times t2_i)}{\sqrt{\sum_{k=1}^{n}(t1_k)^2} \times \sqrt{\sum_{j=1}^{n}(t2_j)^2}}$$

onde *t*1 e *t*2 representam vetores de contexto, $t1_i$, $t2_i$, $t1_k$, e $t2_j$ representam a frequência individual ou o peso dos termos pertencentes aos vetores *t*1 e *t*2.

A Figura 7 representa o grau de similaridade entre vetores. Na Figura 7(a) os termos são poucos semelhantes, enquanto que a Figura 7(b) representa dois vetores mais similares.

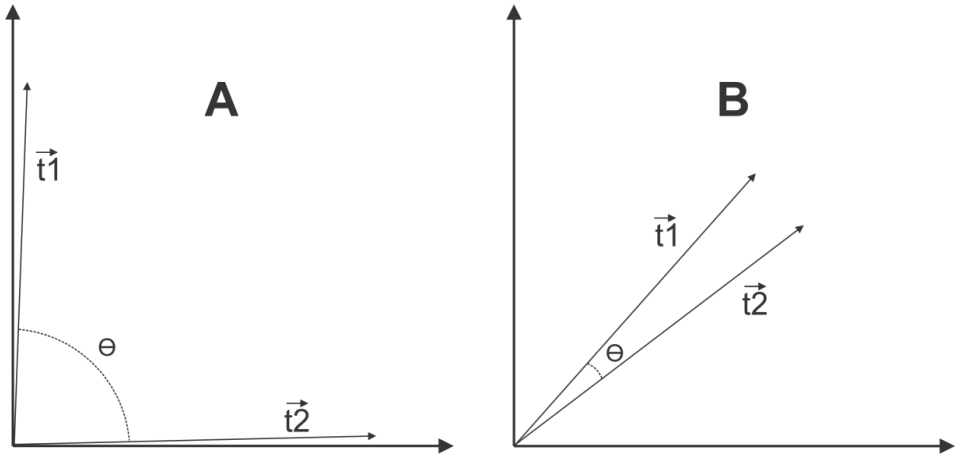

Figura 7 - Representação gráfica da similaridade de vetores;(a) representa vetores pouco similares e (b) representa vetores similares.

A Figura 8 apresenta o resultado da aplicação do cálculo de similaridade (cosseno) entre os termos "web semântica" e "ontologia". Os vetores de contexto destes termos podem ser observados na Tabela 3.

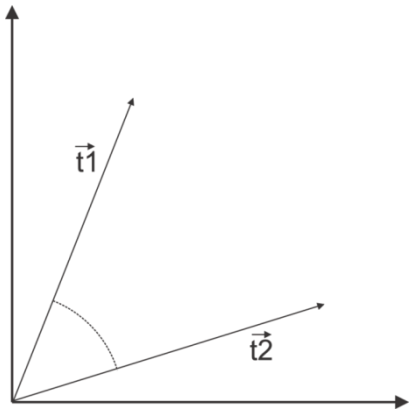

Figura 8 - Representação da similaridade dos vetores de contexto de Web Semântica e Ontologia**.**

O ângulo da Figura 8 foi gerado através da aplicação da equação do cosseno a partir dos vetores de contexto de "Web Semântica" e "Ontologia" apresentados na Figura 4. O cálculo pode ser observado a seguir:

$$\cos(\overrightarrow{We\ b\ Semântica}, \overrightarrow{Ontologia}) = \frac{\sum_{i=1}^{n}(t1_i \times t2_i)}{\sqrt{\sum_{j=1}^{n}(t1_j)^2} \times \sqrt{\sum_{k=1}^{n}(t2_k)^2}}$$

$$\sum_{i=1}^{n}(t1_i \times t2_i) = (0.1 \times 0) + (0.2 \times 0) + (0.1 \times 0.3) + (0.4 \times 0.2) + (0 \times 0.1) = 0.11$$

$$\sqrt{\sum_{j=1}^{n}(t1_j)^2} = 0.1^2 + 0.2^2 + 0.1^2 + 0.4^2 + 0^2 = \sqrt{0.22} = 0.47$$

$$\sqrt{\sum_{k=1}^{n}(t2_k)^2} = 0^2 + 0^2 + 0.3^2 + 0.2^2 + 0.1^2 = \sqrt{0.14} = 0.37$$

$$\cos(\overrightarrow{Web\ Semântica}, \overrightarrow{Ontologia}) = \frac{0.11}{0.47 \times 0.37} = 0.63$$

3. COMPUTAÇÃO DISTRIBUÍDA

3.1 SISTEMAS COM MULTIPLOS PROCESSADORES

Apesar da rápida evolução dos processadores há uma constante busca por poder computacional. Por muito tempo a solução mais comum para este desafio foi o aumento da capacidade de processamento dos processadores. Esta evolução ainda hoje segue a estimativa proposta por Gordon Moore na década de 70, denominada lei de Moore, que prevê a duplicação do número de transistores[1] comportados em uma pastilha a cada 18 meses (MOLLICK, 2006).

Neste contexto os sistemas com múltiplos processadores surgem como uma solução alternativa para o aumento da capacidade de processamento. De acordo com a taxonomia de Flynn os sistemas com múltiplos processadores são classificados como MIMD (múltiplos fluxos de instruções múltiplos fluxos de dados). A taxonomia de Flynn (Figura 9) proposta por Michael J. Flynn ainda é a maneira mais usual de organizar sistemas com capacidade de processamento paralelo (STALLINGS, 2010).

	Single Instruction **(Um fluxo de instruções)**	***Multiple Instruction*** **(Múltiplos fluxos de instruções)**
Single Data **(Um fluxo de dados)**	SISD	MISD
Multiple Data **(Múltiplos fluxos de dados)**	SIMD	MIMD

Figura 9 - Taxonomia de Flynn.
Fonte: adaptado de (DANTAS, 2009)

[1] O aumento do número de transistores proporciona maior capacidade processamento, porém esta solução pode deixar de ser viável devido a limites físicos.

Segundo Stallings (2010) as arquiteturas classificadas como MIMD possuem múltiplos processadores que executam instruções de forma independente dos demais.

Os sistemas com múltiplos processadores possuem duas ou mais UCP's (Unidade Central de Processamento – do Inglês CPU – Central Processing Unit) trabalhando em conjunto que se comunicam por um meio físico (TANENBAUM, 2010). Esta característica possibilita a execução paralela de instruções, oferecendo um ganho de desempenho quando comparado ao processamento sequencial.

A arquitetura destes sistemas pode ser dividida basicamente em três categorias: multiprocessador, multicomputador e sistemas distribuídos. Na Figura 10 e na Tabela 4 é possível verificar algumas características das estruturas citadas.

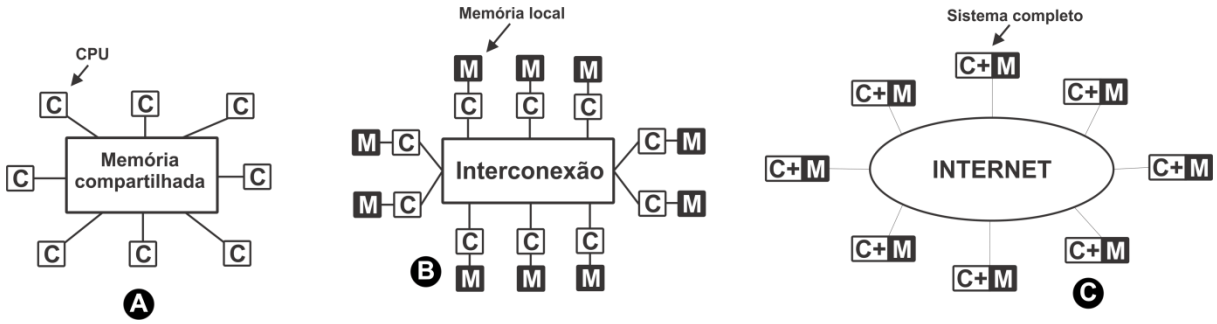

Figura 10 - Sistemas multiprocessadores (a), multicomputadores (b), e sistemas distribuídos (c).
Fonte: adaptado de (TANENBAUM; STEEN, 2007).

Item	Multiprocessador	Multicomputador	Sistema Distribuído
Configuração do nó	CPU	CPU, RAM, Interface de Rede	Computador completo
Periféricos do nó	Tudo compartilhado	Exc. Compartilhada, talvez disco	Conjunto completo por nó
Localização	Mesmo rack	Mesma sala	Possivelmente espalhado pelo mundo
Comunicação entre nós	RAM compartilhada	Interconexão dedicada	Rede tradicional
Sistemas operacionais	Um compartilhado	Múltiplos, mesmo	Possivelmente todos diferentes
Sistemas de arquivos	Um compartilhado	Um compartilhado	Cada nó tem seu próprio
Administração	Um compartilhado	Um compartilhado	Várias organizações

Tabela 4 - Características de Multiprocessador, Multicomputador e Sistemas Distribuídos.
Fonte: (TANENBAUM; STEEN, 2007).

Multiprocessador é um sistema de computador fortemente acoplado no qual duas ou mais CPU's compartilham acesso total a uma memória comum. Os processos se comunicam através da memória; o que cada processo escreve pode ser lido pelos demais. Devido a

memória ser compartilhada este modelo utiliza algum mecanismo de sincronização de processos com objetivo controlar a ordem de acesso a mesma (TANENBAUM, 2010).

Um multicomputador é um computador formado por várias CPU's que não compartilham memória, ou seja, cada CPU possui uma memória local. De acordo com Tanenbaum (2010) multicomputadores são fortemente acoplados. Entretanto, Dantas (2005) afirma que multicomputadores são fracamente acoplados, pois a comunicação entre processos é feita pela troca de mensagens entre os processos em execução. Os nós de uma estrutura de multicomputador geralmente possuem uma CPU, RAM, uma interface de rede e talvez um disco rígido para a paginação. Vale a pena salientar que o meio físico de comunicação entre os nós do multicomputador é de alta velocidade, tornando sua escalabilidade reduzida se comparado com um sistema distribuído e geralmente trabalham paralelamente.

Um sistema distribuído é semelhante a um multicomputador, porém podem ser interligados por redes normais (ETHERNET). Geralmente nesta estrutura cada componente da estrutura é um sistema completo, com todos os periféricos, e executam os processos de forma distribuída.

3.2 COMPUTAÇÃO DISTRIBUÍDA DE ALTO DESEMPENHO

A computação distribuída de alto desempenho vem se tornando mais popular devido ao avanço de diversas tecnologias computacionais, entre elas, o barateamento de microcomputadores e o avanço das redes de computadores têm sido fundamentais na configuração no cenário atual.

Nos últimos 50 anos as tecnologias computacionais vêm evoluindo em ritmo acelerado, segundo Tanenbaum e Steen (2007) neste período máquinas que custavam milhões de dólares e executavam uma instrução por segundo, evoluíram até a concepção de máquinas que custam alguns poucos dólares e podem executar um bilhão de instruções por segundo.

A evolução da capacidade de processamento é evidente, porém esta evolução pode desacelerar devido aos limites físicos. De acordo com Dantas (2005) alguns exemplos de restrições físicas que afetam na evolução dos processadores são:

- O limite na quantidade de componentes que podem ser instalados nos circuitos dos processadores;
- O aquecimento dos processadores atuais quando executando em altas frequências;
- A grande quantidade de colisões e consequente ineficiência da topologia de barramento encontrada nos computadores convencionais;
- A limitação de armazenamento e transferência de informação nas memórias com as tecnologias ora existentes.

A tentativa de contornar as limitações físicas, aliado a computadores mais acessíveis e a evolução das tecnologias de redes, tornou a computação distribuída uma solução viável para problemas que demandam alto poder de processamento. Segundo Dantas (2005) "A computação distribuída de alto desempenho pode ser entendida como um segmento da ciência da computação que tem como objetivo a melhoria do desempenho de aplicações distribuídas e paralelas, utilizando-se para tal de complexa infraestrutura computacional".

Normalmente a computação distribuída é utilizada para resolver problemas que demandam alto desempenho, ou possuam grande volume de informação, como previsão meteorológica, computação gráfica, simulações matemáticas, entre outras. Atualmente as infraestruturas mais utilizadas são os *grids* e os *clusters*.

A Tabela 5 demonstra os 10 computadores mais velozes do mundo, sendo que todos os computadores da lista possuem a estrutura de multicomputadores.

Ranking	País	Nome	RAM (GB)	Núcleos
1	E.U.A	Sequoia	1572864	1572864
2	Japão	K computer	1410048	705024
3	E.U.A	Mira	*	786432
4	Alemanha	SuperMUC	*	147456
5	China	Tianhe-1A	229376	186368
6	E.U.A	Jaguar	298592	598016
7	Italia	Fermi	*	163840
8	Alemanha	JuQUEEN	*	131072
9	França	Curie thin nodes	308736	77184
10	China	Nebulae	*	120640

Tabela 5 - Top 10 supercomputadores.
Fonte: (TOP500, 2012).

3.2.1 CLUSTER

Os *clusters* são agregados de computadores criados para se obter maior desempenho ou disponibilidade. Um *cluster* pode ser classificado como um sistema composto por vários computadores fracamente acoplados que geralmente tem alto poder de processamento. Esta estrutura comporta-se como um único sistema, trabalhando de forma transparente.

Tanenbaum e Steen (2007) afirmam "que em um *cluster* o hardware subjacente consiste em um conjunto de estações de trabalho ou computadores pessoais semelhantes, conectados por meio de uma rede local de alta velocidade. Além disso, cada nó executa o mesmo sistema operacional". Esta definição pode ser considerada tradicional, visto que um cluster pode ter outras características.

Stallings (2010) possui uma visão mais contemporânea e afirma que "Um *cluster* consiste de um conjunto de computadores completos, conectados entre si, que trabalham juntos como um recurso computacional unificado, criando a ilusão de ser uma única máquina". O termo computador completo é utilizado, pois o sistema pode executar independentemente do *cluster*.

Percebe-se que Stallings (2010) apresenta uma definição mais flexível de *cluster* quando comparada a visão de Tanenbaum e Steen (2007). A Figura 11 apresenta métricas para classificação de cluster proposta por Dantas.

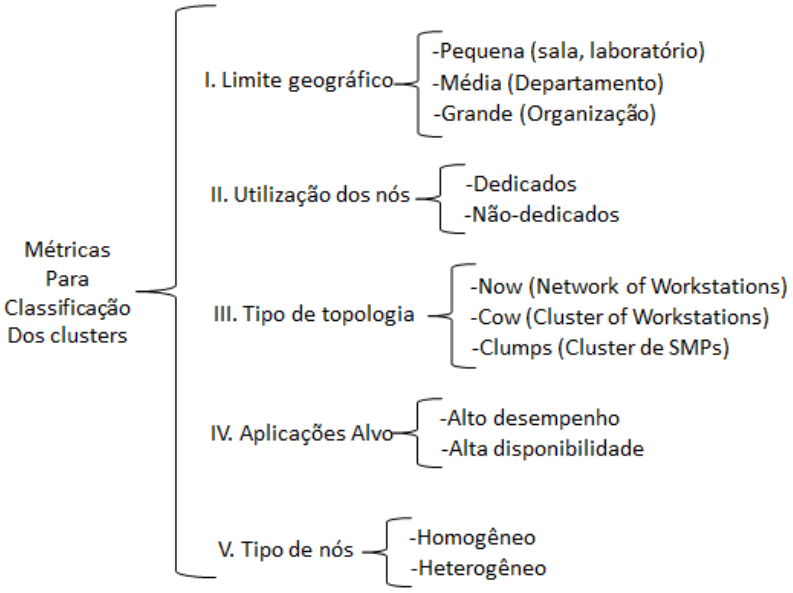

Figura 11 - Classificação de Clusters.
Fonte: Adaptado de (DANTAS, 2009).

Apesar de existirem pequenas divergências na literatura é possível concluir que clusters são compostos por nós que se encontram na mesma organização conectados a uma rede, e tem como objetivo o ganho de desempenho/disponibilidade. Esta estrutura é transparente ao usuário, ou seja, para o usuário é como se ele estivesse acessando um único sistema. Apesar desta arquitetura permitir nós heterogêneos, normalmente ela é homogênea, possuindo hardware e sistemas operacionais em comum.

3.2.2 GRID

Buyya e Venugopal (2005) afirmam que um *grid* computacional é uma infraestrutura que envolve o uso integrado e colaborativo de computadores, redes, bancos de dados e instrumentos científicos que podem pertencer e serem geridos por várias organizações. Também é possivel afirmar que "... cada sistema pode cair sob um domínio administrativo diferente, e podem ser muito diferente no que tange a hardware, software e tecnologia de rede empregada" (TANENBAUM; STEEN, 2007).

A principal motivação para utilização de *grids* computacionais é a possibilidade de utilizar recursos de diversos domínios, sendo que estes recursos se encontram ociosos em grande parte do tempo. Os recursos de um *grid* não são dedicados, tornando possível sua utilização mesmo que em outrora sejam dedicados a outros fins, como computadores de empresas, universidades, de uso pessoal, entre outros. Recursos podem ser sensores, dados, computadores, etc.

Outra característica importante que permite um *grid* ser altamente escalável é possibilidade de utilizar recursos heterogêneos, ou seja, um *grid* pode ser composto por hardwares, sistemas operacionais, redes distintas (TANENBAUM; STEEN, 2007). É válido afirmar que esta estrutura computacional também pode possuir clusters em sua composição.

O projeto SETI@HOME (setiathome.berkeley.edu) é um exemplo de grid computacional. SETI significa *Search for ExtraterresTrial Intelligence* (busca por inteligência extraterrestre). Este projeto busca vida extraterrestre analisando sinais de rádio, com objetivo de identificar mensagens emitidas por outras civilizações. O SETI@HOME foi lançado em 17 de maio de 1999. Em seus 10 anos de operação, atraiu mais de 5 milhões de participantes, localizados em 226 países (KORPEL et al., 2011).

A Figura 12 ilustra uma arquitetura de modelo de *grid* computacional. A arquitetura exposta é composta por quatro camadas.

Figura 12 - Camadas de um Grid.
Fonte: adaptado de (DANTAS, 2009).

Aplicações e Serviços: esta camada representa aplicações e usuário.

Middleware: esta camada atua como mediadora entre a comunicação da aplicação e os recursos disponíveis. Berman, Fox e Hey (2003) afirmam que o *middleware* abstrai a complexidade da infraestrutura, simplificando a comunicação da camada de aplicação com a camada de recursos.

Recursos: Esta camada representa os recursos que compõe o *grid*, podendo conter: computadores pessoais, *clusters*, sensores, entre outros. Além disso, o conjunto de recursos representado por esta camada é altamente dinâmico, ou seja, novos podem ser adicionados ou retirados, e, como resultado pode haver variação do desempenho na estrutura (BERMAN; FOX; HEY, 2003).

REDES: Esta camada representa a comunicação entre recursos. É composta por *hubs*, roteadores e *switchs* e os meios de comunicação, protocolos e topologias são heterogêneos.

3.2.2.1 GridGain

O projeto que iniciou em 2005 atualmente conta com duas versões: GridGain *Community Edition* que possui código aberto, e a versão GridGain *Enterprise Edition*, que é a versão comercial do software. Neste trabalho a versão utilizada é a *Community Edition*.

Segundo Ivanov (2012), o GridGain é um *middleware* desenvolvido em JAVA que permite o desenvolvimento de aplicações distribuidas de alto desempenho. Segundo

Tanenbaum (2010) um *middleware* permite um sistema distribuído manter-se uniforme mesmo operando em hardwares e sistemas operacionais distintos.

Rubbo (2011), afirma que o GridGain é um *framework*/*middleware* de código aberto para computação em GRID, e tem como objetivo fornecer uma plataforma mais simples e produtiva para computação em grade em ambientes empresariais.O GridGain também pode ser considerado um *framework* pois oferece um arcabouço capaz de simplificar o processo de implementação através de reuso. Mattsson (2000) afirma que um *framework* é composto por um conjunto de classes, cujas instâncias trabalham em conjunto. Uma das principais vantagens da utilização de um *framework* é a reusabilidade, tornando viável a possibilidade de expansão.

Basicamente, o GridGain trabalha com o conceito de *task* e *jobs*, sendo que uma tarefa (*task*) é dividida em diversos pequenos problemas (*jobs*). O envio dos *jobs* e o balanceamento de carga fica a cargo do GridGain. Desse modo, a aplicação que está sendo executada deve apenas definir uma *task* e como ela vai ser dividida dando origem aos *jobs*. Após a divisão os *jobs* são enviados aos nós que compõem o *grid* para serem executados.

4. MODELO PROPOSTO

A seguir será descrito o modelo proposto dividindo a apresentação em duas etapas. A primeira etapa refere-se ao modelo lógico em que se detalha a interação entre os diferentes módulos componentes da proposição. A segunda etapa apresenta o modelo físico descrevendo os componentes tecnológicos, bem como, a justificativa de utilização dos mesmos.

4.1 MODELO LÓGICO

O modelo lógico (Figura 13) é composto por um conjunto de passos que possibilitam a interconexão de conteúdo textual, representado por conceitos em um domínio de problema, visando prover suporte a tarefas de descoberta de conhecimento.

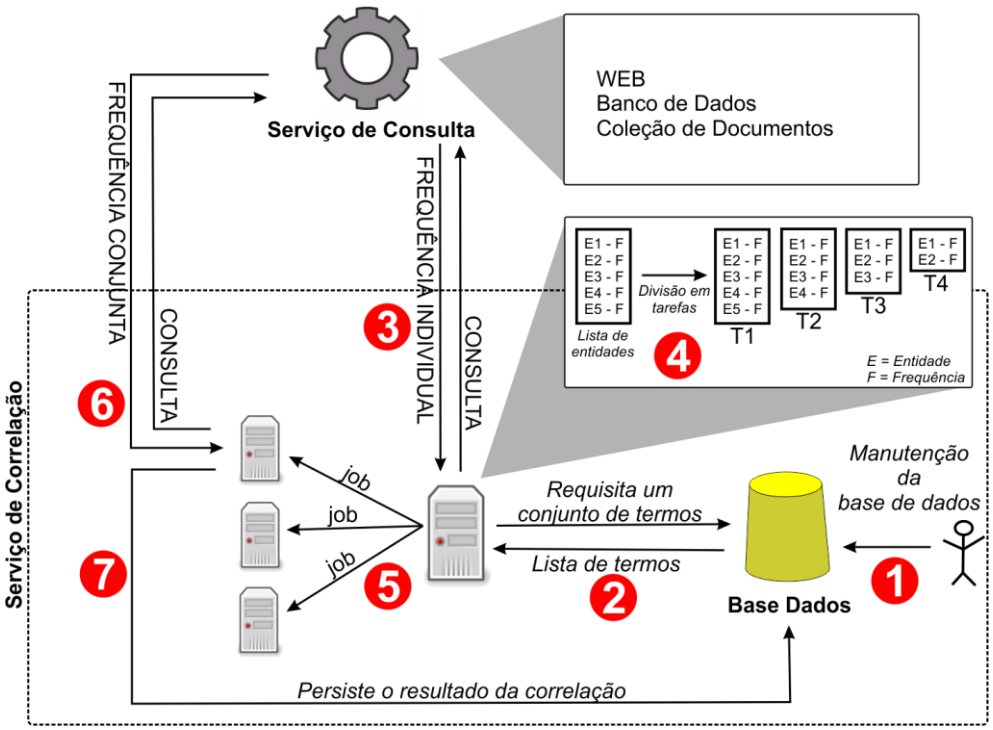

Figura 13 - Modelo lógico da arquitetura de descoberta de conhecimento em bases textuais.

Inserção e classificação dos termos (1): Nesta etapa é realizada a manutenção da base de dados. Aqui são inseridos novos termos, classes, e domínios. Um termo é constituído por mais palavras que, quando associado a uma classe gera um conceito. A diferença entre termos e conceitos é que o segundo possui um significado, enquanto termos isolados representam apenas palavras. Com o intuito de gerar conceitos, os termos são atribuídos de acordo com as classes inseridas. As Tabela 6, 7 e 8 exemplificam os termos, classes, e conceitos:

TERMO
Jaguar
C++
Java
Euro
Brasil

Tabela 6 - Exemplo de termos inseridos na base de dados.

CLASSE
Moeda
Linguagem de programação
Carro
País
Animal

Tabela 7 - Exemplos de classes presentes na base de dados.

CONCEITO	
TERMO	CLASSE
Brasil	País
Jaguar	Animal
Java	Linguagem de programação
C++	Linguagem de programação
Jaguar	Carro
Euro	Moeda

Tabela 8 - Exemplos de conceitos (composto por termo e classe) presentes na base de dados.

A Tabela 8 representa os conceitos, ou seja, contém termos da Tabela 6, que foram classificados a partir das classes contidas na Tabela 7. Fica evidente que este processo agrega sentindo a um termo. Esta característica pode ser observada na Tabela 8, que possui dois termos iguais (jaguar), porém com sentidos (classes) diferentes.

Após o processo descrito anteriormente, no qual foram gerados conceitos, é necessário selecionar o domínio a qual os conceitos pertencem. O domínio pode ser entendido como “domínio do problema”. Desse modo, a inserção de conceitos em um domínio permite

que o processo de correlação seja aplicado para um fim especifico. Abaixo podem ser observados exemplos de domínios na Tabela 9, e conceitos e domínios na Tabela 10:

DOMÍNIO	
ID	DESCRIÇÃO
1	Tecnologia
2	Genérico
3	Economia
4	Saúde

Tabela 9 - Exemplo de domínios.

CONCEITO-DOMÍNIO			
ID	CONCEITO		DOMÍNIO
	TERMO	CLASSE	
1	Jaguar	Animal	Genérico
2	Java	Linguagem de programação	Tecnologia
3	C++	Linguagem de programação	Tecnologia
4	Jaguar	Carro	Genérico
5	Euro	Moeda	Economia

Tabela 10 - Exemplos de conceitos e seus respectivos domínios.

O processo de classificação dos termos, e a seleção de domínios para os conceitos são executados manualmente. Estes processos requerem um conhecimento sobre o domínio do problema. Outro aspecto pertinente diz respeito ao suporte provido pelo modelo de dados à utilização de semântica através de classes e domínios genéricos. Esta característica permite, por exemplo, agregar funcionalidades em serviços de busca que não possuam semântica.

Requisição de um conjunto de termos (2): o serviço de correlação requisita um conjunto de termos que será utilizado na análise. Como resposta o serviço recebe uma lista de termos e as classes desses termos. A lista pode conter todos os conceitos, ou pode ser filtrada por domínios específicos.

Geração da frequência individual (3): a partir da lista de conceitos inicia-se o processo de geração da frequência individual de cada conceito. O valor da frequência individual é obtido através de uma pesquisa no servidor de consulta, onde o valor retornado é referente ao número de páginas em que o conceito ocorre. Após este processo cada conceito deve conter o valor de sua frequência. A Tabela 11 ilustra este cenário.

CONCEITO			DATA
TERMO	CLASSE	FREQUÊNCIA	
Brasil	País	622.000.000	25/03/2012
Jaguar	Animal	24.800.000	25/03/2012
Java	Linguagem de programação	239.000.000	25/03/2012
C++	Linguagem de programação	40.100.000	25/03/2012
Jaguar	Carro	18.400.000	25/03/2012
EURO	Moeda	765.000.000	25/03/2012

Tabela 11 - Exemplos de conceitos e suas respectivas frequências individuais.

Ao finalizar o processo os conceitos e suas respectivas frequências são armazenadas na base de dados. Além de gerar a frequência esta etapa é responsável por obter a data do sistema. A data é um dado fundamental, visto que o processo de correlação é temporal.

Divisão da tarefa (4): o serviço de correlação inicia o processo de divisão separando a lista geral em diversas listas menores. A divisão da lista é realizada para que os diversos computadores que irão atender ao serviço possam chegar a uma solução (processo de correlação) de forma distribuída. Nesse sentido, considera-se a lista inicial como a tarefa que deve ser executada, e as listas menores oriundas da divisão da tarefa os trabalhos. Este processo será detalhado na seção 3.2.4.

Envio dos trabalhos (5): após a criação dos trabalhos o serviço de correlação envia estes para os computadores que compõem a estrutura. O serviço de correlação realiza o controle de modo que quando um computador encerrar seu trabalho este receberá outro enquanto houver itens a serem processados na lista de tarefas. Este processo se repete até não restar mais trabalhos a serem executados, assim finalizando a tarefa. As próximas etapas descritas são executadas de forma distribuída.

Geração da frequência conjunta (6): como visto anteriormente os trabalhos são executados pelos computadores que compõem o serviço de correlação. Um trabalho consiste na geração da frequência conjunta dos conceitos e no cálculo do coeficiente de correlação. A forma de se obter a frequência conjunta e a individual é semelhante, visto que ambas são obtidas através de requisições enviadas ao servidor de consulta. O valor da frequência individual é representado pelo número de documentos que contenham determinado conceito, enquanto que a frequência conjunta se refere ao total de documentos em que dois conceitos quaisquer apareçam conjuntamente. A Tabela 12 exemplifica o resultado obtido após a execução do serviço.

CONCEITO			CONCEITO			FREQUENCIA CONJUNTA
TERMO	CLASSE	FREQUÊNCIA INDIVIDUAL	TERMO	CLASSE	FREQUÊNCIA INDIVIDUAL	
Brasil	País	622.000.000	Jaguar	Animal	24.800.000	1.790.000
Brasil	País	622.000.000	Java	Lin. deProg.	239.000.000	25.300.000
Brasil	País	622.000.000	C++	Lin. deProg.	40.100.000	2.040.000
Brasil	País	622.000.000	Jaguar	Carro	18.400.000	21.300
Brasil	País	622.000.000	EURO	Moeda	765.000.000	101.000.000
Jaguar	Animal	24.800.000	Java	Lin. deProg.	239.000.000	369.000
Jaguar	Animal	24.800.000	C++	Lin. deProg.	40.100.000	69.000
Jaguar	Animal	24.800.000	Jaguar	Carro	18.400.000	194.000
Jaguar	Animal	24.800.000	EURO	Moeda	765.000.000	688.000
Java	Lin. deProg.	239.000.000	C++	Lin. deProg.	40.100.000	18.300.000
Java	Lin. deProg.	239.000.000	Jaguar	Carro	18.400.000	54.600
Java	Lin. deProg.	239.000.000	EURO	Moeda	765.000.000	19.800.000
C++	Lin. deProg.	40.100.000	Jaguar	Carro	18.400.000	87.000
C++	Lin. deProg.	40.100.000	EURO	Moeda	765.000.000	1.220.000
Jaguar	Carro	18.400.000	EURO	Moeda	765.000.000	1.200.000

Tabela 12 - Exemplos dos resultados obtidos após a geração da frequência conjunta.

Cálculo do coeficiente de correlação (7): com os valores da frequência individual e conjunta dos conceitos é possível calcular o coeficiente de correlação, o qual representa a força de correlação entre dois termos. Algumas equações utilizadas para obter o coeficiente de correlação foram apresentadas no Capítulo 2 do presente trabalho. A Tabela 13 representa o resultado obtido ao término deste processo e a Tabela 14 representa como os dados são armazenados no banco de dados considerando um marcação de tempo.

CONCEITO			CONCEITO			FREQUENCIA CONJUNTA	COEFICIENTE DE CORRELAÇÃO
TERMO	CLASSE	FREQUÊNCIA INDIVIDUAL	TERMO	CLASSE	FREQUÊNCIA INDIVIDUAL		
Brasil	País	622.000.000	Jaguar	Animal	24.800.000	1.790.000	0.00016
Brasil	País	622.000.000	Java	Lin. deProg.	239.000.000	25.300.000	0.00056
Brasil	País	622.000.000	C++	Lin. deProg.	40.100.000	2.040.000	0.00045
Brasil	País	622.000.000	Jaguar	Carro	18.400.000	21.300	0.01188
Brasil	País	622.000.000	EURO	Moeda	765.000.000	101.000.000	0.00343
Jaguar	Animal	24.800.000	Java	Lin. deProg.	239.000.000	369.000	0.003
Jaguar	Animal	24.800.000	C++	Lin. deProg.	40.100.000	69.000	0.00909
Jaguar	Animal	24.800.000	Jaguar	Carro	18.400.000	194.000	0.00321
Jaguar	Animal	24.800.000	EURO	Moeda	765.000.000	688.000	0.0031
Java	Lin. deProg.	239.000.000	C++	Lin. deProg.	40.100.000	18.300.000	0.0012
Java	Lin. deProg.	239.000.000	Jaguar	Carro	18.400.000	54.600	0.00043
Java	Lin. deProg.	239.000.000	EURO	Moeda	765.000.000	19.800.000	0.00045
C++	Lin. deProg.	40.100.000	Jaguar	Carro	18.400.000	87.000	0.0007
C++	Lin. deProg.	40.100.000	EURO	Moeda	765.000.000	1.220.000	0.0001
Jaguar	Carro	18.400.000	EURO	Moeda	765.000.000	1.200.000	0.0003

Tabela 13 - Resultado obtido após o processo de correlação.

CONCEITO		CONCEITO		FREQUENCIA CONJUNTA	COEFICIENTE DE CORRELAÇÃO	DATA
TERMO	CLASSE	TERMO	CLASSE			
Brasil	País	Jaguar	Animal	1.790.000	0.00016	25/03/2012
Brasil	País	Java	Lin. de Prog.	25.300.000	0.00056	25/03/2012
Brasil	País	C++	Lin. de Prog.	2.040.000	0.00045	25/03/2012
Brasil	País	Jaguar	Carro	21.300	0.01188	25/03/2012
Brasil	País	EURO	Moeda	101.000.000	0.00343	25/03/2012
Jaguar	Animal	Java	Lin. de Prog.	369.000	0.003	25/03/2012
Jaguar	Animal	C++	Lin. de Prog.	69.000	0.00909	25/03/2012
Jaguar	Animal	Jaguar	Carro	194.000	0.00321	25/03/2012
Jaguar	Animal	EURO	Moeda	688.000	0.0031	25/03/2012
Java	Lin. de Prog.	C++	Lin. de Prog.	18.300.000	0.0012	25/03/2012
Java	Lin. de Prog.	Jaguar	Carro	54.600	0.00043	25/03/2012
Java	Lin. de Prog.	EURO	Moeda	19.800.000	0.00045	25/03/2012
C++	Lin. de Prog.	Jaguar	Carro	87.000	0.0007	25/03/2012
C++	Lin. de Prog.	EURO	Moeda	1.220.000	0.0001	25/03/2012
Jaguar	Carro	EURO	Moeda	1.200.000	0.0003	25/03/2012

Tabela 14 - Exemplificação das informações que irão persistir na base de dados.

4.2 MODELO FÍSICO

Nas próximas seções serão detalhados os componentes tecnológicos e os serviços, e como estes se interconectam visando oferecer uma visão física do modelo proposto.

4.2.1 SERVIÇO DE CONSULTA

O serviço de consulta utilizado foi o BING® (www.bing.com.br). As consultas foram realizadas por meio da Bing Search API 2.0. A utilização deste servidor de consulta é justificada pela possibilidade de cada IP válido realizar 7 pesquisas por segundo, enquanto serviços semelhantes oferecem um número máximo de pesquisas em determinado período. Como exemplo pode-se citar a API de busca do Google®, que possui um limite de 100 consultas diárias. Vale a pena salientar que a limitação da Bing Search API 2.0 não representa um impacto significativo para o cenário do trabalho, visto que, o modelo proposto utiliza uma estrutura de *grid*, onde cada computador poderá realizar 7 buscas por segundo desde que possua um IP válido.

A API utilizada disponibiliza formatos para resposta de uma consulta, sendo possível utilizar XML ou JSON. No modelo proposto a opção escolhida foi o JSON por possuir uma representação mais simplificada em relação ao XML. Segundo Deitel e Deitel (2010), o JSON é uma alternativa à XML para representar dados. Nurseitov (2009) afirmaque o JSON além de menor é mais rápido, e usa menos recursos quando comparado ao XML. Como consequência o JSON vem sendo cada vez mais reconhecido como um padrão adequado para tranferência de dados entre aplicações (LOUDON, 2010).

Apesar do resultado da consulta conter diversas informações, a única informação que é utilizada pelo modelo de correlação refere-se ao número de páginas em que se encontra determinado termo. Com base nisso, a consulta foi refinada objetivando diminuir o tamanho do objeto JSON. A consulta refinada pode ser observada na Figura 14.

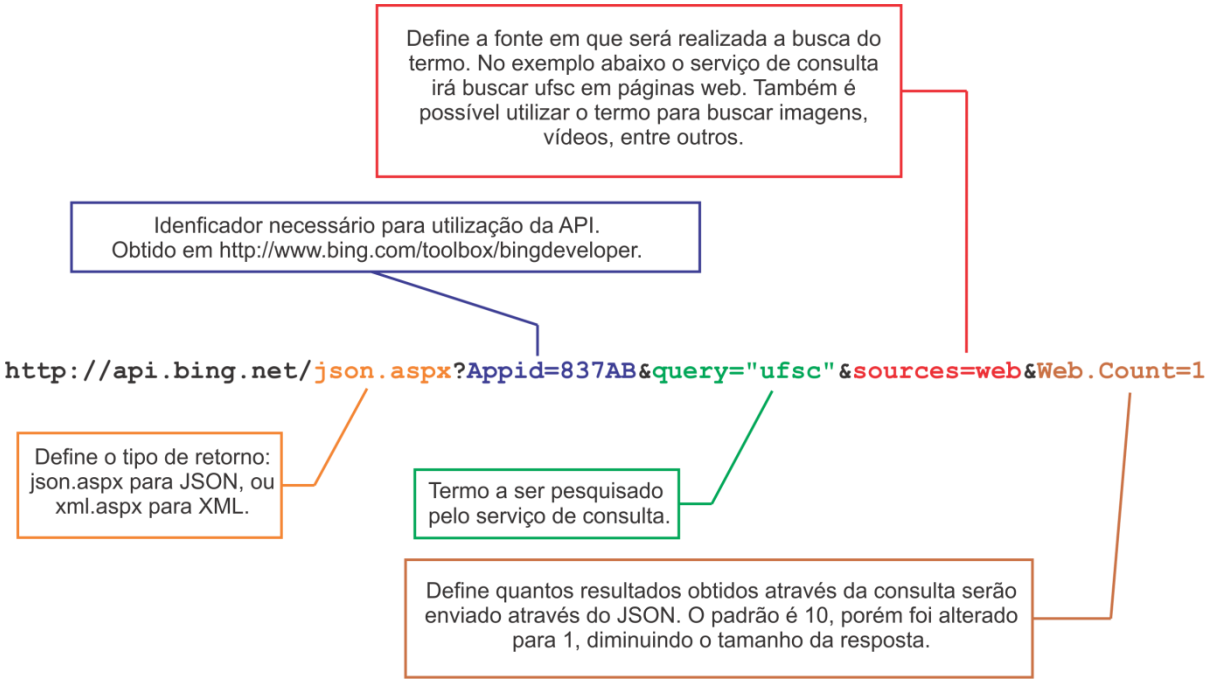

Figura 14 - Detalhamento da requisição enviada ao servidor de consulta.

A consulta da Figura 14 irá retornar um objeto JSON que contém o resultado da consulta. A Figura 15 ilustra a estrutura do objeto JSON enviado pelo servidor de consulta.

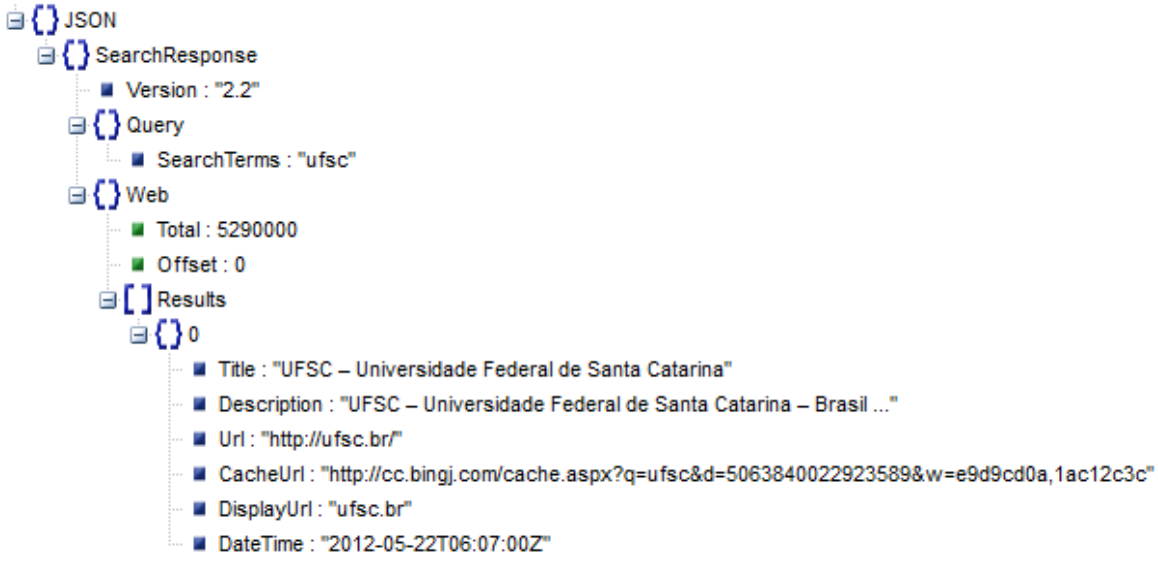

Figura 15 - Exemplo de resposta do servidor de consulta.

Percebe-se que a mensagem JSON está resumida, pois foram retiradas informações que não seriam utilizadas no modelo proposto. Como visto anteriormente a única informação pertinente é a quantidade de páginas em que o termo pesquisado se encontra. Como exemplo, pode ser observada na Figura 15 o resultado para a busca considerando o termo "ufsc" como um total de 5.290.000 páginas.

O serviço de consulta Bing® demonstrou ser adequando para a utilização na aplicação. Apesar da versão do serviço disponibilizada gratuitamente possuir limitações ela se adéqua ao serviço de correlação, e por ser atualmente um dos maiores servidores de consulta possui bom desempenho e boa confiabilidade.

4.2.2 MODELO DIMENSIONAL

A representação de dados do modelo proposto utiliza o conceito *Data Warehouse* (DW). Segundo Inmon (1992) um DW é "uma coleção de dados orientada por assunto, integrada, não volátil, variante no tempo que dá apoio às decisões da administração". ELMASRI (2005) afirma que "os *Data Warehouses* proporcionam acesso aos dados para análise complexa, descoberta de conhecimento e tomada de decisão. Eles dão suporte às demandas de alto desempenho por dados e informações de uma organização". A partir destas alegações é justificada a utilização de um DW no modelo proposto, visto que o modelo

auxilia na tomada de decisão, através de análises complexas a partir de dados oriundos de bases textuais. A Figura 16 ilustra a modelagem de DW utilizada no modelo.

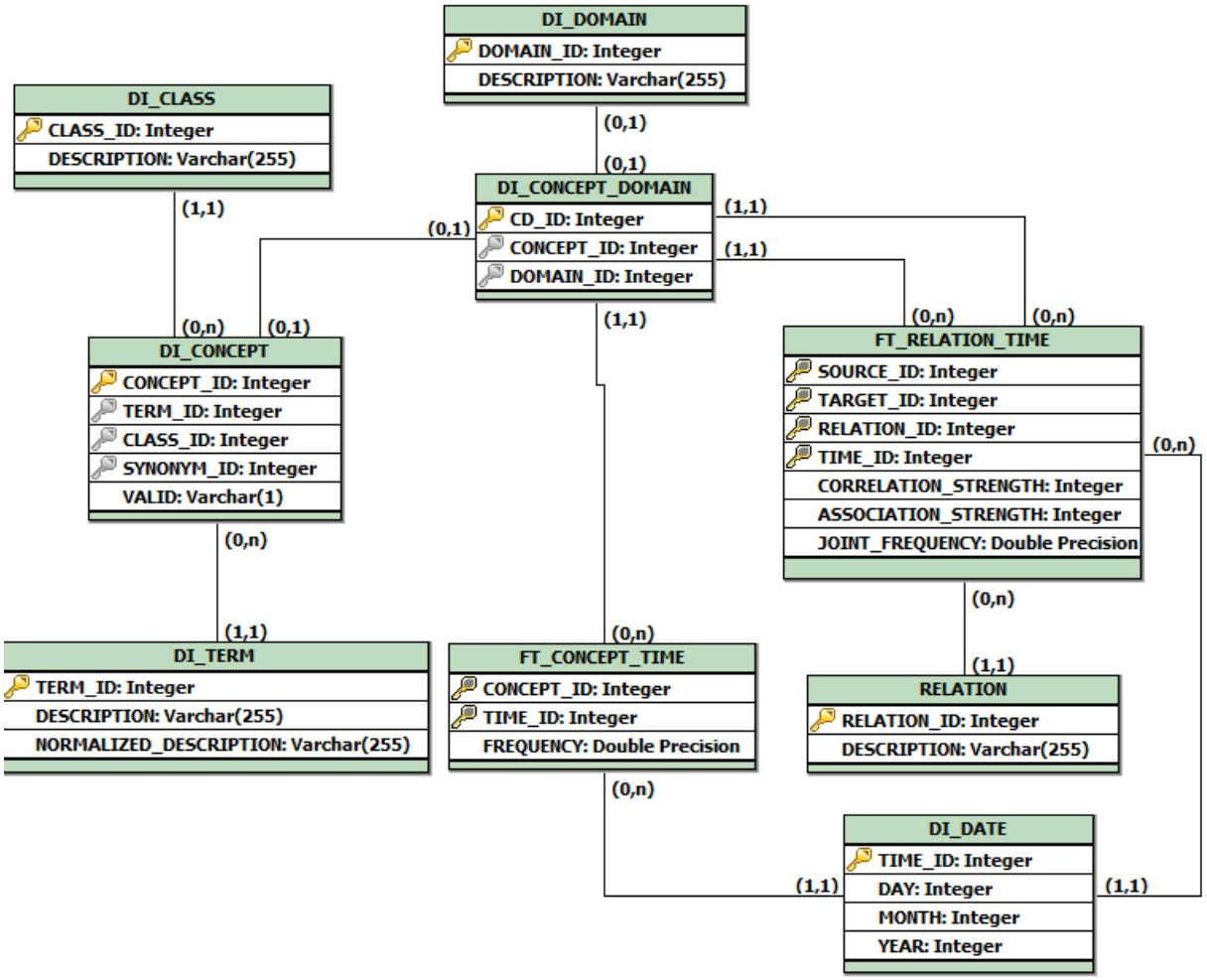

Figura 16 - Modelo lógico da base de dados utilizado no processo de correlação rápida.
Fonte: adaptado de (BOVO, 2011).

A seguir são detalhadas as tabelas (dimensões e fatos) que compõe o modelo do banco de dados:

DI_TERM: dimensão responsável por armazenar os termos sem qualquer contexto. A tabela é composta por um identificador (TERM_ID) que possui um valor sequencial representando a identificação do termo, uma coluna para a descrição do termo (DESCRIPTION), e uma coluna chamada NORMALIZED_DESCRIPTION com a descrição normalizada, ou seja, sem palavras pouco representativas (artigos, preposições, entre outras) e considerando somente a raiz (sem sufixos) das demais palavras. Segundo Bovo (2011) "A normalização refere-se ao processo de reduzir um termo à sua raiz. Por exemplo, os termos "tecnologia" e "tecnologias", serão reduzidos para apenas um termo: "tecnolog".

DI_CLASS: esta dimensão possui um número sequencial que identifica a classe (CLASS_ID) e a descrição da classe (DESCRIPTION). A partir da utilização de uma classe é possível contextualizar um termo, por exemplo, o termo jaguar pode ser atribuído a mais de uma classe.

DI_CONCEPT: esta dimensão representa um conceito, sendo composta por um termo aliado a uma classe. A tabela possui um campo que representa sua identificação (CONCEPT_ID), o identificador do termo (TERM_ID), o identificador da classe (CLASS_ID), um identificador de um termo que permite uma relação de sinonímia (sinônimo) (SYNONYM_ID), e um campo que define se o conceito foi validado pelo usuário (VALID) visto que os termos e conceitos iniciais podem ser carregados utilizando um processo automático de extração de informação. A partir desta tabela é possível contextualizar o termo, definindo uma classe para o mesmo. Este processo de contextualização do termo é realizado manualmente por um especialista do domínio.

DI_DOMAIN: esta tabela representa o domínio da análise. Um domínio é composto por um identificador (DOMAIN_ID) e por uma descrição (DESCRIPTION).

DI_CONCEPT_DOMAIN: esta dimensão é composta por um identificador do domínio (DOMAIN_ID), um identificador do conceito (CONCEPT_ID), e por um identificador sequencial que rotula o domínio e o conceito conjuntamente (CD_ID). A partir desta tabela é possível obter um termo associado a uma classe e um domínio de pesquisa.

DI_DATE: esta tabela é responsável por representar a dimensão do tempo. O tempo é armazenado em diversas granularidades: dia (DAY), mês (MONTH), e ano (YEAR). A entidade também possui um identificador (TIME_ID).

FT_CONCEPT_TIME: tabela de fato que representa a frequência (FREQUENCY) de conceito (CONCEPT_ID), composto basicamente por termo, classe e domínio. Visto a necessidade de análises de frequência de determinado conceito de maneira temporal a tabela possui ainda um identificador para a dimensão de tempo DI_DATE através da coluna (TIME_ID).

DI_RELATION: dimensão responsável por descrever (DESCRIPTION) a relação entre dois termos. Além da descrição há um campo identificador da relação (RELATION_ID).

FT_RELATION_TIME: esta tabela de fato representa o valor da correlação (CORRELATION_STRENGTH), associação (ASSOCIATION_STRENGTH), e frequência conjunta (JOINT_FREQUENCY). Estes valores são obtidos através da análise de dois termos (SOURCE_ID e TARGET_ID), os quais são ligados por algum tipo de relação (RELATION_ID). Há também a data da análise (TIME_ID) que possibilita a análise temporal.

4.2.3 SERVIÇO DE CORRELAÇÃO

A seguir o serviço de correlação será descrito de maneira detalhada a partir da Figura 17 que representa a arquitetura física.

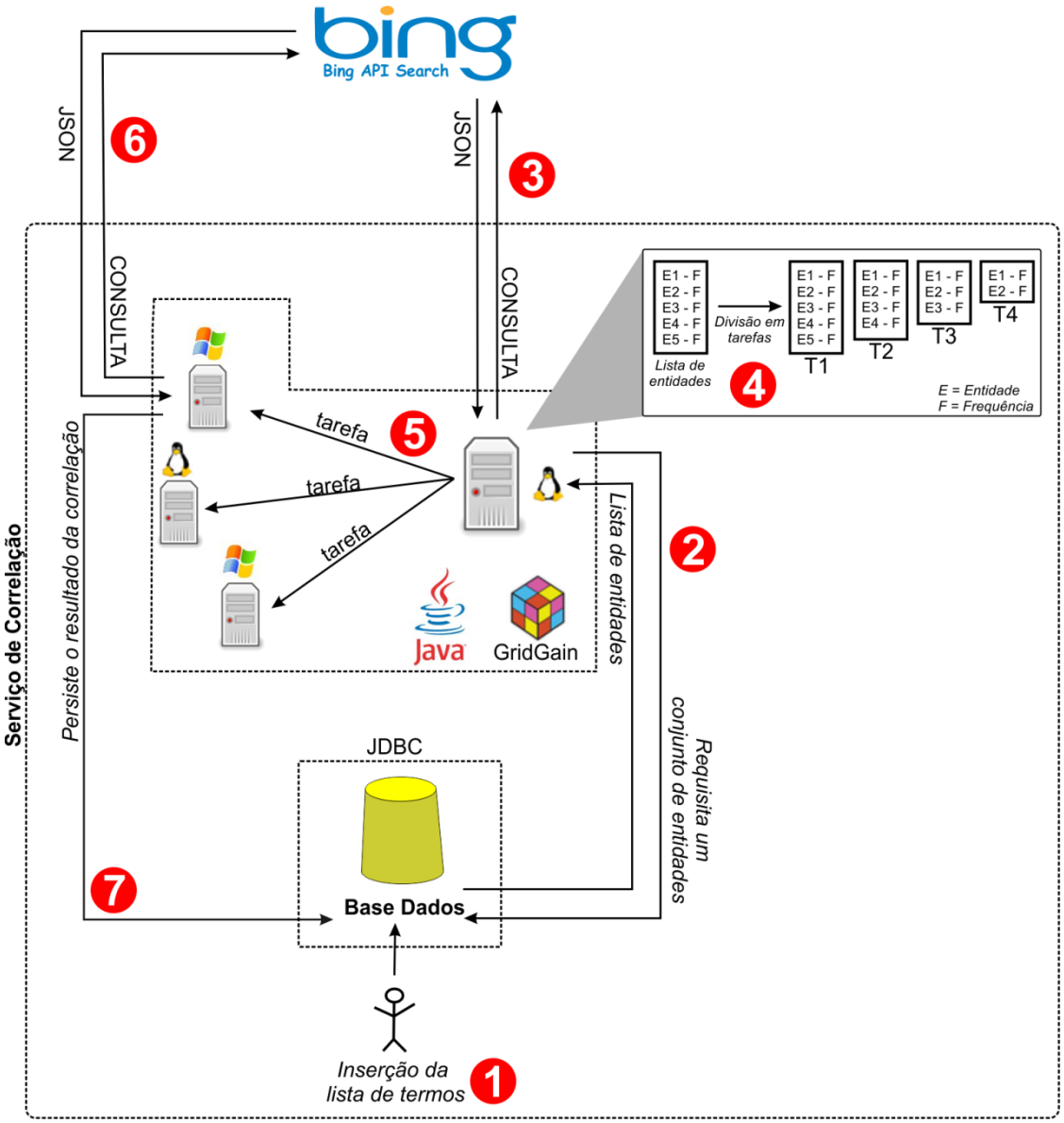

Figura 17 - Detalhamento do modelo de correlação rápida.

O processo de correlação começa requisitando uma lista de termos (2) previamente inseridos na base de dados (1). A comunicação entre a aplicação e a base de dados é realizada através da API JDBC (*Java Database Connectivity*). O computador em que o serviço de correlação foi iniciado é responsável por esta requisição. Ao fim da requisição dos termos o serviço cria uma tabela *Hash* que contém os termos oriundos da seleção. A estrutura criada pode ser observada na Figura 18.

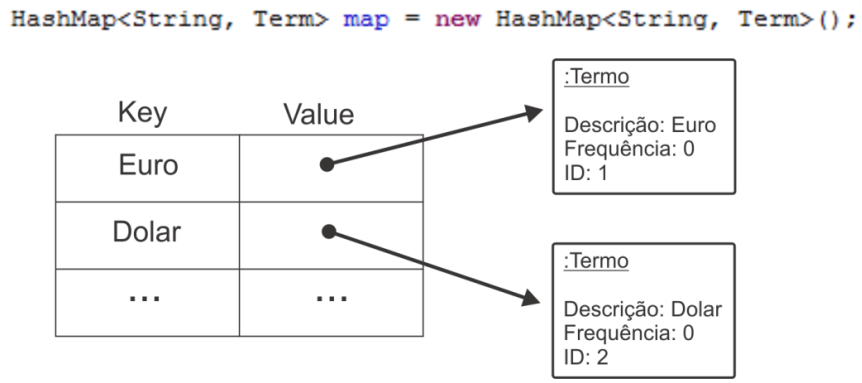

Figura 18 - Exemplo de tabela *Hash* de termos utilizada no processo de correlação rápida.

Após a *Hash* ser populada o nó principal inicia o processo de geração das frequências individuais de todos os termos. Para obter o valor da frequência o nó principal envia uma requisição HTTP (*Hypertext Transfer Protocol*) ao servidor de consulta (Bing®). A requisição pode ser observada na **Erro! Fonte de referência não encontrada.**4. A resposta do servidor e consulta é um objeto JSON que contêm o número de páginas em que o termo se encontra. O valor de retorno é atualizado na variável frequência encontrada no objeto termo referente à posição deste na tabela *Hash*. Após isso, os termos e suas respectivas frequências individuais são armazenadas na base de dados. A estrutura da tabela FT_CONCEPT_TIME, responsável por armazenar estas informações, pode ser observada na Figura 19.

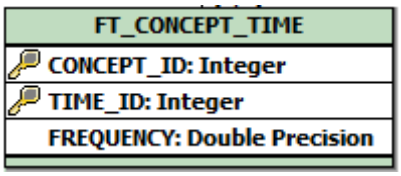

Figura 19 - Tabela de fato FT_CONCEPT_TIME.

Após todos os termos obterem a frequência individual, o nó principal inicia o processo de divisão dos *jobs*. O objetivo de um *job* é gerar a frequência conjunta entre dois termos, calcular o coeficiente de correlação e persistir essa informação no banco de dados. A quantidade de *jobs* gerados é igual ao número de termos presentes na tabela *Hash* menos um

(Figura 20). Para realizar o processo de correlação de forma distribuída a arquitetura utiliza *framework/middleware* GridGain, que foi apresentado na seção 3.3.2.1.

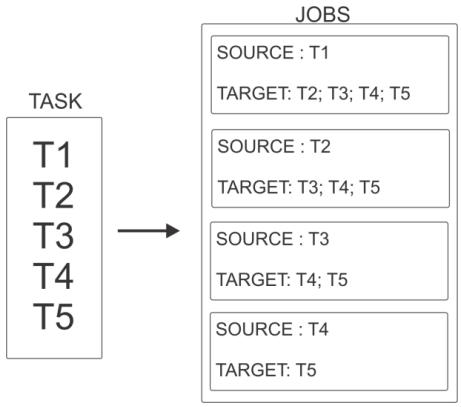

Figura 20 - Exemplo de divisão da tarefa em trabalhos.

Após e divisão da *task* em *jobs* o GridGain inicia o processo de distribuição, enviando os jobs aos computadores do *grid*. Este processo é repetido até que não reste nenhum *job* a ser executado. O GridGain também é responsável pelo balanceamento de carga.

O processo de geração da frequência conjunta é semelhante ao da frequência individual, porém é necessário enviar dois termos ao servidor de consulta. O resultado obtido é o número de páginas em que os dois termos ocorram conjuntamente. Cada *job* possui um termo origem (*source*) e uma lista de termos destino (*target*), sendo assim, o *job* calcula a frequência conjunta do termo origem com cada termo destino que compõe a lista.

Com o valor da frequência individual e conjunta é possível calcular o coeficiente de correlação. O cálculo de correlação utilizado é o Phi-squared, utiliza a tabela de contingência exposta na Tabela 2. Com base na tabela de contingência apresentada percebe-se que para a realização do cálculo é necessário o número de documento da base de dados. Porém, como o Bing® não informa a quantidade de páginas indexadas este valor foi fixado em 50 bilhões. A equação utilizada para calcular o coeficiente de correlação é a Phi-squared, apresentada da seção 2.1.1.5. Sua escolha é justificada por produzir uma normalização dos resultados entre 0 e 1, o que facilita a interpretação na análise de cenários.

O resultado do cálculo é armazenado na base de dados juntamente com os dois termos (origem e destino) e a frequência conjunta desses termos. Por questões de significância dos resultados coeficientes de correlação inferiores a 0.00001 foram padronizados para 0.00001.

5. APRESENTAÇÃO DOS RESULTADOS

5.1 INTRODUÇÃO

A apresentação dos resultados demonstrada neste capítulo objetiva promover uma visão do processo de correlação e permitir a avaliação e discussão dos resultados tendo como base informações coletadas a partir da Web. Este capítulo está dividido em três partes, sendo:

- Cenário da Aplicação: Apresenta de maneira geral o cenário informando características da coleta. Declara a abrangência do cenário sobre a base de dados, ou seja, quais tabelas são envolvidas no processo. Promove uma visão geral das possibilidades de análise a partir do modelo e do cenário de aplicação;

- Análise de Perfil: Realiza uma introdução sobre a análise de correlação e o seu resultado visual através de histogramas. Apresenta alguns casos de análise e expõe a análise de termos em um contexto temporal;

- Mapa de Tópicos: Promove uma visão inicial sobre a análise de mapa de tópicos e sua importância como uma ferramenta para entender determinado contexto/domínio de aplicação. Apresenta ainda alguns casos de análise e projeta uma rede específica ao longo do tempo, com intuito de exemplificação.

5.2 CENÁRIO DE APLICAÇÃO

O serviço de correlação foi executado 19 vezes entre 16/05/2012 a 06/06/2012, sendo realizada uma única execução diária. A hora em que o serviço foi executado era variável, porém não há registro desta informação já que não é pertinente para o modelo proposto.

O processo de correlação utilizou 11 conceitos. A lista de conceitos utilizados pode ser observada na Tabela 15.

Termo	Classe	Domínio
Brasileirão	Genérica	Genérico
Crise	Genérica	Genérico
Dia das Mães	Genérica	Genérico
Dia dos Namorados	Genérica	Genérico
Dolar	Genérica	Genérico
Eleições	Genérica	Genérico
Euro	Genérica	Genérico
Grécia	Genérica	Genérico
Londres	Genérica	Genérico
Olimpíadas	Genérica	Genérico
Vestibular inverno	Genérica	Genérico

Tabela 15 - Termos, classes e domínio que compõem o cenário analisado.

O critério para a seleção dos termos foi a dinamicidade, ou seja, o cenário foi escolhido desta maneira objetivando a variação temporal dos termos que o compõem. A construção de um cenário que visa variação temporal é justificada pelo fato de que a análise de correlação é realizada em médio e longo prazo, sendo assim termos dinâmicos podem exemplificar resultados mais próximos da realidade.

Percebe-se que os conceitos escolhidos são eventos sazonais ou estão em evidência na atualidade. O “Dia das mães”, “Dia dos Namorados”, “Eleições”, “Vestibular de inverno”, “Olimpíadas” e “Brasileirão” foram escolhidos por sofrerem influência sazonal. Os conceitos “crise”, “Euro”, “Dólar”, “Grécia” e “Londres” estão em evidência atualmente. Fica claro que o domínio de problema não é bem definido, porém como já visto este domínio é justificado pela capacidade de exemplificação e observação dos resultados obtidos a partir do serviço de correlação.

As classes e os domínios dos termos foram definidos como genéricos devido ao mecanismo de busca utilizado (servidor de consulta) não possuir semântica, tornando assim irrelevante o contexto das palavras. Caso o serviço de consulta possuísse semântica as classes e domínios dos termos seriam definidos de acordo com o contexto das palavras. A Tabela 16 apresenta um conjunto de termos com suas respectivas classes e domínios. Esta tabela representa de maneira mais fiel um domínio de problema que poderia ser utilizado pelo serviço de correlação caso serviço de consulta suportasse busca semântica.

Termo	Classe	Domínio
Ecossistema	Meio Ambiente	Sustentabilidade
Biodiversidade	Meio Ambiente	Sustentabilidade
Recurso natural	Meio Ambiente	Sustentabilidade
Ecovila	Sustentabilidade	Sustentabilidade
Meio ambiente	Meio Ambiente	Sustentabilidade
Desenvolvimento sustentável	Sustentabilidade	Sustentabilidade
Responsabilidade social	Sustentabilidade	Sustentabilidade
Energias renováveis	Sustentabilidade	Sustentabilidade
Sustentabilidade	Sustentabilidade	Sustentabilidade
Créditos de carbono	Sustentabilidade	Sustentabilidade
Protocolo de Quioto	Sustentabilidade	Sustentabilidade
Educação ambiental	Educação	Sustentabilidade
Ecologia urbana	Meio Ambiente	Sustentabilidade
Efeito estufa	Meio Ambiente	Sustentabilidade
Biodegradável	Material	Sustentabilidade
Energia eólica	Energia Renovável	Sustentabilidade
Energia nuclear	Energia Renovável	Sustentabilidade
Energia hidráulica	Energia Renovável	Sustentabilidade
Energia solar	Energia Renovável	Sustentabilidade
Energia geotérmica	Energia Renovável	Sustentabilidade
Combustíveis fósseis	Energia não Renovável	Sustentabilidade
Biocombustível	Energia Renovável	Sustentabilidade

Tabela 16 - Demonstração de um cenário real.

O serviço de correlação utiliza uma lista de conceitos que alimentam o processo de correlação. Como consequência deste processo, a base de dados sofre impacto direto com a inserção dos resultados, ou seja, se modifica constantemente à medida que novas correlações

são executadas. A seguir será realizado um detalhamento das tabelas (fatos e dimensões) que são influenciadas diretamente pelo serviço de correlação considerando a lista de conceitos da Tabela 15:

DI_DATE: cada execução do serviço gera uma nova entrada na tabela. Visto que a análise é temporal esta tabela armazena a data em que o serviço foi executado. A Figura 21 demonstra o conteúdo da tabela DI_DATE.

	time_id integer	day integer	month integer	year integer
25	55	28	5	2012
26	56	28	5	2012
27	57	30	5	2012
28	58	30	5	2012
29	59	31	5	2012
30	27	16	5	2012
31	28	17	5	2012
32	29	18	5	2012
33	30	19	5	2012
34	60	1	6	2012
35	61	1	6	2012
36	62	2	6	2012
37	63	3	6	2012
38	64	4	6	2012
39	65	5	6	2012

Figura 21 - Detalhamento da tabela DI_DATE.

FT_CONCEPT_TIME: aqui são armazenadas as frequências individuais dos conceitos. Como na tabela descrita anteriormente a cada execução do serviço de correlação, novas entradas são inseridas, ou seja, a frequência de cada conceito que faz parte do estudo é armazenada. A Figura 22 demonstra o conteúdo da tabela que está sendo descrita.

	concept_id integer	time_id integer	frequency bigint
536	27	59	2440000
537	27	60	2060000
538	27	61	2060000
539	27	62	2100000
540	27	63	2120000
541	27	64	2110000
542	27	65	2100000
543	27	66	2110000
544	28	34	1440000
545	28	35	1440000
546	28	36	1390000
547	28	37	1340000
548	28	54	1350000
549	28	55	1340000
550	28	56	1340000
551	28	57	1710000

Figura 22 - Detalhamento da tabela FT_CONCEPT_TIME.

FT_RELATION_TIME: esta tabela é atualizada frequentemente durante a execução do serviço de correlação. Considerando a lista de conceitos utilizada (11 conceitos), para cada execução, são geradas 55 novas entradas/tuplas. Apesar d o exemplo anterior não gerar muitas entradas, em um domínio real que poderia conter facilmente 1000 conceitos seriam gerados 499.500 novas entradas a cada execução do serviço de correlação. A Figura 23 representa a tabela que armazena o resultado do processo de correlação.

	source_id integer	target_id integer	relation_id integer	time_id integer	correlation_strength double precision	association_strength integer	joint_frequency bigint
616	28	19	1	34	0.000117298395828934	0	36000
617	31	30	1	34	0.00202825409678639	0	123000
618	17	31	1	34	1e-005	0	231
619	13	33	1	34	0.000580265566478572	0	41400
620	1	4	1	34	0.00164999033534733	0	25500000
621	9	8	1	34	0.00011053929039226	0	30300
622	20	17	1	34	6.81029572259255e-005	0	10400
623	26	18	1	34	7.95514275698942e-005	0	42100
624	21	32	1	34	0.00399099116241951	0	386000
625	15	33	1	34	8.14496980478544e-005	0	37000
626	25	16	1	34	0.000182783998039794	0	36200
627	11	24	1	34	1e-005	0	53300
628	2	29	1	34	1e-005	0	14300
629	14	34	1	34	0.00139438565391529	0	117000
630	7	34	1	34	1.1770297759171e-005	0	15500
631	27	1	1	34	2.98013796681021e-005	0	911000

Figura 23 - Detalhamento da entidade FT_RELATION_TIME.

Basicamente esta tabela possibilita a persistência da frequência conjunta e do coeficiente de correlação. Também é possível observar o campo que permite armazenar o grau de associação. Apesar de oferecer suporte ao cálculo de associação visando trabalhos futuros, o presente trabalho não implementa esta funcionalidade.

As tabelas apresentadas são utilizadas como subsídios para a geração de análises. A seguir serão descritos alguns tipos de análise e representações gráficas que podem ser geradas a parir dos resultados oriundos ao serviço de correlação.

5.3 ANÁLISE DE PERFIL

A análise de perfil avalia o comportamento dos conceitos ao longo do tempo. É possível utilizar para esta análise, a frequência individual, a frequência conjunta e/ou o coeficiente de correlação, entre outras. Com o resultado desta análise é possível gerar representações gráficas com intuito de facilitar o entendimento dos resultados.

A representação que será aplicada no decorrer deste tópico é o histograma. Lopes (1999) afirma que um histograma pode ser utilizado para ilustrar o comportamento de valores agrupados em classes, ou seja, um histograma é gráfico de colunas composto de vários retângulos adjacentes. Este tipo de representação foi utilizado devido ao fato de que através da análise do histograma é possível interpretar informações de forma mais fácil e simples, do que acompanhando uma grande tabela ou um relatório com somente números ou valores. Além de ser adequando para representar grande quantidade de dados. (KUROKAWA; BORNIA, 2002, p. 1).

A seguir serão apresentadas algumas análises representadas por histogramas, assim como, uma discussão dos histogramas gerados.

5.3.1 Frequência Individual

A frequência individual como já apresentado anteriormente representa o número de páginas da internet que possuem o termo, neste caso “Dia das Mães”. O histograma do termo “Dia das Mães” pode ser observado na Figura 24.

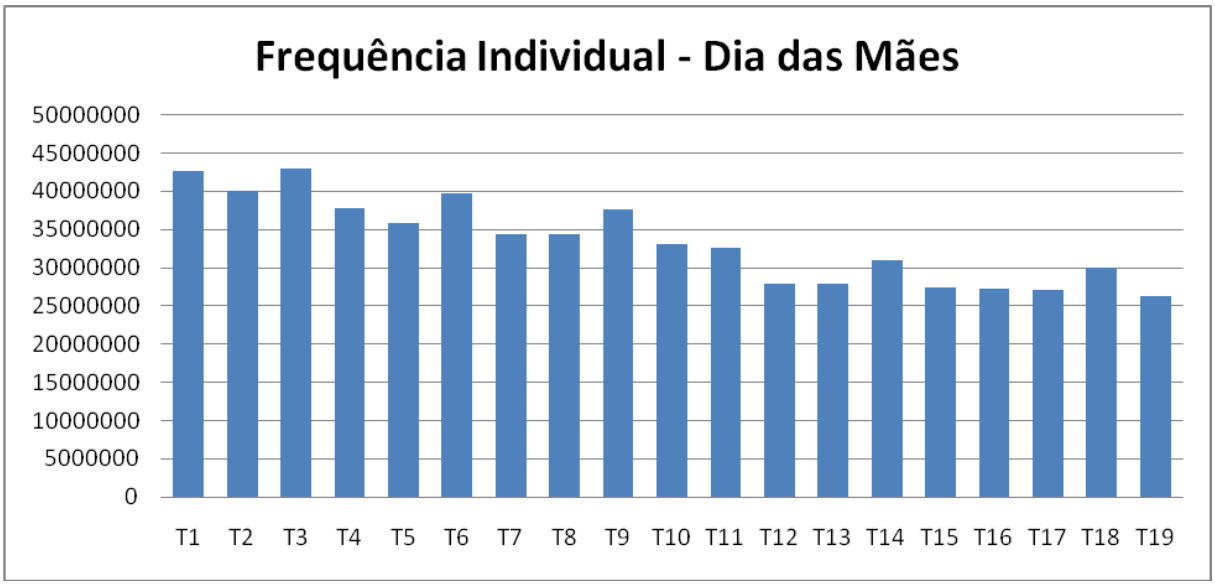

Figura 24 - Histograma da frequência individual do termo Dia das Mães.

A análise de perfil da frequência individual dos termos tem a finalidade de informar a evolução dos termos ao longo do tempo, viabilizando identificação de padrões. Além disso a frêquencia individual é utilizada na geração do coeficiente de correlação, tornando assim uma informação importante para o processo.

5.3.2 Frequência Conjunta

A frequência conjunta é caracterizada por representar o número de páginas que contêm dois termos simultaneamente, neste caso “Crise” e “Grécia”. Através desta informação pode-se gerar um histograma como exemplificado na Figura 25.

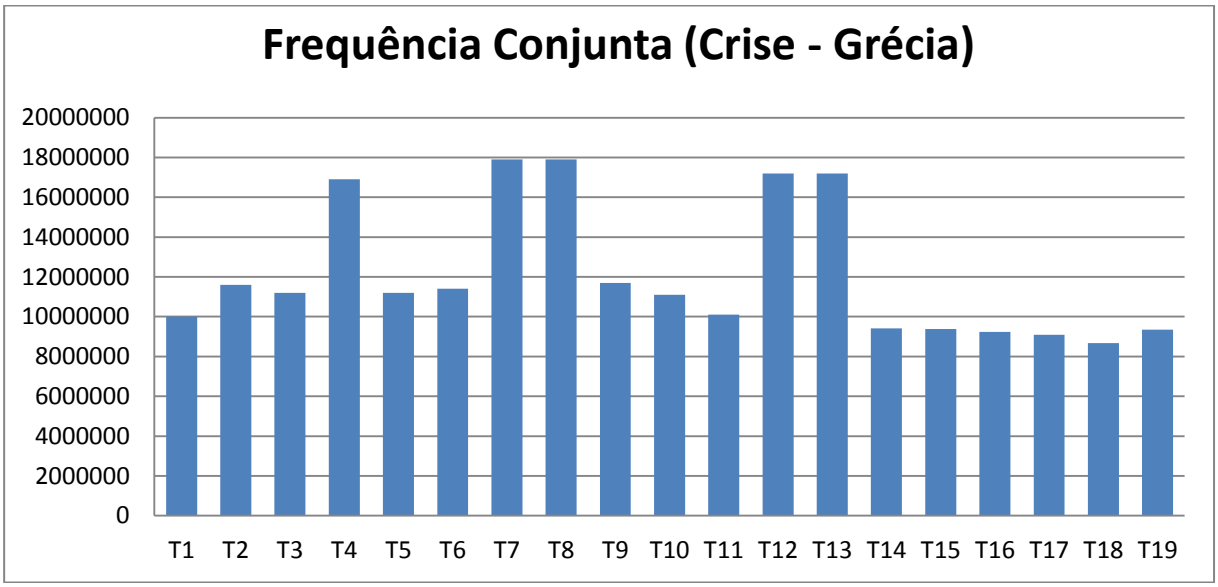

Figura 25 - Histograma da frequência conjunta dos termos Crise e Grécia.

Com o histograma apresentado é possível visualizar a evolução conjunta dos termos utilizados. Assim como a frequência individual, a frequência conjunta também é utilizada no processo de correlação. Com isso se justifica a geração de histogramas a partir das frequências conjuntas.

5.3.3 Correlação entre termos

Outra informação pertinente que pode ser obtida através do processo de correlação diz respeito aos termos mais correlacionados considerando um termo origem em particular. Esta análise é exibida em forma de gráfico, como pode ser observado na Figura 26, que tem “Crise” como termo origem.

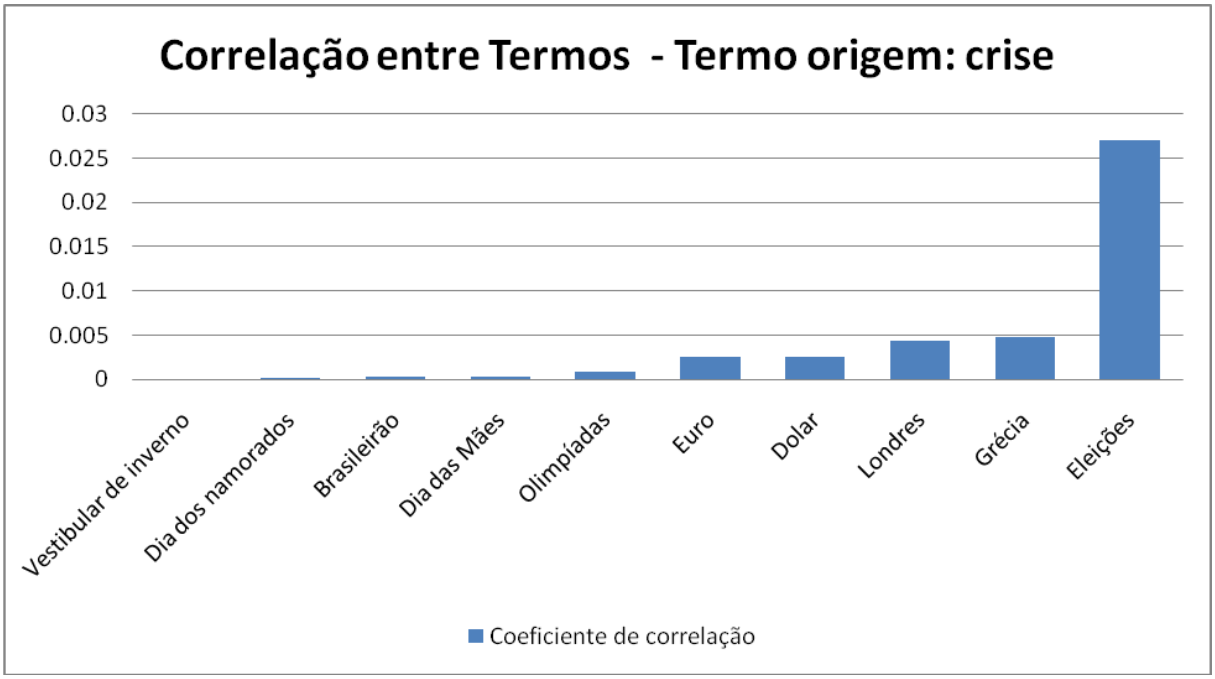

Figura 26 - Análise dos termos mais correlacionados a partir do termo crise.

O gráfico criado é capaz de exibir os termos mais correlacionados gerando um contexto em que o termo origem se encontra. Na seção 4.4 também será apresentada uma forma de apresentar o contexto dos termos através da utilização de mapa de tópicos.

5.3.4 Correlação entre termos (temporal)

Esta análise pode ser considerada a mais significativa das apresentadas. O coeficiente de correlação mede o grau de correlação entre dois termos. Na equação utilizada no protótipo, Phi-squared, são consideradas as frequências individuais e conjuntas descritas anteriormente. A Figura 27 exemplifica um histograma gerado através do coeficiente de correlação entre os termos “Euro” e “Grécia”.

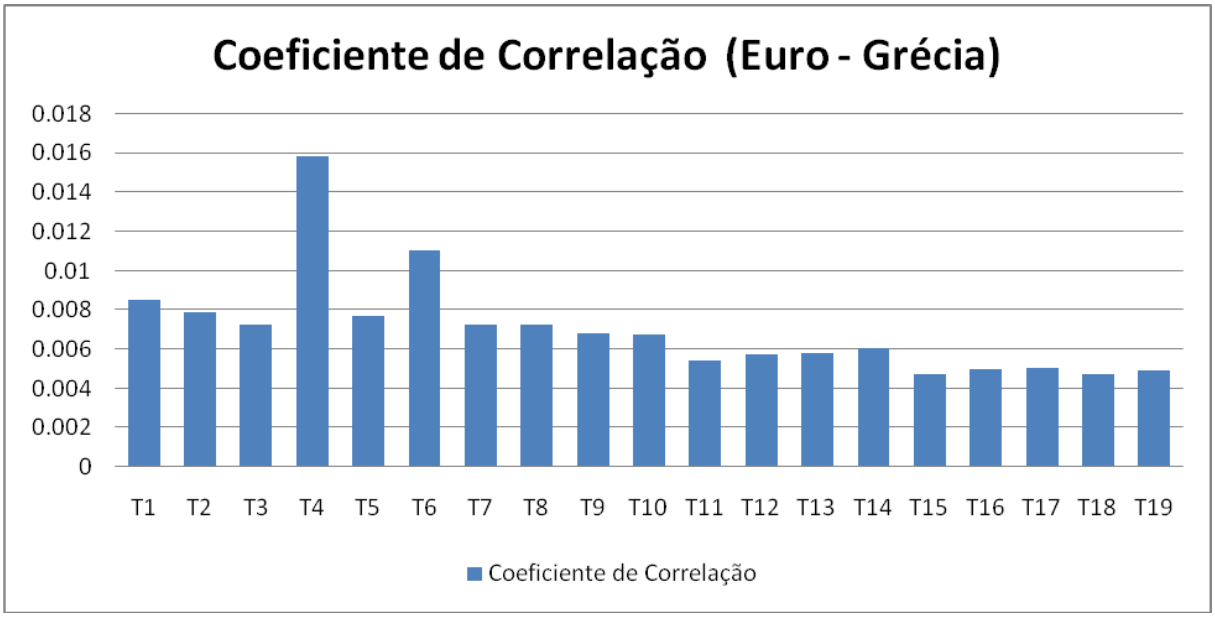

Figura 27 - Histograma do coeficiente de correlação entre os termos Euro e Grécia.

O próximo tópico descreverá com maior detalhamento uma situação real observada durante a fase em que o serviço de correlação foi executado.

5.3.5 Análise dos Resultados Obtidos

Durante o período em que processo foi executado, diversas informações foram acumuladas gerando um grande volume na base de dados. A fim de exemplificação foi selecionado um fato específico para uma análise mais detalhada.

Abaixo segue os histogramas do fato em questão. É pertinente a observação do tempo T4, que se refere ao dia 19/05/2012. Neste dia é possível observar que todos os histogramas apresentados acusam o pico. Os histogramas a seguir referem-se ao coeficiente de correlação, tendo como termo origem “Grécia” e como termos destinos “Eleições” (Figura 28), “Euro” (Figura 29) e “Crise” (Figura 30).

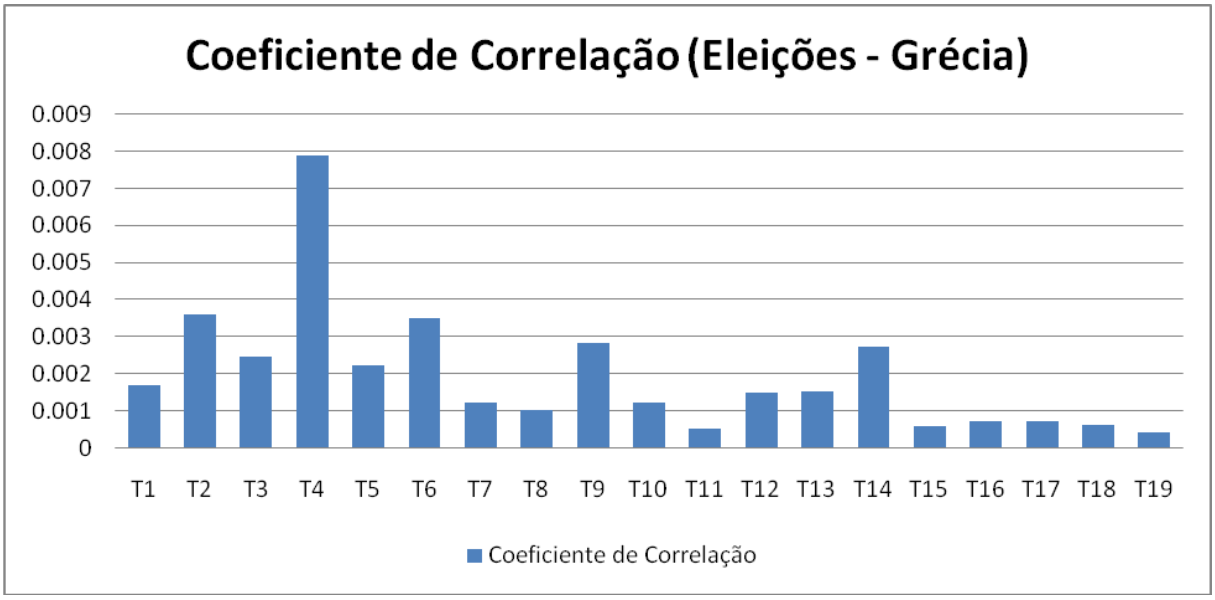

Figura 28 - Histograma do coeficiente de correlação entre os termos Eleições e Grécia.

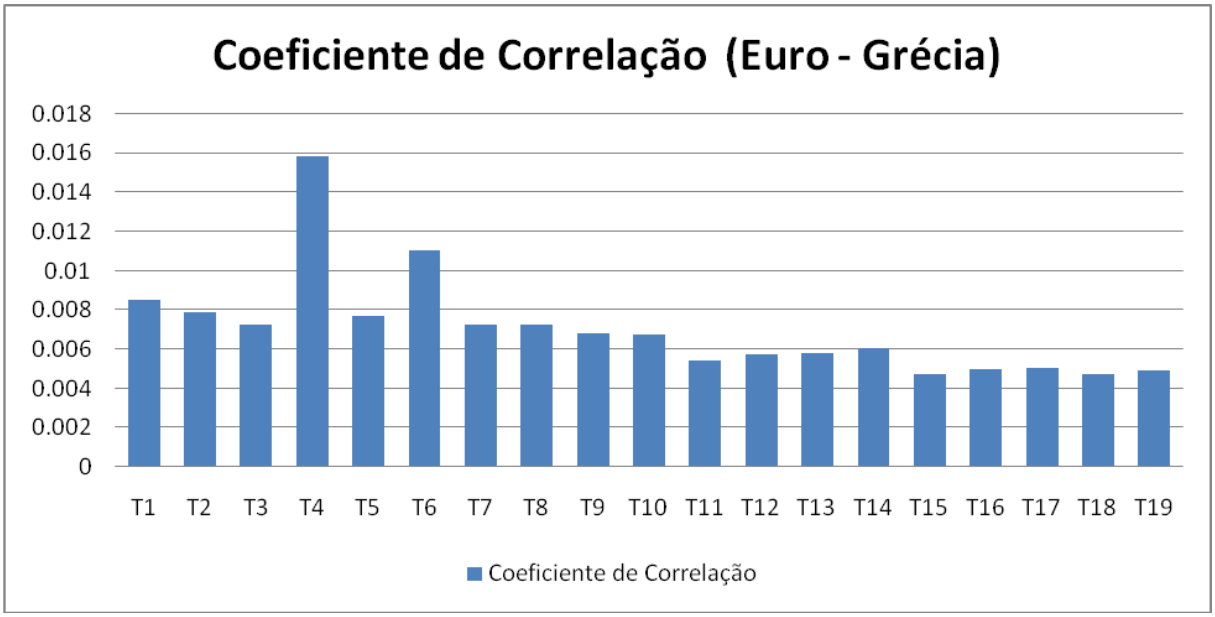

Figura 29 - Histograma do coeficiente de correlação entre os termos Euro e Grécia.

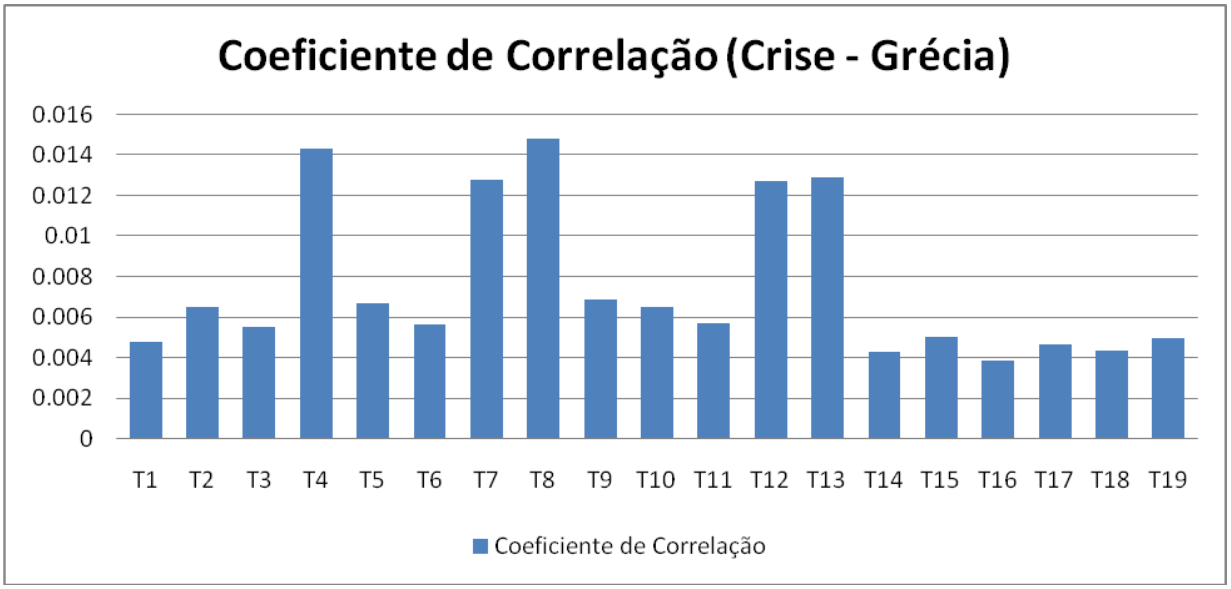

Figura 30 - Histograma do coeficiente de correlação entre os termos Crise e Grécia.

Como já retradado nas Figura 28, 29 e 30 há um pico (T4) em todos os histogramas apresentados. Em uma busca na internet sobre esses termos e o dia em questão, foram encontradas possíveis eventos que possam ter influenciado o valor elevado do coeficiente de correlação. As Figura 31 e 32 ilustram os resultados obtidos a partir da busca manual no servidor de consulta.

Figura 31 - Notícia encontrada no estudo de caso[2].

[2]http://zerohora.clicrbs.com.br/rs/economia/noticia/2012/05/grecia-critica-merkel-por-sugerir-referendo-sobre-o-euro-3763912.html

Figura 32 - Notícia encontrada no estudo de caso[3].

Através da observação de diversas notícias que contêm os termos analisados, encontraram-se algumas possibilidades que justifique o pico que afetou os termos "Crise", "Euro", "Grécia" e "Eleições". Os possíveis eventos que ocasionaram o pico foram:

- Reunião do G8 que, entre outros assuntos, abordou temas referentes à crise dos países da zona do Euro, principalmente a Grécia;
- A possibilidade de que a Grécia saia da zona do euro foi intensificada após a pesquisa eleitoral demonstrar empate entre os partidos da situação e da oposição, visto que a oposição já expôs o desejo de retirar a Grécia da zona do euro.

A partir destes eventos os picos apresentados nos histogramas são justificados e demonstram que os conceitos interagem entre si, formando um cenário dinâmico e complexo. Também é possível afirmar que o serviço de correlação é capaz de capturar variações, com bom grau de precisão, refletindo nos histogramas.

5.4 MAPA DE TÓPICOS

Como já visto anteriormente a correlação entre os termos é dinâmica, ou seja, o contexto de um termo é alterado constantemente ao longo do tempo. Esta característica faz que os termos mais significativos de um termo origem mudem com frequência.

[3]http://www.estadao.com.br/noticias/internacional,g8-defende-permanencia-da-grecia-na-zona-do-euro,875182,0.htm

A análise de redes facilita o entendimento de determinado contexto/domínio de aplicação. Está análise parte de um termo origem e gera um mapa com os termos mais significativos. Os termos mais significativos em relação a uma origem também podem ser analisados, expandindo assim o mapa de tópicos que será gerado.

Como resultado deste mapeamento obtém-se grafos que facilitam a visualização e entendimento do contexto dos termos. Segundo Gersting (1993) “Um grafo é uma representação gráfica de elementos de dados e das conexões entre alguns destes itens”.

A seguir serão apresentadas imagens de mapa de tópicos gerados através do termo Grécia. O exemplo realizado utiliza os três conceitos mais relevantes a partir do termo origem, e é repetido mais uma vez a partir dos termos obtidos anteriormente. Foram produzidas três imagens que representam os mapas de tópicos geradas nos dias 16/05/2012 (Figura 33), 23/05/2012 (Figura 34) e 06/06/2012 (Figura 35).

Figura 33 - Mapa de tópicos gerado a partir do conceito “Grécia” em 16/05/2012.

Figura 34 - Mapa de tópicos gerado a partir do conceito "Grécia" em 23/05/2012.

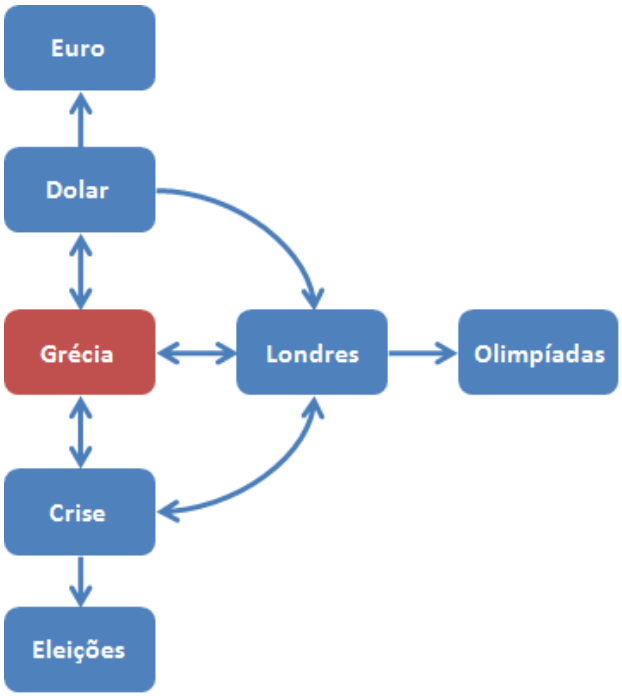

Figura 35 - Mapa de tópicos gerado a partir do conceito "Grécia" em 06/06/2012.

6. CONSIDERAÇÕES FINAIS

O objetivo geral desse trabalho foi desenvolver uma arquitetura que permita, de maneira distribuída, extrair ativos de conhecimento a partir de bases textuais. Nesse sentido, foi realizada uma revisão das áreas de descoberta de conhecimento e computação distribuída visando suportar a proposição do trabalho.

Na base da arquitetura encontra-se o modelo de correlação rápida. Esse modelo procura simplificar a correlação tradicional uma vez que, ao invés de inspecionar todos os documentos para verificar a quantidade de determinado padrão (termo), considera-se somente a quantidade de documentos que mencionaram o termo. Tal abordagem possibilita ganho de desempenho quando aplicado de maneira distribuída. Salienta-se que em função dessa simplificação a precisão do processo de correlação é minimizada. Por outro lado, métodos tradicionais são custosos quando aplicados a grandes bases textuais.

O método de correlação rápida que simplifica o processo de KDT e permite a análise temporal, característica baseada na proposta de Bovo (2011), se mostrou consistente e viabilizou a aplicação do modelo em grandes bases textuais. Com o intuito de melhorar o desempenho e oferecer possibilidades adicionais, o modelo foi implementado de modo que este possa ser executado de forma distribuída através do *framework/middleware* GridGain.

O protótipo desenvolvido atendeu as expectativas gerando resultados satisfatórios e permitindo a produção de análises sobre determinado domínio de aplicação que podem ser expostas através de histogramas, gráficos, grafos, entre outros. A arquitetura distribuída do protótipo demonstrou flexibilidade e escalabilidade podendo ser expandida quando necessário por meio de computadores com hardware e sistemas operacionais distintos.

Outro ponto importante a considerar refere-se ao modelo de dados que suporta o protótipo desenvolvido. Para tal, foi utilizado o conceito de modelagem dimensional em que tabelas são entendidas como dimensões (suporte) e fatos (registros que possuam alguma medida de valor). Esse modelo pode em princípio representar qualquer domínio de aplicação que se baseie em relacionamentos entre conceitos. Possui ainda, como característica importante, a possibilidade de representação de relacionamentos de maneira temporal.

Ao longo deste trabalho surgiram novas possibilidades que não foram desenvolvidas, pois tornariam este trabalho muito extenso. As duas principais possibilidades são a implementação do modelo de associação entre elementos textuais e a adaptação do protótipo com o objetivo de utilizar semântica em seus processos. Apesar destes conceitos não terem sidos acoplados ao protótipo o modelo dimensional foi projetado pensando nestas futuras melhorias.

REFERÊNCIAS

ALAVI, M; COOK, J; COOK, L; LEIDNER, D.E. Review: Knowledge Management and knowledge management systems: Conceptual foundations and research issues. **MIS Quarterly**, v. 25, n. 1, p. 107-136, 2001.

BILLHARDT, Holger; BORRAJO, Daniel; MAOJO, Victor. A context vector model for information retrieval. **Journal Of The American Society For Information Science And Technology**, New York, p.236-249, fev. 2002.

BUYYA, Rajkumar; VENUGOPAL, Srikumar. A Gentle Introduction to Grid Computing and Technologies. **Csi Communications**, S. L, n. 1, p.9-19, 19 jul. 2005.

BARION, Eliana Cristina Nogueira; LAGO, Decio. Mineração de Textos. **Revista de Ciências Exatas e Tecnologia**, Valinhos, Sp, p.123-140, 08 dez. 2008.

BERMAN, Fran; FOX, Geoffrey; HEY, Tony. The Grid: Past, Present, Future. In: BERMAN, Fran; FOX, Geoffrey; HEY, Tony (Editores). **Grid Computing: Making the Global Infrastructure a Reality**. UK: Wiley and Sons, p. 9-50, mar. 2003.

BOVO, Alessandro Botelho. **Um Modelo de descoberta de conhecimento inerente à evolução temporal dos relacionamentos entre elementos textuais.** 155 p. Tese (Doutorado) - Universidade Federal de Santa Catarina, Centro Tecnológico, Programa de Pós-Graduação em Engenharia e Gestão do Conhecimento, Florianópolis, 2011

CECI, Flávio. **UM MODELO SEMIAUTOMÁTICO PARA A CONSTRUÇÃO E MANUTENÇÃO DE ONTOLOGIAS A PARTIR DE BASES DE DOCUMENTOS NÃO ESTRUTURADOS.** 2010. 131 f. Dissertação (Mestrado) - Universidade Federal de Santa Catarina, Florianópolis, 2010.

CHURCH, K. W.; GALE, W. A. **Concordances for Parallel Text**. Proceedings of the Seventh Annual Conference of the UW Centre for the New OED and Text Research. Oxford, England**:** 40-62 p. 1991.

CONRAD, J. G.; UTT, M. H. **A system for discovering relationships by feature extraction from text databases**. Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval. Dublin, Ireland: Springer-Verlag New York, Inc. 1994.

DANTAS, Mario A. R. **Computação distribuída de alto desempenho:** redes, clusters e grids computacionais. Rio de Janeiro: Axcel Books, 2005. 278 p.

DEITEL, Harvey M.; DEITEL, Paul J. **Java como programar.** 8. ed. São Paulo (SP): Pearson Prentice Hall, 2010. xxix, 1144 p.

EGGHE, L.; MICHEL, C. Strong similarity measures for ordered sets of documents in information retrieval. **Information Processing and Management: an International Journal**, v. 38, n. 6, p. 823-848, 2002.

ELMASRI, Ramez; NAVATHE, Shamkant B. **Sistemas de banco de dados.**4. ed.São Paulo (SP): Pearson Addison Wesley, 2005. 724p.

FAYYAD, U. M. Data Mining and Knowledge Discovery: Making Sense Out of Data. **IEEE Expert: Intelligent Systems and Their Applications,** v. 11, n. 5, p. 20-25, 1996.

FAYYAD, U. M.; PIATETSKY-SHAPIRO, G.; SMYTH, P. From data mining to knowledge discovery: an overview. In: (Ed.). **Advances in knowledge discovery and data mining**: American Association for Artificial Intelligence, 1996. p.1-34.

GANIZ, M. C.; POTTENGER, W. M.; JANNECK, C. D. Recent Advances in Literature Based Discovery. **Journal of the American Society for Information Science and Technology, JASIST**, 2006.

GERSTING, Judith L. **Fundamentos matemáticos para a ciência da computação:** um tratamento moderno de matemática discreta. 5. ed Rio de Janeiro (RJ): LTC, 2004. xiv, 597p.

GÓES, L. F. W.; NETO, D. O. G.; FERREIRA, R.; CIRNE, W. Computação em Grade: Conceitos, Tecnologias, Aplicações e Tendências. In: ERI-MG, 5., 2005, Lavras Mg. **ESCOLA REGIONAL DE INFORMÁTICA DE MINAS GERAIS.** Lavras Mg: Anais, 2005. Disponível em: <http://www.ppgee.pucminas.br/lsdc/publicacoes.html>. Acesso em: 01 mar. 2012.

GONÇALVES, Alexandre L. ; UREN, Victoria ; KERN, Vinícius Medina ; PACHECO, Roberto C S . Mining Knowledge from Textual Databases: An Approach using Ontology-based Context Vectors. In: **International Conference on Artificial Intelligence and Applications (AIA 2005)**, 2005, Innsbruck. Proceedings of the 23rd IASTED International Multi-Conference on Artificial Intelligence and Applications. Innsbruck, 2005. p. 66-71.

GONÇALVES, Alexandre Leopoldo. **Um modelo de descoberta de conhecimento baseado na correlação de elementos textuais e expansão vetorial aplicado à engenharia e gestão do conhecimento.** Florianópolis, SC, 2006. 196 f. Tese (Doutorado) - Universidade Federal de Santa Catarina, Centro Tecnológico. Programa de Pós-Graduação em Engenharia de Produção.

HILBERT, Martin; LÓPEZ, Priscila. The World's Technological Capacity to Store, Communicate, and Compute Information. **Science,** [S. l.], p. 60-65. 01 abr. 2011. Disponível em: <http://www.sciencemag.org/content/332/6025/60.full>. Acesso em: 17 fev. 2012.

INMON, Willian H. **Builging the data warehouse.** 3rd ed New York: J. Wiley, 2002. 412p.

IVANOV, Nikita. **Real Time Big Data Processing with GridGain**, 2012**.** Disponível em: <http://www.gridgain.com/book/book.html>. Acesso em: 20 maio 2012.

JONES, W. P.; FURNAS, G. W. Pictures of relevance: a geometric analysis of similarity measures. **Journal of the American Society for Information Science,** v. 38, n. 6, p. 420-442, 1987.

KORPELA, Eric J. et al. Status of the UC-Berkeley SETI Efforts. In: INSTRUMENTS, METHODS, AND MISSIONS FOR ASTROBIOLOGY, v. 14, 2011, San Diego. **Proceedings...** San Diego: Spie, 2011.

KUROKAWA, Edson; BORNIA, Antonio Cezar. Utilizando o histograma como uma ferramenta estatística de análise da produção de água tratada de Goiânia. In: XXVIII CONGRESO INTERAMERICANO DE INGENIERÍA SANITARIA Y AMBIENTAL, 28., 2002, Goiânia. **Anais... .** Cancun: Congreso Interamericano de Ingeniería Sanitaria y Ambiental, 2002.

LIRA, Sachiko Araki. **ANÁLISE DE CORRELAÇÃO:** ABORDAGEM TEÓRICA E DE CONSTRUÇÃO DOS COEFICIENTES COM APLICAÇÕES. 2004. 209 f. Dissertação (Mestrado) - UFPR, Curitiba, 2004

LOPES, Paulo Afonso. **Probabilidades & Estatística**. 1. ed. Rio de Janeiro: R&A, 1999, 174 p.

LOUDON, Kyle. **Desenvolvimento de grandes aplicações Web.** São Paulo (SP): Novatec, 2010. 325 p.

LYMAN, Peter; VARIAN, Hal R. **How much information**? Executive summary. 2003.

MATTSSON, Michael. **Evolution and Composition of Object-Oriented Frameworks.** 2000. 231 f. Tese (Doutorado) - Departamento de Department Of Software Engineering And Computer Science, University Of Karlskrona/ronneby, Karlskrona, 2000.

MOLIK, E. Establishing Moore's Law. **Annals of the History of Computing**, v. 28, n. 3, p. 62 – 75, 2006.

MOONEY, Raymond J.; NAHM, Un Yong. Text Mining with Information Extraction. In: INTERNATIONAL MIDP COLLOQUIUM DAELEMANS, 4., September 2003, Bloemfontein, South Africa. W., du PLESSIS, T., SNYMAN, C. and TECK, L. (Eds.).**Proceedings…** Bloemfontein, South Africa: Van Schaik Pub., 2005. p.141-160.

NONAKA, I.; TAKEUCHI, H. **The knowledge-creating company**: How japanese companies create the dynamics of innovation, Oxford, UK: Oxford University Press, 1995.

NURSEITOV, Nurzhan et al. Comparison of JSON and XML Data Interchange Formats: A Case Study. In: INTERNATIONAL CONFERENCE ON COMPUTER APPLICATIONS IN INDUSTRY AND ENGINEERING, 22., 2009, Bozeman. **Proceedings...** San Francisco, California: Caine, 2009. p. 157 - 162.

PACKER, Abel Laerte; TARDELLI, Adalberto Otranto; CASTRO, Regina Célia Figueiredo. A distribuição do conhecimento científico público em informação, comunicação e informática em saúde indexado nas bases de dados MEDLINE e LILACS. **Ciênc. saúde coletiva**, Rio de Janeiro, v. 12, n. 3, jun. 2007.

RAJMAN, M.; BESANÇON, R. Text mining: Natural language techniques and text mining applications. In: **IFIP TC2/WG2.6 WORKING CONFERENCE ON DATABASE SEMANTICS (DS-7)**, 7., 1997, Leysin. **Proceedings...** .Chapman & Hall, 1997. p. 7 - 10.

RAMOS, Hélia de Sousa Chaves; BRASCHER, Marisa. Aplicação da descoberta de conhecimento em textos para apoio à construção de indicadores infométricos para a área de C&T. **Ci. Inf.**, Brasília, v. 38, n. 2, ago. 2009.

RUBBO, B. Fernando. **Um estudo do framework de grid GridGain**. Porto Alegre, RS, 8 p. Trabalho não publicado. Disponível em: <https://saloon.inf.ufrgs.br/twikidata/Disciplinas/CMP157/TF07FernandoRubbo/TF07FernandoRubboArtigo.pdf > Acesso em 14 abr. 2010.

SALTON, G.; BUCKLEY, C. Therm-weighting approachs in automatic text retrieval. **Information Processing & Management**, v. 24, n. 5, p. 512-523, 1988.

SCHREIBER, G. et al. **Knowledge Engineering and Management: The CommonKADS Methodology**. Cambridge, Massachusetts: The MIT Press, 2002.

SILBERSCHATZ, Abraham.; GALVIN, Peter B.; GAGNE, Greg. **Fundamentos de sistemas operacionais.** 8. ed. Rio de Janeiro: LTC, 2010. xvii,515p.

SILVA, E. R. G.; ROVER, A. J. O Processo de descoberta do conhecimento como suporte à análise criminal: minerando dados da Segurança Pública de Santa Catarina. In: **International Conference on Information Systems and Technology Management**, 2011, São Paulo. Anais da International Conference on Information Systems and Technology Management. São Paulo: FEA, 2011. v. 8.

SILVA, M P. Mineração de Dados: Conceitos, Aplicações e Experimentos com Weka. In: ESCOLA REGIONAL DE INFORMÁTICA RJ/ES, 4., 2004, Rio Das Ostras, RJ. **Anais... .** São José Dos Campos, SP. Sociedade Brasileira de Computação, 2004.

STALLINGS, William. **Arquitetura e organização de computadores.** 8. ed. São Paulo (SP): Pearson, 2010. xiv, 624p.

STEVENSON, William J. **Estatística aplicada a administração.** São Paulo: HARBRA, 2001. 495p.

TANENBAUM, Andrew S. **Sistemas operacionais modernos.** 3. ed. Rio de Janeiro (RJ): Prentice-Hall do Brasil, 2010. xiii, 653p.

TANENBAUM, Andrew S.; STEEN, Maarten van. **Sistemas distribuídos:** princípios e paradigma. 2. ed. São Paulo: Pearson Prentice Hall, 2007. x, 402 p.

TOP500. **TOP 10 Sites for June 2012.** Disponível em: <http://www.top500.org/lists/2012/06>. Acesso em: 17 jun. 2012.

TRIOLA, Mario F. **Introdução á estatística.** 10. ed. Rio de Janeiro (RJ): LTC, 2008. xxiii, 656p.

TRONCHONI, Alex B.; PRETTO, Carlos O.; ROSA, Mauro A. da.; LEMOS, Flávio A. Becon. **Descoberta de conhecimento em base de dados de eventos de desligamentos de empresas de distribuição**. Sba Controle & Automação. 2010, vol.21, n.2, p. 185-200.

VIANNA, Rossana Cristina Xavier Ferreira et al . Mineração de dados e características da mortalidade infantil. **Cad. Saúde Pública**, Rio de Janeiro, v. 26, n. 3, Mar. 2010. Disponível em: < http://dx.doi.org/10.1590/S0102-311X201000030001>. Acesso em 03 jul. 2012.

WEEBER, M. Advances in Literature-based Discovery. **Journal of the American Society for Information Science and Technology,** v. 54, n. 10, p. 913-925, 2003.

WEEBER, M. et al. Using concepts in literature-based discovery: simulating Swanson's Raynaud-fish oil and migraine-magnesium discoveries. **Journal of the American Society for Information Science and Technology,** v. 52, n. 7, p. 548-557, 2001.

WILKS, Yorick; CATIZONE, Roberta. Can We Make Information Extraction More Adaptive? In: SCIE99 WORKSHOP, 1999, Sheffield. **Proceedings... .** Berlin: Springer-verlag, 1999. p. 1 - 16.

WILSON, T.D. The nonsense of knowledge management. **Information Research**, v. 8, n. 1, Out.de 2002.

WIVES, Leandro Krug, LOH, Stanley. Tecnologias de descoberta de conhecimento em informações textuais; ênfase em agrupamento de informações. In: OFICINA DE INTELIGENCIA ARTIFICIAL (OIA) III, 1999, Pelotas (RS). **Anais…** . Pelotas: EDUCAT, 1999. p. 28-48.