Vilmar César Pereira Júnior

Recuperação Associativa de Recursos de Informação Utilizando Spreading Activation

Florianópolis

2011

Vilmar César Pereira Júnior

Recuperação Associativa de Recursos de Informação Utilizando Spreading Activation

Trabalho de Conclusão de Curso apresentado como requisito parcial para obtenção do grau de Bacharel em Ciência da Computação na Universidade Federal de Santa Catarina.

Orientador:
Prof. Dr. Renato Fileto

Co-orientador:
Wanderson Rigo

UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA
CIÊNCIA DA COMPUTAÇÃO

Florianópolis

2011

O presente Trabalho de Conclusão de Curso foi aprovado como requisito parcial para obtenção do grau de Bacharel em Ciência da Computação na Universidade Federal de Santa Catarina sob o título *"Recuperação Associativa de Recursos de Informação Utilizando Spreading Activation"*, elaborado por Vilmar César Pereira Júnior e aprovado em 2011, em Florianópolis, Estado de Santa Catarina, pela banca examinadora constituída pelos membros:

Prof. Dr. Renato Fileto
Orientador

Msc. Wanderson Rigo
Co-orientador

Profa. Dra. Carina Friedrich Dorneles
Universidade Federal de Santa Catarina

Prof. Dr. Ronaldo dos Santos Mello
Universidade Federal de Santa Catarina

AGRADECIMENTOS

Primeiramente agradeço a Deus por prover sabedoria e discernimento às pessoas.

Aos meus pais, Vilmar e Araci, pela dedicação, apoio e carinho empregados em toda a minha formação, mesmo nos momentos mais difíceis.

A minha esposa Gabriela, pelo companheirismo, amor e cumplicidade em todas as horas em que não estive tão presente.

Ao meu irmão Adriano, pelo incentivo e exemplo de dedicação.

Ao professor Renato Fileto por ter dado o desafio inicial e a oportunidade de desenvolver este trabalho.

Ao co-orientador Wanderson Rigo, pelos conselhos e ajuda proporcionada durante todo este trabalho.

Aos meus amigos Franklin, Samuel e Lizandro, pela amizade irrestrita e momentos de descontração.

LISTA DE ABREVIATURAS

AS	Associação Semântica.
AVEA	Ambiente Virtual de Ensino e Aprendizagem.
DeCS	Descritores em Ciências da Saúde
IR	Recuperação de Informação (*Information Retrieval*).
OA	Objeto de Aprendizagem.
RDF	*Resource Description Framework*.
RI	Recurso de Informação.
RS	Rede Semântica.
SA	Ativação por Espalhamento (*Spreading Activation*).
SAN	*Spreading Activation Network*.
UnA-SUS	Universidade Aberta do Sistema Único de Saúde.
VC	Vocabulário Controlado.
W3C	*World Wide Web Consortium*.

RESUMO

O crescimento do volume de dados da *Web* faz com que mecanismos de buscas tradicionais apresentem limitações na recuperação de recursos de informação semanticamente relacionados, exigindo técnicas mais avançadas. O enfoque da *Web* Semântica apresenta possibilidades de se aprimorar o processo de recuperação de informação, de maneira a utilizar o potencial semântico dos termos buscados. Para isso este trabalho realiza um levantamento bibliográfico a respeito dos conceitos que fundamentam a *Web* Semântica e os processos de recuperação de informação.

O trabalho objetiva desenvolver um mecanismo de buscas semânticas baseado em uma técnica chamada *Spreading Activation* (SA). Para cumprir este objetivo é necessário compreender a técnica de *Spreading Activation* original e adaptações propostas por outros trabalhos para finalmente definir uma proposta de adaptação do SA em nosso trabalho.

Formalizada a proposta de adaptação do SA, será desenvolvido um protótipo de execução deste algoritmo, que deve ser incrementalmente refinado e adicionado ao CIBELE, um sistema de catalogação, recuperação e apresentação de recursos, desenvolvido no Laboratório para Integração de Sistemas e Aplicações Avançadas - LISA. Tal protótipo será avaliado no contexto das necessidades de catalogação e recuperação de objetos de aprendizagem da Universidade Aberta do SUS (UnA-SUS).

Palavras-chave: Buscas Semânticas, Recuperação de Informação, *Web* Semântica, Ontologias, *Spreading Activation*.

ABSTRACT

The volume of data from Web makes traditional search engines have limitations in recovery of semantically related information resources, requiring more advanced techniques. The focus of the Semantic Web presents opportunities to improve the process of information retrieval in order to utilize the potential of semantic search terms. For this, this paper surveys the literature on the concepts that underlie the Semantic Weband the processes of information retrieval.

The work aims at developing a semantic search engine based in the Spreading Activation (SA) algorithm. To achieve this it is necessary to understand the technique of original Spreading Activation and adjustments proposed by other studies to ultimately define a proposal to adapt our work in SA.

Formalized the proposal to adjust the SA, will develop a prototype implementation of this algorithm, which must be incrementally refined and added to CIBELE, a cataloging system, retrieval and presentation of resources, developed at the Laboratory for Systems Integration and Advanced Applications - LISA .

This prototype will be evaluated in the context of the needs of cataloging and retrieval of learning objects from the Open University of SUS (UNA-SUS). Keywords: Semantic Search, Information Retrieval, Semantic Web, Ontologies, Spreading Activation.

SUMÁRIO

LISTA DE FIGURAS

1 INTRODUÇÃO

O crescente volume de recursos de informação (textos, imagens, vídeos, etc.) disponíveis atualmente na *Web* ou mesmo em grandes repositórios com acesso controlado demanda processos de recuperação de informação cada vez mais sofisticados (Baeza-Yates e Ribeiro-Neto 1999). O objetivo dos sistemas de busca é recuperar eficientemente todos (cobertura) e somente (precisão) os recursos que satisfaçam os critérios de busca do usuário (Mangold 2007).

Os mecanismos de busca de propósito geral (tais como *Google*, *Bing* , etc..) permitem a recuperação eficiente de informação da *Web*, mas apresentam deficiências de cobertura e precisão. Essas limitações devem-se à natureza desses mecanismos, que se baseiam em técnicas fundamentadas principalmente nos aspectos léxicos e sintáticos das palavras-chave (*keywords*) utilizadas na especificação das buscas.

Esses mecanismos, embora amplamente utilizados, muitas vezes não satisfazem as necessidades dos usuários em domínios específicos (e.g., medicina, biologia, engenharia, direito, etc.). Profissionais de domínios específicos costumam utilizar vocabulários especializados. Eles freqüentemente precisam obter e reusar recursos de informação que podem ser referenciados em buscas por uma variedade de palavras-chave, com denotação específica na área de conhecimento. Sendo assim, os aspectos semânticos devem ser levados em conta, possibilitando a recuperação de recursos descritos com termos relacionados às palavras-chave utilizadas na busca, apesar destes termos não apresentarem necessariamente correlações léxicas com as palavras utilizadas na busca.

Buscas semânticas (Mangold 2007) levam em consideração o significado dos termos pesquisados. Elas procuram tirar vantagem do conhecimento (possivelmente de um domínio específico), descrito em estruturas como ontologias, para auxiliar na recuperação de informação. A *Web Semântica* (“*Web 3.0*”) (Berners-Lee T.; Hendler e Lassila 2001) propõe a descrição (anotação semântica) dos recursos de informação utilizando ontologias e o uso de inferências sobre essas ontologias e anotações semânticas para uma variedade de aplicações, incluindo bus-

cas semânticas. Esta nova *Web*, definida pelo *World Wide Web Consortium - W3C*[1], deve aos poucos substituir a *Web* atual. Mas, para que isso ocorra, os novos sistemas precisam ser desenvolvidos com funcionalidades baseadas em semântica, processando as buscas de acordo com o significado preciso atribuído aos termos usados para referenciar os recursos de informação desejados.

O trabalho de (Guha, McColl e Miller 2003) expõe a importância de buscas semânticas:

> "Search is both one of the most popular applications on the Web and an application with significant room for improvement. We believe that the addition of explicit semantics can improve search. Semantic search attempts to augment and improve traditional search results (based on Information retrieval technology) by using data from the Semantic Web."(Guha, McColl e Miller 2003, p. 702)

O processamento de buscas semânticas requer técnicas avançadas. Este trabalho utiliza a técnica de "Espalhamento por Ativação" (*Spreading Activation* - SA) formalmente descrita em (Crestani 1997) e (Androutsopoulos, Tsatsaronis e Vazirgiannis 2007). Tal técnica explora os relacionamentos entre termos semanticamente próximos em uma ontologia ou vocabulário controlado, para expandir semanticamente as consultas a partir de termos com correspondências léxicas com a palavras-chave dessas consultas. Ela segue a metáfora de uma pedra lançada em um lago, onde as ondas vão perdendo a força à medida em que se afastam do centro de colisão da pedra (a pedra pode ser vista como uma palavra-chave informada). Esta técnica permite a recuperação de recursos anotados semanticamente com termos relacionados às palavras-chave utilizadas na busca e a ordenação dos resultados por uma medida de relevância semântica. Este algoritmo atua sobre uma estrutura definida por uma ontologia (conceitos e instâncias, referenciados por termos, juntamente com relações semânticas entre esses) e anotações semânticas (associações entre termos da ontologia e recursos de informação, utilizadas para descrevê-los). Esta estrutura é uma especialização da rede semântica definida por Crestani como **rede associativa**.

O problema de busca semântica sobre a rede associativa pode ser formalmente definido da seguinte maneira: dado um conjunto de palavras-chaves de consulta K, retornar uma lista de pares $R = (r, \rho)$ ordenada (ranked) de maneira decrescente segundo os valores de ρ, onde: r é um recurso de informação anotado na rede associativa com um ou mais termos de um conjunto $K' \supseteq K$ de termos visitados por um conjunto de percursos na rede associativa partindo dos termos em K; e ρ é uma função de relevância do respectivo recurso de informação r para K, a qual deve maximizar a precisão e a cobertura de cada subconjunto de resultado definido como porção anterior de R.

[1] http://www.w3c.org/

1.1 OBJETIVO

O objetivo deste trabalho é desenvolver e testar um mecanismo de buscas semânticas baseado em *Spreading Activation* que apresente resultados ordenados por sua relevância semântica para as palavras-chave pesquisadas. O propósito é otimizar o processo de recuperação de recursos de informação em um domínio específico, em termos de um bom contrato de compromisso entre o tempo de execução do processo e a cobertura e a precisão dos resultados obtidos.

1.1.1 OBJETIVOS ESPECÍFICOS

Os objetivos específicos deste trabalho são:

1. Pesquisar, estudar, avaliar e definir abordagens adequadas para o processamento de buscas semânticas. O intuito é a compreensão, avaliação e adaptação do mecanismo de buscas por *Spreading Activation*.

2. Implementar o mecanismo de busca semântica proposto e integrá-lo ao repositório DSpace, inclusive na ordenação dos resultados quanto à relevância semântica para palavras-chave e valores de outros campos de metadados (título, autor, etc.).

3. Validar o mecanismo proposto. Para isto, foi desenvolvido um estudo de caso usando um vocabulário controlado da área de saúde (*DeCS - Descritores em Ciências da Saúde*)[2] e realizados experimentos com usuários da UnA-SUS[3].

1.1.2 CARACTERÍSTICAS FUNCIONAIS E REQUISITOS

A proposta defendida por este trabalho deve atender as necessidades dos usuários, de forma que contemple as seguintes características:

- Modo de especificação de consulta simples e intuitivo: o sistema deve permitir que o usuário realize as buscas de forma intuitiva e ágil, sem necessitar de habilidades sofisticadas com conectores lógicos (AND, OR, XOR, etc.). Assume-se que o usuário não possua nenhuma formação em computação ou lógica, mas apenas conhecimentos básicos em informática.

[2] http://decs.bvs.br/
[3] http://unasus.ufsc.br/

- Solução customizável para diferentes domínios de aplicação: nosso estudo de caso está focado na área da saúde, porém o mecanismo desenvolvido deve ser facilmente adaptável a outros domínios. Os algoritmos são parametrizados de maneira a permitir a execução de testes com diversas configurações de parâmetros, possibilitando assim que várias combinações sejam testadas de forma a se alcançar os melhores resultados na recuperação da informação, independente de domínio de aplicação.

- Resultados relevantes e abrangentes: o sistema deve apresentar resultados ordenados por sua relevância em relação às palavras-chave utilizadas nas consultas. Assim, pode retornar grande número de resultados, sem no entanto importunar o usuário, que pode visualizar os resultados mais relevantes no topo da lista.

O mecanismo de buscas semânticas desenvolvido deve atender os seguintes requisitos:

- Permitir ao usuário estipular consultas por palavras-chave e mediante navegação na ontologia ou em uma visão da mesma. Através de uma interface onde poderá escolher as palavras-chave, além de especificar valores para outros campos de metadados, como título e autor;

- Seguir a arquitetura definida pelo grupo de pesquisa e desenvolvimento, utilizando ferramentas já implantadas no repositório UnA-SUS da UFSC, valendo-se de ferramentas e algoritmos de código aberto, bem documentados e embasando as escolhas através de fundamentação científica

O processo de desenvolvimento de software segue uma metodologia interativa e incremental com prototipação. Os resultados das buscas são ordenados (*ranked*) por relevância, utilizando métricas que combinem os resultados obtidos com as buscas sintática e semântica.

O SA é configurado de maneira a explorar a potencialidade semântica da rede associativa, com a finalidade de se ampliar a cobertura e a relevância dos resultados em comparação a uma busca tradicional. Essas idéias são implementadas e validadas em um repositório de Objetos de Aprendizagem (uma forma de recurso de informação) da área da saúde, no contexto do projeto *UnA-SUS* [4], caracterizando assim um estudo de caso.

[4] http://unasus.ufsc.br/

1.2 MOTIVAÇÃO

A popularização da Internet em todos os segmentos da sociedade gerou uma nova demanda de serviços que utilizam a web para compartilhar. Esta informação inicialmente era distribuída de forma desordenada pela web, nos mais diversos formatos e repositórios.

Em vista do crescimento exponencial da produção desses recursos de informação disponíveis e potencialmente melhor organizados, a necessidade passou a ser a **recuperação** de tais conteúdos. Métodos tradicionais de busca foram desenvolvidos (Baeza-Yates e Ribeiro-Neto 1999) (Manning, Raghavan e Schtze 2008), porém os níveis de cobertura e precisão desses métodos são limitados em diversas situações pela dificuldade de identificar significados precisos e correlacionar termos semanticamente relacionados com a palavra-chave informada. Os trabalhos de (Mangold 2007) e (Guha, McColl e Miller 2003) afirmam que esta característica dos buscadores de propósito geral motivou a criação de mecanismos semânticos, os quais podem atuar sobre as estruturas de representação de conhecimento como ontologias e anotações semânticas (Berners-Lee T.; Hendler e Lassila 2001) (Guha, McColl e Miller 2003) (Han e Reeve 2005).

Na área de saúde, constatou-se uma necessidade dos usuários da UnA-SUS de recuperar recursos já existentes no repositório e reusá-los pra produzir cursos com baixo custo e de forma rápida, principalmente para atender demandas emergenciais da saúde. Com o objetivo de auxiliar na melhoria da recuperação de recursos realizou-se um estudo de caso na área da saúde, utilizando o DeCS e visando produzir um sistema de buscas semânticas.

O trabalho busca desenvolver um sistema para processamento de buscas semânticas sobre redes associativas que permita: (i) adequação a diversos domínios, mediante a troca da ontologia utilizada; (ii) o ajuste de diversos parâmetros dos algoritmos utilizados para sua otimização em diferentes domínios de aplicação; (iii) facilite a execução de experimentos para calibração e validação dos resultados obtidos; (iv) obter um sistema customizado e integrado ao DeCS para atender necessidade específicas da UnA-SUS.

1.3 METODOLOGIA

A metodologia deste trabalho está fundamentada nos seguintes passos:

1. Revisão bibliográfica: descrever o que está sendo feito atualmente, trabalhos relacionados à proposta e o que foi utilizado no desenvolvimento;

2. Elaboração dos algoritmos de busca : entendimento do *Spreading Activation* e da estrutura (termos e relações) do vocabulário controlado do DeCS. Também são definidas heurísticas para o *ranking* dos resultados. O algoritmo possui parâmetros bem definidos para o andamento e a parada do processo, com a possibilidade de se variar estes parâmetros e observar o comportamento dos resultados obtidos através de experimentos rigorosamente executados e documentados;

3. Estudo de caso: entender a arquitetura em que o mecanismo de buscas será acoplado para validar os testes;

4. Implementação do protótipo: foi implementado um protótipo do mecanismo de buscas. Esta versão possui uma interface que permite a seleção dos valores dos diversos parâmetros do algoritmo implmentado, permitindo assim ajustes do método em experimentos; Foi utilizada a biblioteca *Texai Spreading Activation*[5], desenvolvida em Java sob a licença *GPL (GNU General Public License)*, possibilitando assim que eventuais melhorias no algoritmo de SA sejam implementadas, sobre um código já testado e otimizado;

5. Experimentos: os protótipos iniciais foram testados por membros do LISA, e versões mais elaboradas serão submetidas a testes com usuários finais (em trabalhos futuros). Estes testes comparativos têm o objetivo de aferir o comportamento do mecanismo desenvolvido em relação a desempenho (tempo e consumo de CPU) e recursos obtidos (resultados das buscas).

1.4 ESTRUTURA DO TRABALHO

O trabalho apresenta 5 capítulos subsequentes, definidos da seguinte maneira:

- Capítulo 2: este capítulo faz uma revisão bibliográfica, contextualizando o problema abordado e apresentando conceitos básicos da Web Semântica;

- Capítulo 3: apresenta os fundamentos e a justificativa para a escolha do algoritmo de *Spreading Activation*;

- Capítulo 4: apresenta a proposta do mecanismo de buscas implementado, contextualizando o escopo em que ele foi desenvolvido;

- Capítulo 5: apresenta a implementação da proposta definida no capítulo anterior, detalhando o protótipo do algoritmo de *Spreading Activation* desenvolvido;

[5] http://sourceforge.net/projects/texai/

- Capítulo 6: define a metodologia adotada nos experimentos e os resultados finais;
- Capítulo 7: apresenta conclusões e perspectivas de trabalhos futuros.

1.5 TRABALHOS RELACIONADOS

O trabalho de (Crestani 1997) apresenta um *survey* do *Spreading Activation*, apresentando o modelo original, origem histórica e exemplos de sistemas implementados, como o GRANT, um sistema que utilizava o SA para obter correlacionamentos entre agências e tópicos de pesquisa, sendo este o primeiro sistema a apresentar o problema da configuração dos parâmetros do SA. Com as experiências do *survey* de Crestani e o trabalho de (Androutsopoulos, Tsatsaronis e Vazirgiannis 2007), que implementa uma *SAN* a partir de um *thesauro* (análogo a um Vocabulário Controlado) foi possível entender como funciona o SA e como definir uma *SAN* para este algoritmo. Nosso trabalho utiliza as definições de RS os cálculos do SA e adapta estas definições para o contexto do CIBELE. A proposição de um mecanismo de buscas híbrido é apresentada por (Aragao M. P.; Rocha e Schwabe 2004), de maneira a inspirar a busca composta no DSpace.

Outros trabalhos relacionados ao SA, porém com focos distintos do nosso são: o trabalho de (D'Agostini C.S.; Fileto e Gauthier 2008), que utiliza o *Spreading Activation* com o objetivo de capturar contextos entre usuários, o de (Nilas N.; Nilas e Masakul 2007), que usa o SA para implementar recomendações em um sistema de *e-commerce* e o de (Aswath D.; Ahmed e Davulcu 2005), usa o SA para identificar termos que afetam positivamente e termos que afetam negativamente um determinado documento, de maneira a treinar um algoritmo de inteligência artificial.

2 *WEB SEMÂNTICA*

A melhoria da qualidade dos sistemas de processamento de buscas exige uma organização e descrição apropriadas dos dados a serem recuperados. Para atender a esta exigência, novas formas da *Web* foram desenvolvidas, com a missão de aumentar o seu potencial semântico. A *Web* semântica tem o propósito de definir metadados de maneira a criar instâncias de um modelo em vez de apenas simples páginas *Web* (Breitman, Casanova e Truszkowski 2007) (Davies, Studer e Warren 2006) (Aragao M. P.; Rocha e Schwabe 2004). Essa mudança conceitual é a chave para o sucesso das buscas semânticas, pois a recuperação de informação semântica somente é efetiva se atuar sobre uma estrutura de conhecimento bem definida para suportar a recuperação de instâncias de recursos de informação. Este capítulo revisa conceitos, padrões e técnicas relacionados à *Web* Semântica utilizados no desenvolvimento deste trabalho.

2.1 *WEB* SOCIAL E *WEB* SEMÂNTICA

A *Web* 2 ou *Web* Social permitiu a participação dos usuários, de tal maneira que estes usuários são tanto produtores quanto consumidores das informações. Uma definição da *Web* Social foi proposta no trabalho de (O'Reilly 2005) como segue:

> "*Web* 2.0 é a mudança para uma internet como plataforma, e um entendimento das regras para obter sucesso nesta nova plataforma. Entre outras, a regra mais importante é desenvolver aplicativos que aproveitem os efeitos de rede para se tornarem melhores quanto mais são usados pelas pessoas, aproveitando a inteligência coletiva"

Este aproveitamento da inteligência coletiva das pessoas é obtido através de aplicações *Web* que possuam interfaces capazes de prover enriquecimento de conteúdo através de comentários, avaliação, ou personalização.

A **Figura 1** adaptada de [1] apresenta uma quadro comparativo entre a *Web* convencional ou *Web* 1.0, a *Web* Social ou *Web* 2.0 (O'Reilly 2005) e a *Web* Semântica

[1] http://adrianazardini.blogspot.com/2008/11/web-semntica.html

(Berners-Lee T.; Hendler e Lassila 2001). A informação na *Web* 1.0 era simplesmente definida por um Produtor e lida por um Consumidor. Na *Web* 2.0 todos os usuários são tanto produtores quanto consumidores, gerando a "*Web* Social". Contudo a **interpretação** da informação na *Web* 2.0 ainda continuava a cargo dos seres humanos. A *Web* semântica propõe padrões que permitam anotações semânticas nos documentos, via ontologias por exemplo, tornando estes documentos *inteligíveis* às máquinas e permitindo o trabalho colaborativo com os seres humanos.

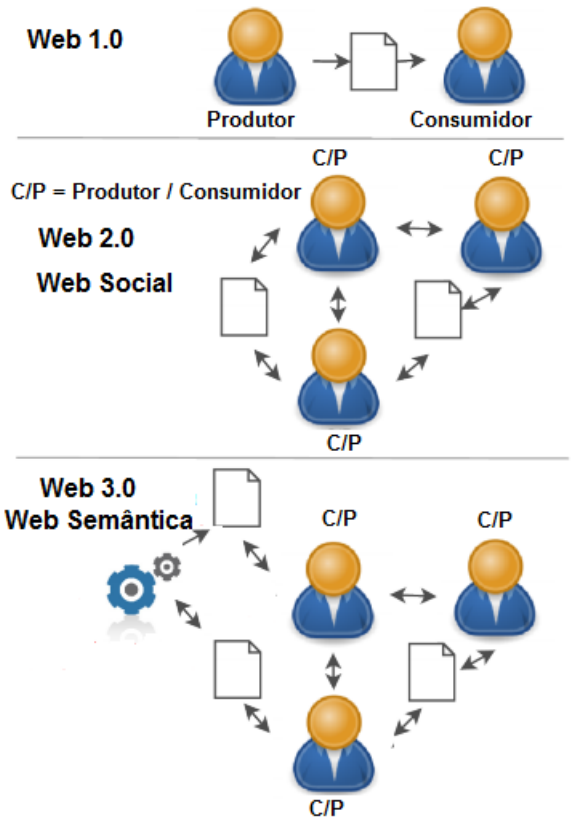


Figura 1: Comparativo entre *Web* 1.0, *Web* 2.0 e *Web* Semântica.

O aumento expressivo da quantidade de dados na *Web* Social tornou o processo de recuperação por palavras-chave ineficaz em algumas situações, devido a limitações de precisão e cobertura. Profissionais da área de organização do conhecimento têm trabalhado em uma extensão da *Web* 2.0, que explora as correlações semânticas entre conteúdos da *Web*. O objetivo é tornar os recursos disponíveis via *Web* mais inteligíveis às máquinas, fazendo com que os algoritmos por elas executados (principalmente as buscas) contribuam mais para os humanos. Surge assim a *Web* semântica, proposta por Tim Berners-Lee, James Hendler e Ora Lassila com a seguinte definição:

> "The Semantic Web is an extension of the current Web in which information is given well-defined meaning, better enabling computers and people to work

in cooperation."(Berners-Lee T.; Hendler e Lassila 2001, p. 34)

A partir dessa definição passou-se a desenvolver aplicativos que, além de processamento meramente sintático, como ocorre na *Web* atual, tratam a semântica. A *Web* semântica (Berners-Lee T.; Hendler e Lassila 2001) apresenta uma nova forma de interação onde as máquinas podem interpretar a informação produzida com a finalidade de atender à demanda de outros humanos que utilizam mecanismos de buscas, por exemplo. A arquitetura da *Web* Semântica requer uma forma de representação dos recursos (documentos, pessoas, textos, objetos, mídias, etc.) e das relações que definem a semântica de cada recurso. As ontologias cumprem a tarefa de representar o conhecimento e as anotações semânticas definem como os recursos se relacionam com tal conhecimento. Para compreender a arquitetura da *Web* Semântica é necessário compreender todos os conceitos envolvidos em seu desenvolvimento. Com esta finalidade (Leon e Perojo 2005) apresenta um mapa conceitual com os relacionamentos dos conceitos que formam a *Web* semântica, sendo este próprio mapa uma forma de apresentar uma ontologia com as relações entre as instâncias do domínio. A **Figura 2** mostra este mapa conceitual, apresentando todos os conceitos envolvidos desde a elaboração das *Web* atual (2.0) até linguagens e ferramentas para definição dos metadados da *Web* Semântica. Os nodos que apresentam agentes inteligentes e padrões de metadados não são abordados neste trabalho.

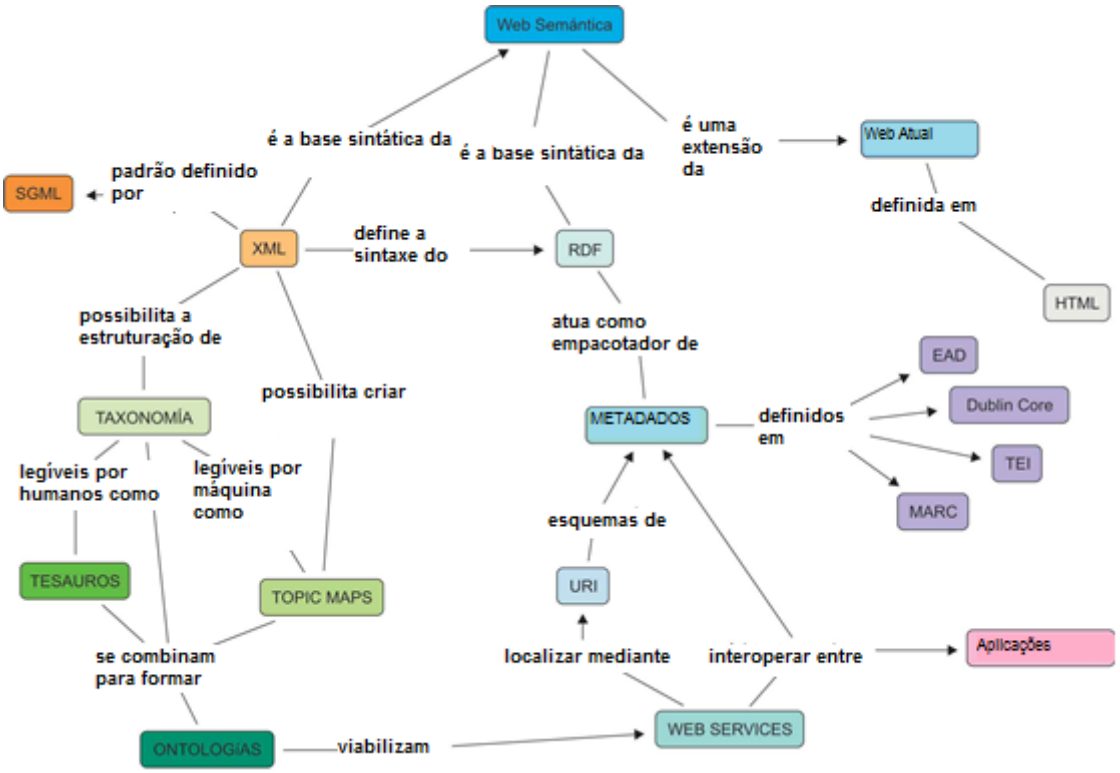


Figura 2: Mapa Conceitual da *Web* semântica. Adaptado de (Leon e Perojo 2005).

A *Web* Semântica que tem como principal objetivo estender a interação da *Web* 2.0 para as máquinas, fazendo com que dados formem o que James Hendler chama de "ilhas de conhecimento", ou seja, nichos de conhecimento específicos de algum domínio que através da interoperabilidade das ontologias poderão interagir de maneira inteligível para humanos e máquinas. A *Web* Semântica permite uma melhoria no processo de IR (*Information Retrieval*) (Baeza-Yates e Ribeiro-Neto 1999) desde que as informações estejam bem definidas.

A representação e o intercâmbio de ontologias e anotações semânticas entre aplicativos da Web semântica requer a adoção de certos padrões. O W3C descreve a *Web* Semântica[2] como uma *Web* de dados, com os significados a eles associados e duas preocupações:

- Padrões para representação de dados amplamente aceitos que promovam o intercâmbio, integração e combinação de dados de diversas fontes (XML);
- Linguagens de representação de conhecimento para registrar a semântica de entidades do mundo real e como os dados se relacionam com essas entidades (RDF e OWL).

Esses padrões de representação de dados e linguagens de representação de conhecimento permitem que uma pessoa ou máquina interaja com várias bases de dados conectadas através de significados comuns.

2.2 ONTOLOGIAS

De maneira informal definimos uma ontologia como uma forma de representação de conhecimento de um determinado domínio, descrevendo seus conceitos, instâncias e relações semânticas entre os primeiros. O trabalho de (Guarino 1998.) apresenta uma ontologia como "uma conceitualização compartilhada de um universo de discurso". O mesmo Guarino faz uma distinção entre o conceito de ontologia na filosofia e em ciência da computação (geralmente na área de inteligência artificial). Ele considera uma ontologia como um sistema particular de categorias que representam certa visão do mundo, não importando a linguagem que a descreve. Sob a ótica da ciência de computação podemos considerar uma ontologia como um artefato de engenharia, constituído por um vocabulário específico usado para descrever certa realidade, mais um conjunto de pressupostos explícitos sobre o significado pretendido das palavras do vocabulário.

Este trabalho considera ontologias com os seguintes elementos, cada qual rotulado (nomeado) por um ou mais termos (palavras) de um vocabulário:

[2]http://www.w3.org/2001/sw/

- Indivíduos: são os objetos básicos do domínio. E.g.: uma pessoa que sofreu AVC;
- Conceitos: definem as coleções, conjuntos e tipos dos objetos de domínio. E.g.: doença, gripe, pessoa, garganta;
- Atributos: propriedades características ou parâmetros que os objetos podem ter e compartilhar. E.g.: nome da doença, sintomas, nome da parte do corpo acometida pela doença;
- Relacionamentos: são as formas como os objetos podem se relacionar com outros objetos. E.g.: AVC **acomete** Cérebro.

Um elemento de uma ontologia (classe, indivíduo, atributo ou relacionamento) pode ser referenciado por mais de um termo de um vocabulário, i.e., sinônimos referindo-se ao mesmo elemento. Por outro lado, algumas vezes também acontece de um mesmo termo de um vocabulário ser utilizado para se referir a diferentes elementos, i.e., homônimos referindo-se a elementos distintos. Vocabulários controlados procuram capturar relações de sinonímia, as quais permitem expansão semântica, e evitar homonímias, pois estas levam a ambiguidades.

A **Figura 3** mostra os indivíduos *AVC* e *Cérebro* conectados por um relacionamento *acomete*, caracterizando assim uma relação semântica entre termos na ontologia do domínio da saúde.

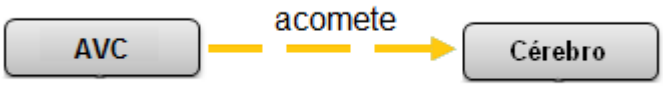


Figura 3: Exemplo de Relação Semântica.

Uma ontologia bem definida deve atender as seguintes propriedades:

- Formal: uma ontologia deve ser formalmente definida, seguindo padrões de linguagens de maneira a não permitir ambiguidades;
- Explícita: todo o conhecimento expresso em uma ontologia deve ser explicitado e não subentendido ou passível de diferentes interpretações;
- Compartilhada: o conhecimento explicitado em uma ontologia é válido para todo o seu domínio.

2.3 ANOTAÇÕES SEMÂNTICAS

Uma anotação semântica descreve recursos de informação pela associação desses a termos referenciando conceitos ou instância de uma ontologia (Han e Reeve 2005) (Ciravegna et al. 2005).

A **Figura 4** a seguir ilustra anotações semânticas dos recursos *Recurso1* e *Recurso2* usando o termo *Acidente Vascular Cerebral*, através das linhas pontilhadas. As linhas tracejadas representam relações semânticas entre os termos *AVC*, *Ictus Cerebral* e *Apoplexia* com o termo *Acidente Cerebral Vascular*, do tipo *sinônimo*. A relação não rotulada entre *Acidente Cerebral Vascular* e *Transtornos Cerebrovasculares*, denotada pela linha contínua representa um relacionamento de especialização entre esses dois conceitos.

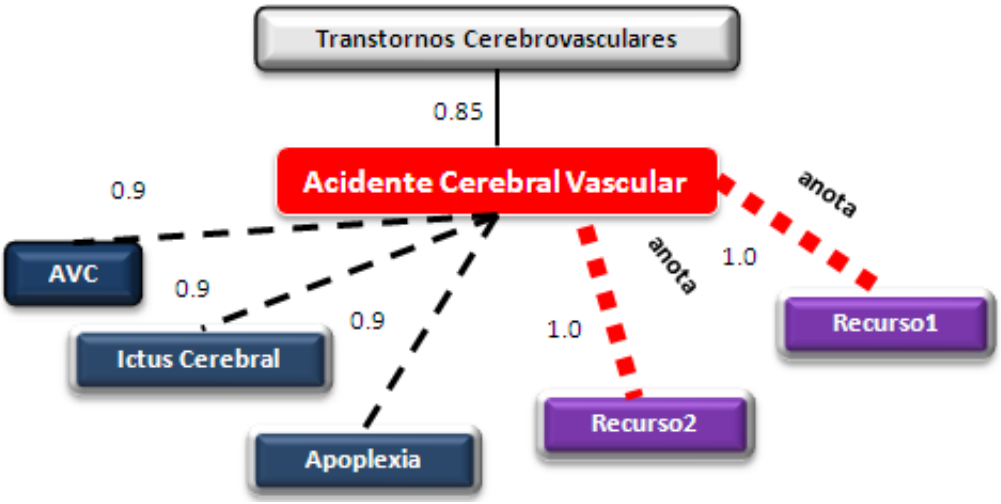


Figura 4: Exemplo de anotação semântica de Recursos de Informação.

A linguagem RDF (*Resource Description Framework*) [3] é utilizada para representar anotações semânticas de recursos de informação com termos referentes a conceitos ou instâncias de uma ontologia, pois é padronizada e possui interoperabilidade com outros sistemas baseados em ontologias. Esta característica permite o reuso de algoritmos de recuperação de informações que utilize as anotações semânticas.

Para ilustrar a sintaxe do RDF utilizamos o exemplo "*AVC acomete Cérebro*" apresentado anteriormente. Os objetos *AVC* e *Cérebro* são representados pela *tag* <**rdf:Description**>, o tipo dos objetos TERMO é representado pela *tag* <**nsRelDeCS:tipoNodo**> e a relação **acomete** é representada pela *tag* <**nsRelDeCS:acomete**>. A definição destas *tags* assemelham-se a definições de *tags* em HTML. O código fica desta maneira:

[3] http://www.w3.org/standards/techs/rdf

```
<?xml version="1.0" encoding="utf-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:nsRelDeCS="http://decs.org/relationship/">
      <rdf:Description rdf:about="http://decs/AVC">
    <nsRelDeCS:acomete rdf:resource="http://decs/Cerebro"/>
    <nsRelDeCS:tipoNodo>TERMO</nsRelDeCS:tipoNodo>
    <nsRelDeCS:decs_id
          rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
          03.383.773.170
             </nsRelDeCS:decs_id>
      </rdf:Description>
</rdf:RDF>
```

Figura 5: Exemplo de Ontologia em RDF.

Aplicando o código RDF acima na ferramenta *RDF Gravity* [4] obtém-se a visualização da **Figura 6** abaixo. Observa-se que o recurso *AVC* possui os atributos **decsID** = (03.383.773.170) e **tipoNodo** = TERMO e tem um relacionamento com o recurso *Cerebro* do tipo *acomete*. Este simples exemplo apresenta sucintamente o potencial semântico explorado pela definição de uma ontologia simples em RDF, pois este modelo permite o funcionamento de mecanismos de buscas semânticas.

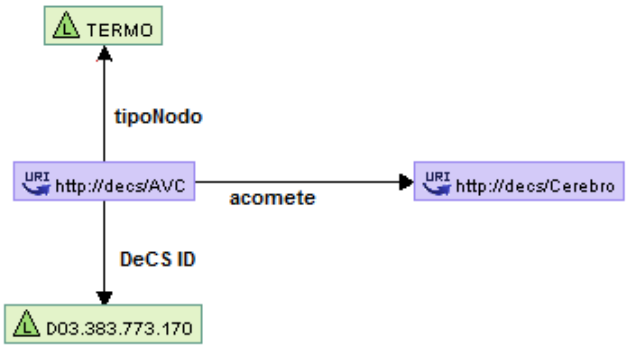


Figura 6: Exemplo de Ontologia em RDF visualizado na ferramenta *RDF Gravity*.

As anotações podem ser geradas de duas formas: (i) manualmente, i.e., com o usuário especialista definindo quais relacionamentos existem entre os recursos ou (ii) semi-automaticamente, forma no qual o programador define os tipos de anotações que existem na ontologia e determina algoritmos que inferem e geram novas anotações a partir deste conhecimento, percorrendo toda a ontologia.

Como visto neste capítulo, as buscas semânticas são amparadas por ontologias e anotações semânticas, sendo que a união destes dois elementos forma uma Rede Semântica, descrita no capítulo a seguir.

[4] http://semweb.salzburgresearch.at/apps/rdf-gravity/

3 BUSCAS SEMÂNTICAS COM SPREADING ACTIVATION

A recuperação de informação (IR) levando em conta a semântica requer técnicas mais avançadas que o simples processamento sintático de *keywords*. Com este objetivo o trabalho de (Mangold 2007) apresenta abordagens para buscas semânticas em documentos, destacando técnicas de recuperação baseadas em grafos (*graph-based query modification*), onde a expansão da consulta é obtida através da execução de buscas sobre uma estrutura de grafos. Esta expansão permite obter um conjunto maior de *keywords* semanticamente relacionadas com o conjunto de *keywords* informado pelo usuário, determinando assim a noção de associatividade entre termos. Este grafo onde a busca semântica atua é conhecido com Rede Semântica.

Com a definição de Rede Semântica o processo de buscas semânticas implementado com o algoritmo de *Spreading Activation* ("Ativação por Espalhamento").

3.1 REDES SEMÂNTICAS

Redes Semânticas (RS) foram introduzidas no trabalho de (Collins e Quillian 1968) e desempenham um papel significativo nas pesquisas sobre representação do conhecimento (Mangold 2007) (Crestani 1997) (Androutsopoulos, Tsatsaronis e Vazirgiannis 2007). O mesmo Quillian afirma que redes semânticas "expressam conhecimento em termos de conceitos, suas propriedades, e as relações de hierarquia sub-superclasse entre os conceitos.". Cada conceito é representado por um nodo e as relações hierárquicas entre conceitos são representadas por conexões estabelecidas entre nodos (Schiel 1989). Podemos concluir que uma estrutura de rede semântica forma um grafo que, dependendo da natureza das relações semânticas neles presentes, pode ser uma árvore ou grafo acíclico direcionado (*Directed Acyclic Graph - DAG*). Por exemplo, se forem utilizadas somente relações dos tipos IS-A (holonímia), PART-OF (meronímia), TYPE-OF (instanciação) e anotações semânticas em uma rede semântica, considerando todas essas relações direcionadas do elemento mais genérico para o mais específico,

esta rede semântica é um *DAG*.

A principal característica de uma representação em rede é que ela apresenta um objeto em termos de suas relações com outros objetos. A vantagem de uma rede reside no seu poder de expressividade, considerando que no processo de IR o significado de um objeto só pode ser completamente capturado se as relações semânticas entre objetos forem bem explicitadas. O trabalho (Crestani e Rijsbergen 1993) diz que a complexidade do problema de recuperação de informação (IR) reside na as relações e não nos dados.

Em uma rede associativa com hierarquia de classes e objetos os nodos de nível mais baixo da árvore denotam classes ou categorias de indivíduos mais especializados, enquanto que nodos de níveis mais elevados denotam classes ou categorias de indivíduos mais abstratos. Propriedades também são representadas por nodos. Uma propriedade que se aplica a um conceito é representada através da ligação do nodo propriedade ao nodo conceito através de um link rotulado adequadamente. Normalmente, uma propriedade está ligada ao conceito mais elevado na hierarquia conceitual onde ela se aplica. Se uma propriedade é anexada a um nodo, presume-se que se aplica a todos os nodos que são descendentes desse nodo (Crestani 1997), lembrando o conceito de herança utilizado em Orientação a Objetos, por exemplo.

Um exemplo de RS completa é mostrado na **Figura 7**, onde temos uma representação de uma hierarquia de termos relacionados a saúde (em cinza), termos que representam sinônimos (em azul escuro) e Recursos de Informação (em roxo). As relações entre os termos são: anônima (em preto contínuo), sinônimo (tracejada), anota (pontilhada) e específica de domínio (tracejado com seta).

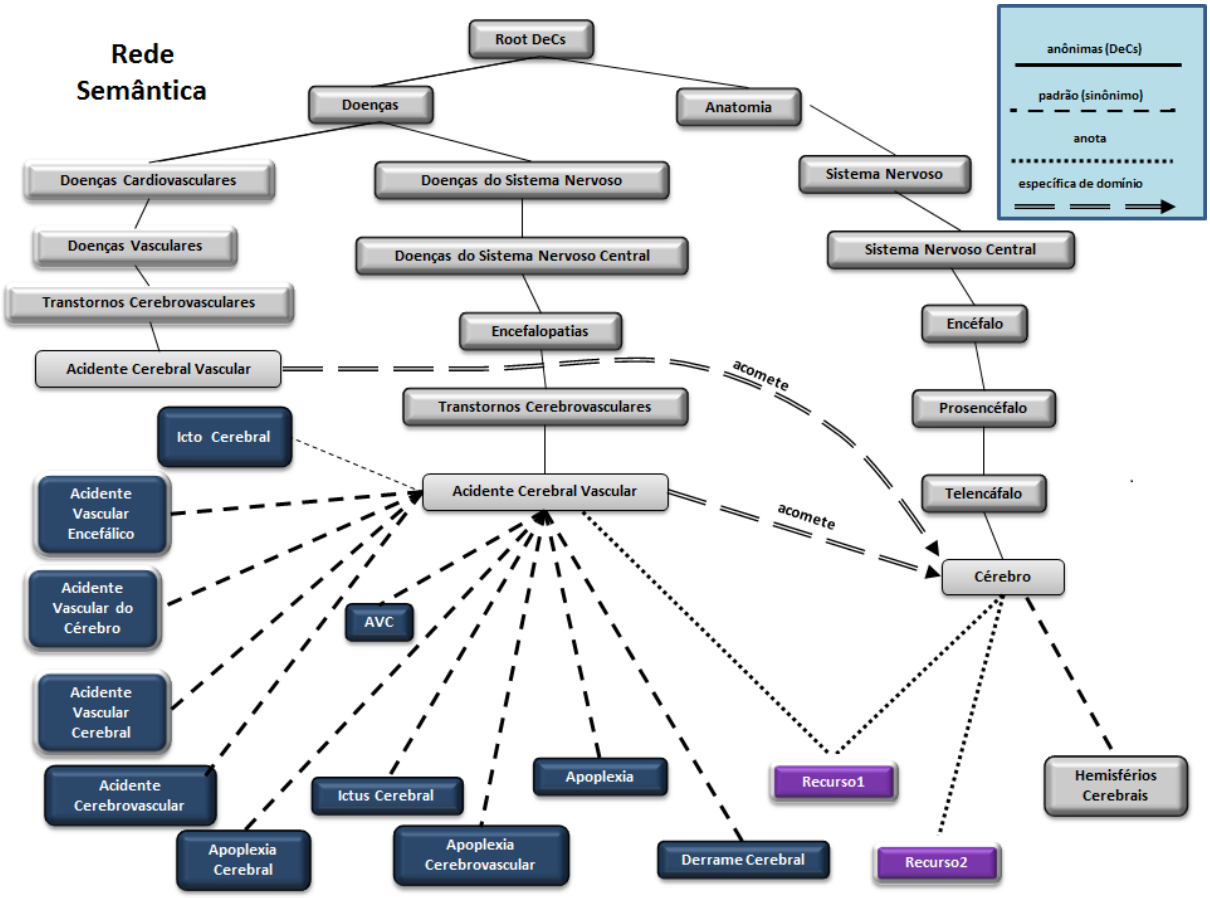


Figura 7: Exemplo de rede semântica.

As arestas de uma rede associativa podem ter pesos, os quais adicionam a noção de força (importância) das associações. Pesos também podem ser adicionados aos vértices, sendo que essas parametrizações permitem uma avaliação dos resultados obtidos na recuperação. O algoritmo de *Spreading Activation* atua sobre uma RS da seguinte maneira: dado um conjunto de nodos K que constituem as palavras-chave iniciais, o SA expande K para um conjunto $K' \supseteq K$, percorrendo a RS iterativamente nodo a nodo e atualizando os valores numéricos de relevância semântica dos nodos visitados. Esta atualização do valor dos nodos leva em conta os seguintes atributos: peso das arestas entre o nodo disparado e seus adjacentes, valor de cada nodo adjacente e outros fatores que serão apresentados na seção 3.3.1. O peso das arestas é definido de acordo com o tipo da relação semântica que esta aresta representa. E.g.: a aresta *sinonimo* que conecta os nodos *AVC* e *Acidente Vascular Cerebral* possui peso associado 1.0, pois a relação de sinonímia tem valor máximo atribuído. Já a relação *Transtornos Cerebrovasculares generaliza Acidente Vascular Cerebral* tem valor 0.85, pois estes termos já possuem um distanciamento semântico.

3.2 BUSCAS SEMÂNTICAS

A maioria dos sistemas de buscas tradicionais apoiam-se na ocorrência de palavras nos documentos (Mangold 2007). Esta característica torna estes sistemas genéricos e simples, entretanto impede a expansão semântica das consultas, pois os relacionamentos entre as palavras-chave e termos semanticamente associados não são considerados. Por outro lado, buscas semânticas exploram o conhecimento do domínio, formalizado em uma ontologia. O trabalho de (Guha, McColl e Miller 2003) concorda que a tecnologia de recuperação de informação tradicional é baseada quase que puramente na ocorrência de palavras em documentos.

Dessa forma percebe-se que os sistemas tradicionais apenas utilizam processamento léxico e sintático, sem considerar ou mesmo mencionar a semântica das buscas. Com o desenvolvimento da nova *Web*, espera-se que a disponibilidade de dados anotados semanticamente aumente e novos sistemas de busca, que levem em conta a semântica, sejam desenvolvidos. Sistemas de busca semântica objetivam que seu mecanismo de busca não faça apenas uma simples pesquisa por palavras-chave, mas que reconheçam o significado das palavras pesquisadas, e em que contexto elas residem. Com a informação da semântica associada aos documentos disponíveis pode-se aumentar a cobertura e a relevância dos resultados recuperados.

O trabalho de (Guha, McColl e Miller 2003) descreve dois objetivos nas buscas semânticas: (i) estender os resultados dos sistemas de busca tradicionais usando semântica (e.g., termos semanticamente relacionados aos usados na busca, descritos em uma ontologia) e (ii) detectar e estimar prováveis denotações que aumentem a precisão da busca (e.g., busca por possíveis significados de um termo em uma ontologia e utilização de informação de contexto dos usuários para desambiguação quando necessário).

Uma das formas de se realizar buscas semânticas se dá através da utilização das propriedades da ontologia que representa o domínio pesquisado. Esse processo é conhecido como *recuperação associativa de informação*, definida por (Crestani 1997):

> "The idea behind this form of information retrieval is that it is possible to retrieve relevant information by retrieving information that is **'associated'** with some information the user already retrieved and that is know it to be relevant. The associations between information can either be static and already existing at the time of the query session, or dynamic and determined at run time. In the first case, associations among information items (document or parts of documents, extracted terms, index terms, concepts, etc.) are created before the query session, and they make use of semantic relationships between these items, such as for example thesaurus-like relationships among index terms, bibliographic citations among documents, or statistical similarity among documents or terms. In the last case, instead, the system determines associations between information items through interaction with the user, for example by

> retrieving documents that are similar to documents the user points out to be relevant."

Os trabalhos de (Mangold 2007) e (Chang E; Hai Dong; Hussain 2008) afirmam que o processo de buscas semânticas pode utilizar algumas propriedades das ontologias. são elas:

- Propriedades anônimas: que indicam meramente que os conceitos pertencem a uma mesma ontologia. Segundo (Mangold 2007): "in the case of anonymous properties, the system disregards the name and the semantics of the property.", ou seja, a inter-relação entre dois conceitos indica apenas que eles compartilham o mesmo contexto.
- Propriedades padrão: são as propriedades que definem relações "tradicionais"entre conceitos: sinonímia, meronímia(PART-OF), hipernímia (IS-A), instância-de. Essas propriedades potencializam a recuperação de informação, mas introduzem dependências na estrutura da ontologia.
- Propriedades de domínio específico: são propriedades inerentes ao domínio da ontologia definida, sendo muito importantes no processamento das buscas. Estas propriedades podem apresentar nuances dependentes do domínio em questão. Ex.: AVC **acomete** Cérebro.

Essas propriedades possibilitam o correlacionamento dos termos da ontologia, permitindo assim o processo de buscas semânticas. Entretanto é necessária uma solução algorítmica para atuar sobre o modelo de representação de conhecimento. Este trabalho optou pelo *Spreading Activation*, apresentado na seção a seguir.

3.3 *SPREADING ACTIVATION*

O processo de buscas semânticas foi realizado com a técnica de *Spreading Activation* ("Espalhamento por Ativação"), o qual atua sobre uma RS propagando ondas de ativação iterativamente por meio das relações estabelecidas na RS, caracterizando assim um busca associativa sobre o conhecimento modelado pela RS. A justificativa da escolha do *Spreading Activation* está fundamentada no fato que este algoritmo representa a organização da memória humana, de maneira que termos (ou pensamentos) mais próximos estejam mais conectados na rede semântica da mesma forma como neurônios que desempenham atividades similares estão fisicamente próximos.

O comportamento do SA pode ser entendido através da metáfora com o fenômeno de uma pedra lançada em um lago. Formam-se "ondas de ativação" que têm sua força diminuída à

medida em que se afastam do ponto de colisão da pedra com o lago. Sob esta ótica, podemos considerar que o lago é a rede associativa, a pedra é o conjunto K de palavras-chave e as ondas de ativação são os passos de execução do SA. Ao final da execução do SA, o conjunto de palavras-chave K inicial é expandido para um conjunto K' com mais palavras-chave, aumentando assim o potencial da consulta inicial. A **Figura 8** apresenta a intuição que permeia o *Spreading Activation*, onde o conjunto K tem a palavra-chave *AVC* que atua como semente do SA. O SA,quando executado, expande K para um conjunto K' com termos semanticamente relacionados na RS. Este conjunto K' contém os termos *AVC, Apoplexia, Transtornos Cerebrovasculares, Acidente Cerebral Vascular, Derrame Cerebral*.

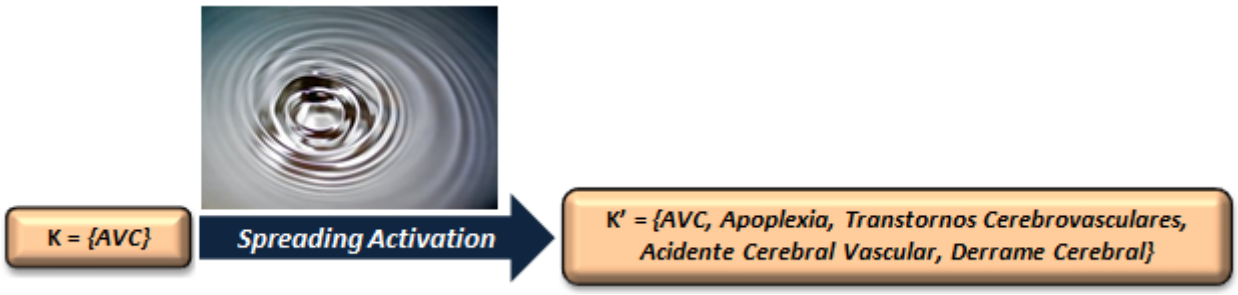


Figura 8: Exemplo de comportamento do SA.

3.3.1 HISTÓRICO

O trabalho de (Crestani 1997) nos dá um panorama de toda a evolução das técnicas que embasam o *Spreading Activation* (SA). A seguir temos um resumo deste histórico:

A técnica de SA não foi a pioneira do processo de recuperação associativa de informação, existindo relatos de processos desenvolvidos nos anos 60 como o *associative linear retrieval model*, um modelo de expansão de consultas baseado em estatísticas de associações entre termo-termo, termo-documento e documento-documento. A idéia desta técnica era construir uma matriz de similaridade entre termos e documentos.

O SA foi desenvolvido nos anos 80, baseado em experimentos psicológicos de (Rumelhart D.E; Norman 1983) que foram aplicados inicialmente à Ciência da Computação através da Inteligência Artificial, com o objetivo de prover um framework de processamento para redes semânticas. Este modelo de recuperação associativa é baseado no mecanismo de operação de memória humana, onde termos mais relacionados estão conectados por neurônios vizinhos. Seu uso tem sido elogiado e criticado, mas atualmente é adotado em diversas áreas, tais como: Ciência Cognitiva, Bases de Dados, Inteligência Artificial, Psicologia, Biologia, e ultimamente da recuperação de informação (RI). O modelo básico do SA, no entanto, tem sido

objeto de diversas melhorias, a fim para torná-lo mais adequado às aplicações de RI. A próxima seção apresenta este modelo puro sob esta ótica da RI.

3.3.2 MODELO ORIGINAL

Para compreender o *Spreading Activation* - SA primeiramente deve-se compreender os conceitos envolvidos neste algoritmos, que são apresentados a seguir. Dada uma rede semântica $RS(NS, AS)$ (RS de tamanho finito e limitado pelo ambiente onde o algoritmo executa), onde NS é o conjunto de nodos e AS o conjunto de arestas de RS, o algoritmo de SA atua iterativamente sobre esta RS de acordo com as seguintes definições:

- nodo (i): é um elemento de NS, isto é, um nodo da RS, podendo ser tanto um termo do vocabulário controlado (VC) quanto um Recurso de Informação (RI). Um termo pode se referir a um conceito ou instância de conceito, os quais podem ser referenciados por um termo canônico (nome principal) e outros alternativos (sinônimos)[1]. Um Recurso de Informação pode ser qualquer artefato de informação (documento, imagens, som, vídeo, Objeto de Aprendizagem, etc.). E.g: o termo AVC ou o RI $O.A.1$.

- sementes (K): o conjunto palavras-chave da consulta constituído de termos $k \in NS$. Estes termos $k \in NS$ constituem o conjunto K de sementes do SA, ou seja, o conjunto de nodos que iniciarão a execução do algoritmo.

- peso da relação (W_{ij}): uma aresta ij, w_{ij} , $i, j \in AS$ determina uma associação semântica entre os nodos i e j. Esta relação possui um peso $W_{ij} \in [0,1]$, que pode ser determinado pelo seu tipo. E.g.: a relação sinônimo entre $i =$AVC e $j =$Acidente Cerebral Vascular tem peso $W_{ij} = 1$.

- pulso (p): um ciclo de disparo do *Spreading Activation*, onde os nodos i selecionados para ativação tem seus valores A$i(p-1)$ atualizados para A$i(p)$, considerando os pesos dos nodos adjacentes. A idéia de um pulso é caminhar um passo na Rede Semântica RS, através das associações semânticas entre i e seus adjacentes.

- *threshold*: é o limiar de ativação mínimo aceito para a continuidade dos pulsos em um determinado nodo. Este parâmetro também situa-se entre [0,1]. E.g.: em um determinado pulso p o nodo i possui valor de ativação A$i(p)$ = 0,79, se *threshold* = 0,8, então o nodo i não será marcado para ativação e não terá seu valor de ativação atualizado.

[1] Vocabulários bem definidos e estruturados, como é o caso do DeCS, evitam ambiguidades, isto é, o mesmo termo referindo-se a conceitos ou instâncias distintos.

- fator de decaída (D) - *decay factor*: este fator faz com que o valor de ativação Ai de um nodo diminua à medida que se afasta da semente, caracterizando assim uma diminuição de força, semelhante à metáfora da pedra no lago.

- valor de ativação (A_i): o valor de ativação refere-se ao valor atribuído a um nodo i em uma passagem do SA por este nodo, situado no intervalo [0,1]. Se $i \in$ K, o seu valor de ativação inicial A_i = 1, caso contrário A_i = 0. O A_i é calculado em função de seu valor no passo anterior, valores de ativação de nodos adjacentes calculados no passo anterior e peso das arestas levando esses nodos a i, de acordo com **Definição 2**.

- relevância semântica: este valor é o valor de ativação final A_i de um nodo i, pois o SA pode passar várias vezes por i e atualizar o valor de ativação. Este valor de relevância semântica é retornado em uma listagem junto a cada um dos nodos que tenham alcançado o valor de *threshold* mínimo aceito.

O SA atua sobre uma rede associativa, calculando os valores de relevância no intervalo [0,1] para os nodos alcançados a cada iteração do algoritmo. O algoritmo inicia com um conjunto K de nós e a cada iteração determina um novo conjunto de nodos a ser ativado. Cada iteração é composta por um ou mais pulsos e a verificação de uma condição de término.

A **Figura 9** apresenta um trecho da rede semântica visto anteriormente com o conjunto de sementes $K = \{AVC\}$. O $Pulso(p) = 0$ indica que o algoritmo ainda não iniciou, o SA dispara o primeiro $Pulso(p) = 1$ e com isso o valor de ativação AVC é atualizado e os seus nodos adjacentes são marcados para o próximo pulso. Na segunda parte desta figura temos a ilustração do nodo *Acidente Cerebral Vascular* marcado para disparo, repetindo assim o processo iterativo.

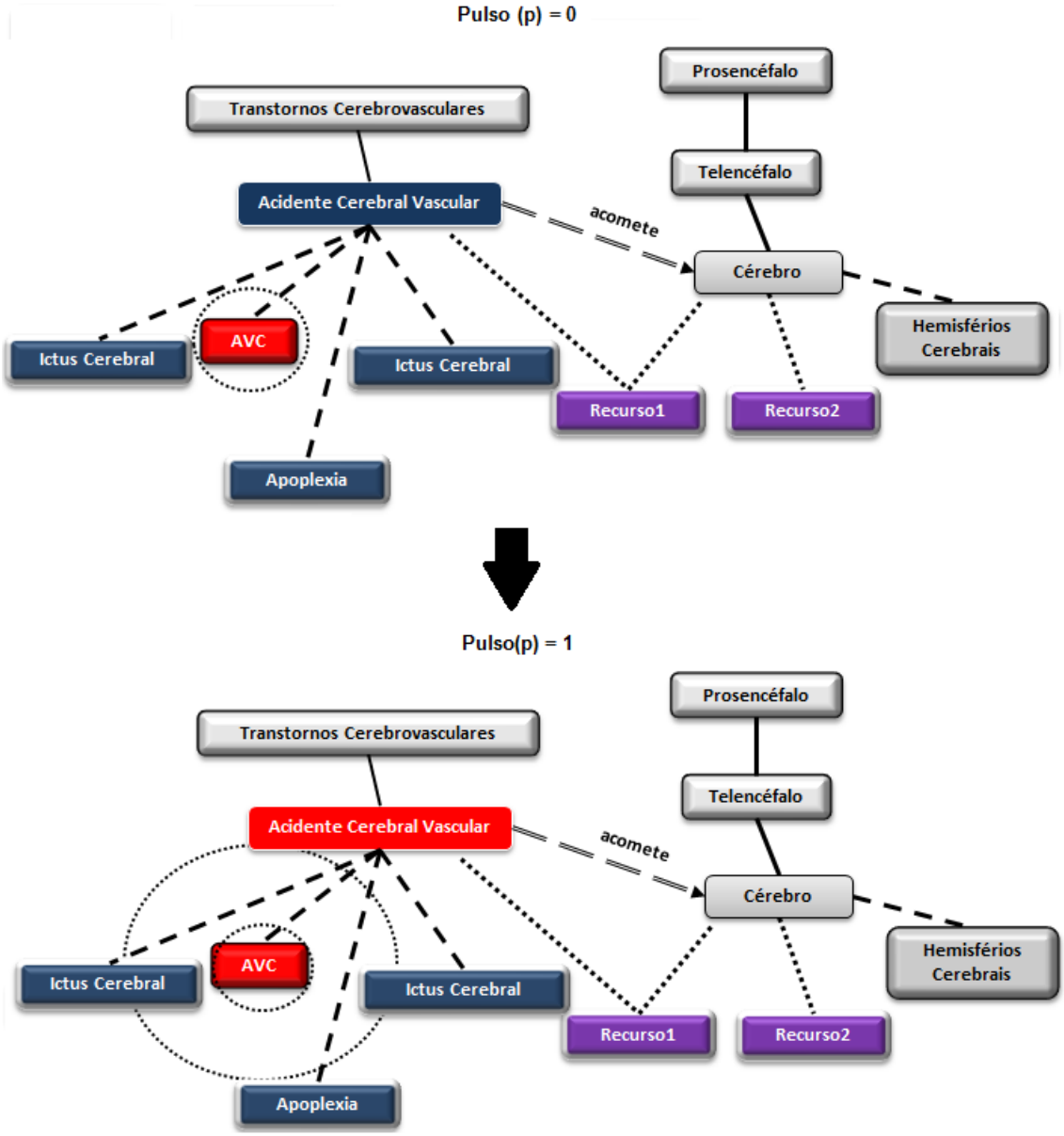


Figura 9: Exemplo de um pulso do *Spreading Activation*.

Em uma busca sintática simples nenhum Recurso de Informação (RI) seria encontrado, pois o termo AVC não anota nenhum RI. Entretanto pode-se notar que AVC é sinônimo de Acidente Cerebral Vascular e este termo anota um objeto *Recurso*1. O SA consegue capturar este conhecimento, expandindo conjunto K obtendo $K' = \{AVC, Acidente\ Vascular\ Cerebral\}$ no primeiro pulso.

Depois desta visão geral apresentamos os detalhes do algoritmo de *Spreading Activation* (SA) usado para processar uma RS. A técnica de processamento é definida por uma sequência de iterações como o descrito esquematicamente na Figura 10. Os pulsos são disparados até o processo ser interrompido pelo usuário ou até que alguma condição de término seja alcançada.

Uma iteração é composta por:

1. um ou mais pulsos;
2. verificação de término.

O que distingue o modelo SA puro de outros modelos mais complexos é a sequência de ações que compõe o pulso. Um pulso é composto de três fases:

1. pré-ajustamento
2. propagação
3. pós-ajustamento

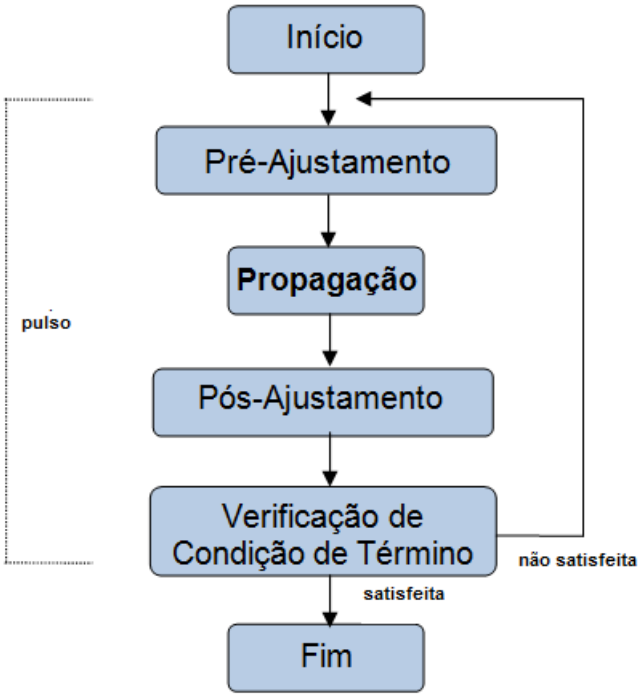


Figura 10: Passos de execução do algoritmo original de *Spreading Activation*.

A seguir temos o detalhamento de cada uma destas fases:

- Pré-ajustamento: esta fase determina o valor dos pesos iniciais dos nós, preparando a rede associativa. Este passo determina o valor inicial dos nodos (1 para os nodos pertencentes ao conjunto de *keywords K* e 0 para os demais).

- Propagação: uma propagação (ou pulso) compreende um passo na execução do algoritmo. Esta fase consiste na passagem de ondas de ativação de um nodo para todos os outros nodos conectados a ele (adjacentes), ou seja: dado um nodo i em que o algoritmo está situado, uma propagação significa expandir em 1 nivel todos os nodos adjacentes i. A cada propagação um conjunto de nodos é *ativado*, ou seja, seus valores de relevância semântica são redefinidos. O cálculo do valor de ativação segue a definição 2 a seguir, onde são ponderados o valor anterior de ativação do nodo disparado i e a somatória dos valores de saída dos nodos adjacentes a i.

Cada nodo i selecionado para disparar em um pulso p tem seu valor de ativação $A_i(p)$ calculado da seguinte maneira (Androutsopoulos, Tsatsaronis e Vazirgiannis 2007): o valor de ativação do pulso anterior $p-1$ é somado ao produto de todos os pesos W_{ji} dos nodos adjacentes j com o valor de ativação de cada um destes nodos adjacentes j e com o fator de decaimento D. A **Definição 2** a seguir apresenta estes cálculos:

Definição 2 (Atualização de um nodo no SA): A atualização do valor de ativação de um nodo i ativado no SA dada por

$$A_i(p) = A_i(p-1) + \sum_{j}^{n} (A_j(p-1)) * (W_{ji}) * (D)$$

onde:

- $A_i(p)$ é o valor de ativação do nodo i no pulso p.
- $A_i(p-1)$ é o valor de ativação do nodo i no pulso $p-1$.
- $A_j(p-1)$ é o valor de ativação do nodo j na iteração $p-1$.
- W_{ij} é o peso associado ao link que conecta o nodo i ao nodo j.
- D é o fator de decaída. O valor de D é um número real $\in [0-1]$
- n é o número de arestas que incidem no nó k

- Pós-condições: considerado um pulso em **n**, a verificação de pós-condições determina quais serão os nodos adjacentes de **n** que serão marcados para ativação no pulso seguinte. Para isso, são utilizados os critérios de número de pulsos e limiar (*threshold*). O número de pulsos limita a quantidade de pulsos realizados e o limiar é um valor mínimo de relevância aceitável de um nodo.

- Verificação de condição de término: o SA é um algoritmo iterativo e requer critérios de parada para que o mesmo não entre em *loop*. Os principais critérios utilizados são apresentados por (Crestani 1997) como segue:

1. Distância: determina a quantidade de níveis existente entre um nodo a ser ativado e um nodo do conjunto de sementes, chamado também de ordem entre 2 nodos. Este critério garante a localidade das soluções, pois a expansão está diretamente ligada à quantidade de pulsos executada;

2. *Fan-out*: este critério faz com que o SA pare a ativação caso encontre um nodo com uma quantidade muito grande de associatividade (muitas arestas), evitando que o algoritmo seja degenerado em uma busca exaustiva. A intuição envolvida no conceito de Fan-out é que um termo com muitos relacionamentos geralmente é demasiadamente genérico e não deverá se expandido.

3. Caminho: levar em conta o caminho percorrido até a ativação de um nodo. O SA pára de executar quando todos os nodos são disparados ou quando chega a um mesmo nodo por caminhos distintos.

4. Ativação: este critério define um limiar (*threshold*) de relevância para ativação de nodo. Caso um nodo k não alcance este limiar o processo de SA não prossegue em k, ou seja, k não é marcado (*firing*) para o próximo pulso.

Observa-se que o SA é um algoritmo iterativo, composto por uma sequência de pulsos e verificação das condições de término. Pulso após pulso, a ativação se espalha pela rede atingindo os nodos que estão longe dos nodos ativados inicialmente. Após um número de pulsos ter sido alcançado ou todos os nodos terem sido disparados, a condição de término é alcançada. Então o processo pára, caso contrário se inicia uma nova série de pulsos. O resultado do processo de SA é um subgrafo K' composto pelos de nodos que foram ativados até a condição de término. A interpretação do nível de ativação de cada nodo depende da aplicação e principalmente das características do objeto que está sendo modelado por aquele nodo. Na **Definição 3** um pseudocódigo do algoritmo de SA é apresentado.

Definição 3 (Algoritmo de *Spreading Activation*)

Dado um conjunto de Sementes K

Seja *Condição de término*:

- quando todos os nodos são disparados;

- quando um número máximo de pulsos é alcançado;
- partindo de sementes $i_k \in K$ distintas, o espalhamento da ativação alcança o mesmo nodo por arestas diferentes

Temos a seguir a rotina de execução principal do SA:

```
1  executaSA();
2  nodosMarcadosParaDisparo = {};
3  float thresholdDisparo, fatorDecaida;
4  for i ∈ K do
       /* Inicia o conjunto de sementes K                          */;
5      atribuirAtivacaoMaxima(i);
6      marcarNodoComoDisparado(i);
       /* Prepara adjacentes para disparo                          */;
7      Adjacentes ←retornaAdjacentesDe(i);
8      marcarNodosParaDisparo(Adjacentes);
9  end
10 while not condição de término and nodosMarcadosParaDisparo ≠ ∅ do
11     for n ∈ nodosMarcadosParaDisparo do          /* Dispara nodo n */
12         calcularAtivação(n);
13         dispararNodo(n);
14         nodosMarcadosParaDisparo.remove(n);
15         marcarComoDisparado(n);
16     end
17 end
```

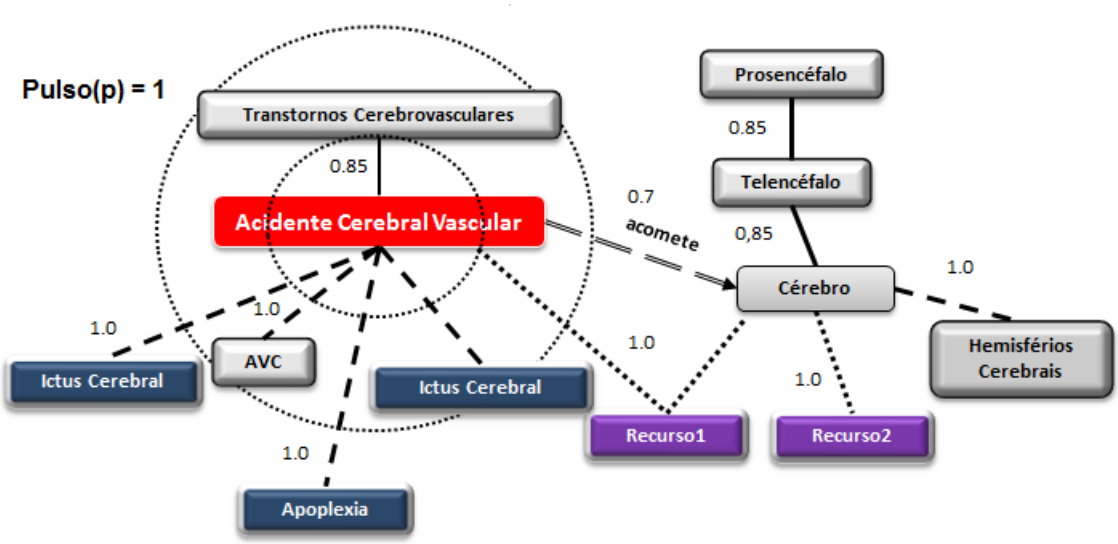
Algorithm 1: Rotina principal do Algoritmo de *Spreading Activation*

Esta rotina principal executa iterativamente até que uma das condições de término seja alcançada. A cada pulso ela *dispara* os nodos marcados, ou seja, os valores destes nodos são atualizados. A seguir temos a descrição das rotinas disparaNodo(), marcarNodosParaDisparo e finalmente calcularAtivação():

```
1  dispararNodo(i);
2  if A_i > thresholdDisparo then
3  |   Adjacentes ← retornaAdjacentesDe(i);
4  |   marcarNodosParaDisparo(Adjacentes);
5  end
6  marcarNodosParaDisparo(Adjacentes);
7  for n ∈ Adjacentes do
8  |   if not disparado(n) then
9  |   |   marcarParaDisparo(n);
10 |   |   nodosMarcadosParaDisparo.add(n);
11 |   end
12 end
13 calcularAtivacao(n);
   /* Ver Definição 2                                                    */;
14 ativacao(i) ← ativacao(i) + somaPesosLinksAdjacentes(i) * fatorDecaida;
```

Algorithm 2: Rotina de atualização de um nodo *i* do Algoritmo de *Spreading Activation*

A **Figura 11** a seguir apresento o cálculo da atualização do valor de ativação $A_i(p)$ do nodo *i*, de acordo com a **Definição 2**. O problema encontra-se na RS do SA puro, pois esta rede não possui especializações nos tipos de arestas, de tal forma que todas as arestas apresentem valores de pesos semelhantes, gerando assim uma motivação para a adaptação deste algoritmo



$$i = AcidenteVascularCerebral$$
$$j = \{Apoplexia, Recurso1, Cérebro, TranstornosCerebrovasculares\}$$
$$A_i(1) = A_i(0) + \sum_{j}^{n} (O_j(p)) * (W_{ji}) * (D)$$
$$A_i(1) = 0 + ((1 * 1 * 0.85) + (0 * \ 1 * 0.85) + (0 * \ 1 * 0.85) + \ (0 * \ 1 * 0.85)$$
$$A_i(1) = 0.85$$

Figura 11: Exemplo de execução do *Spreading Activation*

Com estas definições temos o conhecimento necessário da técnica básica do SA. O próximo capítulo apresenta estas adaptações, descreve a arquitetura do sistema em que ele foi inserido (o CIBELE) e os detalhes de implementação.

4 PROPOSTA

Conhecendo o problema de recuperação associativa de informação e a técnica de *Spreading Activation*, pode-se formalizar o ambiente de trabalho em que este algoritmo foi inserido. Este capítulo descreve o a arquitetura do projeto CIBELE, apresenta o estudo de caso do projeto UnA-SUS da UFSC, finalmente as adaptações e os detalhes do nosso algoritmo de SA.

4.1 CIBELE

O CIBELE é um sistema que provê módulos adicionais e adaptações a repositórios de Objetos de Aprendizagem, provendo interfaces de catalogação, gerência e recuperação de RIs baseadas em conhecimento. Ele foi concebido com o objetivo de melhorar a interação do usuário com repositórios digitais de Objetos de Aprendizagem modificando a menor quantidade possível de código nativo. Para isto, o CIBELE usa técnicas de Programação Web, afim de desenvolver módulos fracamente acoplados com as ferramentas de repositório, ou seja, os módulos do CIBELE são desenvolvidos fora do repositório e são chamados por *links*.

O CIBELE tem o objetivo de melhorar 4 processos básicos de repositório de dados em um AVEA (Ambiente Virtual de Ensino e Aprendizagem): 1) adaptação inicial do conhecimento, 2) catalogação de Objetos de Aprendizagem, 3) gerência do repositório e 4) recuperação de RIs. Cada um destes processos possui tarefas e usuários distintos, gerando demandas de uso de interfaces distintas. A seguir detalhamos cada um destes processos.

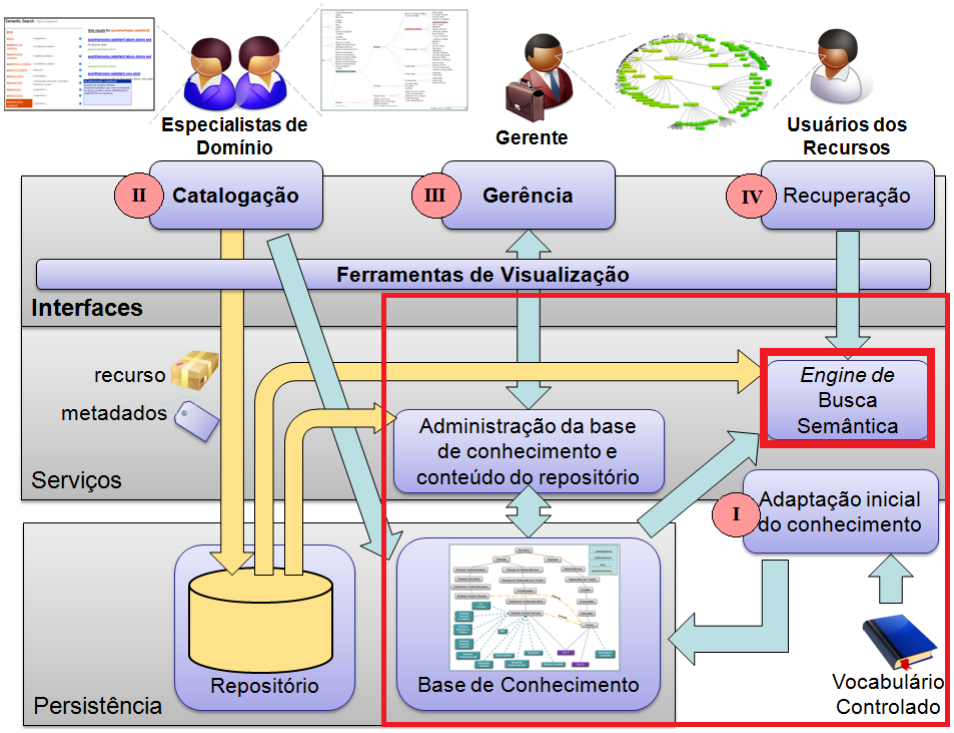


Figura 12: Arquitetura do sistema CIBELE.

I- ADAPTAÇÃO INICIAL DO CONHECIMENTO

Os Objetos de Aprendizagem devem ser corretamente armazenados, para isso eles devem ser **catalogados** com palavras-chave que os identifiquem corretamente. Uma boa prática na catalogação é o uso de dicionários ou **vocabulários controlados**, que evitam problemas de ambiguidade, grafia incorreta e conceitos errôneos. Para evitar estes problemas o CIBELE tem suas interfaces amparadas por vocabulários controlados, que são repositórios com taxonomias e classificações de descritores de determinadas áreas de conhecimento. E.g.: um vocabulário com termos da saúde (nome do descritor, código, sinônimos).

Entretanto nem todos os termos e relações de um vocabulário controlado são úteis a uma aplicação. Recortes temáticos podem ser feitos visando elencar somente porções convenientes de conhecimento. O ideal é que estas porções sejam arranjadas em um conjunto parcialmente ordenado ou **poset** *(partially ordered set)* (Wikipedia 2011). Ele consiste de um conjunto e de relações binárias que indicam que, para determinados pares de elementos do conjunto, um dos elementos precede o outro, ou seja, um poset formaliza o conceito intuitivo de ordenação, seqüenciamento ou arranjo dos elementos de um conjunto.

As relações binárias são chamadas de ordem parcial e refletem o fato de que nem todos os pares de um poset precisam ser relacionados. Formalmente, ordem parcial é uma relação

binária R sobre um conjunto Pq que é reflexiva, anti-simétrica e transitiva, ou seja:

$\forall\ a, b$ e $c \in$ P:

i) a R a (reflexividade);
ii) se a R $b \wedge b$ R $a \rightarrow (a = b)$ (anti-simetria);
iii)se a R $b \wedge b$ R $c \rightarrow (a$ R $c)$ (transitividade).

Um exemplo de conjunto parcialmente ordenado é uma coleção de pessoas ordenada por descendência genealógica. Alguns pares de pessoas carregam a relação ancestral-descendente, enquanto outros pares não. Dada esta contextualização, este trabalho prega a construção de uma estrutura de conhecimento ordenada parcialmente a partir de um vocabulário controlado. Posteriormente ela é enriquecida colaborativamente durante o processo de catalogação.

II- CATALOGAÇÃO

O processo de buscas semânticas depende diretamente de como os Objetos de Aprendizagem são catalogados, pois a recuperação é baseada justamente nos termos que foram definidos como palavras-chave. O CIBELE cataloga os RIs com as palavras obtidas através das interfaces de catalogação amparadas pelos vocabulários controlados. Estas interfaces contornam problemas de desconhecimento de nome ou grafia de um termo e também de sinônimos.

III- GERÊNCIA

Os AVEAs proveem controle de acesso de usuários, via login, possibilitando assim a criação de diferentes papéis, de acordo com o nível de acesso. Um destes papéis é o de gerente de repositório, o profissional que terá pleno acesso à todos os Recursos de Informação, afim de saber a quantidade de RIs catalogados em cada uma das áreas de conhecimento. Um gerente de repositório necessita de ferramentas mais abrangentes que uma busca direta, de maneira a apresentar de forma global a distribuição de Recursos de Informação. Esta visão global auxilia o gerente na definição de esforços futuros na produção de conteúdos, pois ele consegue identificar quais assuntos estão bem ou mal servidos de RIs.

A proposta do CIBELE provê interfaces de visualização hierárquica e hiperbólica do conhecimento, onde pode-se "navegar" pelo repositório afim de facilitar a gerência de todo o conteúdo.

IV- RECUPERAÇÃO

Como destacado na **Figura 12** o foco deste trabalho está na melhoria do processo de recuperação de informação, pois sabe-se que o sistemas de repositórios de Objetos de Aprendizagem possuem por *default* apenas um mecanismo de busca sintática. A busca semântica por Spreading Activation procura melhorar esta deficiência, e o CIBELE acopla uma implementação do SA ao repositório em questão, de forma a prover um módulo de buscas semânticas adicional.

4.2 REDE SEMÂNTICA ADAPTADA

Como visto no capítulo 3, uma RS é a provê a base de relacionamentos entre conhecimentos capaz de realizar buscas semânticas. Contudo, para a execução do SA em um repositório de Recursos de Informação catalogados por palavra-chave é necessária uma formalização desta RS, de maneira a incorporar tanto nodos que representem termos de um vocabulário controlado quanto nodos que representem RIs. A **Definição 4** a seguir apresenta esta Rede Semântica adaptada.

Definição 4 (Rede Semântica): Uma rede semântica é um grafo da forma

$$RS(NS, AS)$$

onde:

- O conjunto de nodos $NS = T \cup RI$ de RS é a união do conjunto de termos do vocabulário T de uma ontologia com um conjunto de recursos de informação RI anotados com termos de T.
- O conjunto de arestas $AS = R \cup A$ é a união do conjunto $R \subset T \times T$ de relações semânticas entre termos de T em uma ontologia com um conjunto de anotações semânticas $AS \subset RI \times T$ de anotações semânticas de recursos de RI com termos de T.

Além de incluir nodos de dois tipos (termo e RI), a **Definição 4** caracteriza o seu conjunto de arestas AS como uma união de dois conjuntos distintos: o conjunto de relações semânticas R e o conjunto de anotações semânticas A. Estes dois conjuntos apresentam diversos tipos que determinam os pesos das arestas de toda a RS. São alguns exemplos de associações semânticas:

- anônima: relação de especialização/generalização entre dois nodos $\in T$, de acordo com sentido em que se percorre a RS. Esta relação deve possuir um peso menor que 1 para caracterizar o distanciamento em relação a semente. E.g: *Acidente Cerebral Vascular* especializa *Transtornos Cerebrovasculares*.

- sinônimo: compreende uma aresta entre dois nodos $\in T$ sinônimos, onde geralmente atribui-se peso 1. E.g.: *Acidente Cerebral Vascular* é *sinonimo* de *AVC*.

- anota: é uma relação semântica entre um nodo $\in T$ e um nodo $\in R$, constituindo assim uma anotação semântica. Esta relação determina um campo de metadados "keyword"de um determinado Recurso de Informação. E.g.: o termo *Acidente Cerebral Vascular anota* o Recurso de Informação *Recurso*1, ou seja, *Recurso*1 tem como palavra-chave este termo já citado.

- específica de domínio: são relações adicionais que podem ser capturadas pelo tipo de domínio em que está situada a RS. No caso do domínio da saúde, pode-se inferir uma relação de causa-efeito entre dois nodos $\in T$, sendo um deles uma doença e outro uma parte do corpo humano. Essas relações adicionais enriquecem a RS, pois atribuem mais conhecimento à ontologia. E.g: *Acidente Cerebral Vascular acomete Cerebro*.

  Para compreender esta definição temos na **Figura 13** um exemplo de RS adaptada, onde os nodos em verde (por exemplo, *AVC*) $\in T$ são os termos do vocabulário controlado, os nodos em roxo (*Recurso*1 e *Recurso*2) $\in RI$ são os RIs (ou recursos de informação), as arestas tracejadas $\in R$ caracterizam a relação semântica *sinonimo* e as arestas pontilhadas $\in A$ são as anotações semânticas dos RIs com os termos do vocabulário controlado.

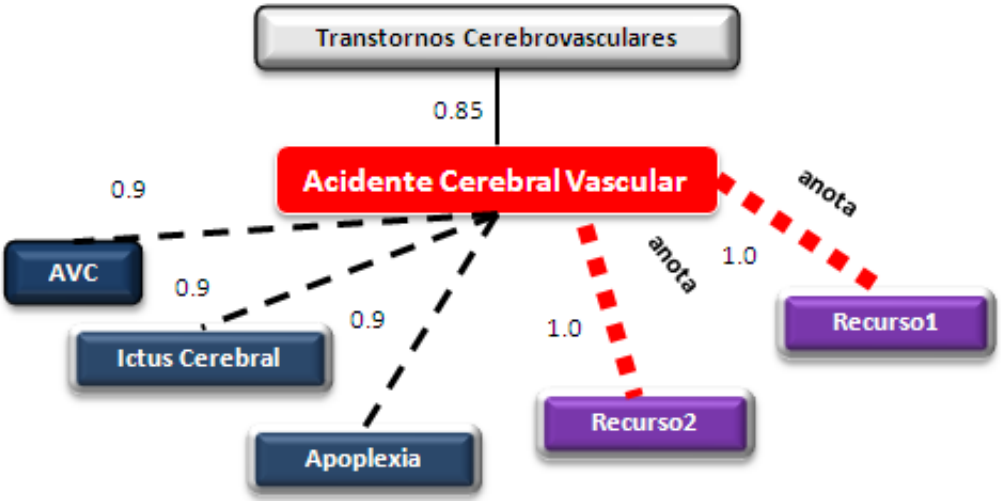


Figura 13: Rede Semântica adaptada.

Em uma rede semântica os nodos apresentam os conceitos de grau e fan-out, relacionados à quantidade de arestas associadas a cada nodo. Estes conceitos podem determinar a execução do *Spreading Activation* em um nodo, podendo receber um tratamento especial no algoritmo para que ele não degenere os resultados. Estes conceitos são apresentados a seguir:

- grau: é a quantidade de arestas que incide sobre um determinado nodo i, ou seja, a conectividade de i. Quanto maior a conectividade, maior o grau de i. O significado do grau varia de acordo com o tipo do nodo e dos tipos das arestas. Caso o nodo $i \in T$ geralmente um grau elevado de i indica que ele possui muitos sinônimos ou é um termo muito genérico que possui diversas especializações. Entretanto se o nodo $i \in RI$ possuir um grau elevado tem-se o indício de que ele possui um conjunto de *keywords* extenso. E.g.: na **Figura 14** a seguir o nodo *Acidente Cerebral Vascular* tem grau 5 e o nodo *AVC* tem grau 2.

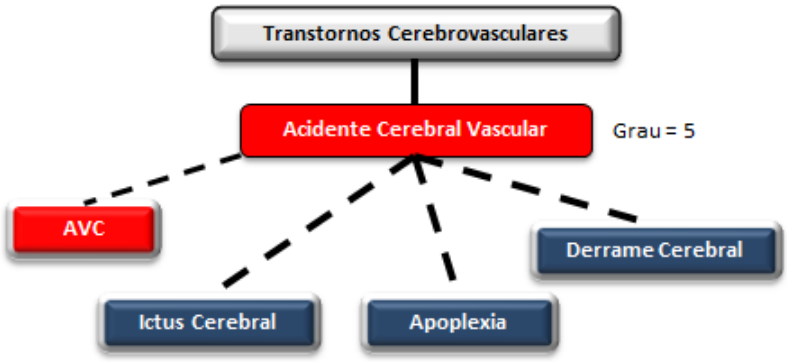


Figura 14: Exemplo de grau de um nodo.

- fan-in: determina a quantidade de nodos adjacentes a i já foram visitados, ou seja, quais os nodos "anteriores"a i no percorrimento da RS. E.g.: se *AVC* for marcado como semente em uma execução do SA, no segundo pulso o fan-in de *Acidente Cerebral Vascular* é 1, pois o percorrimento veio apenas de *AVC*.

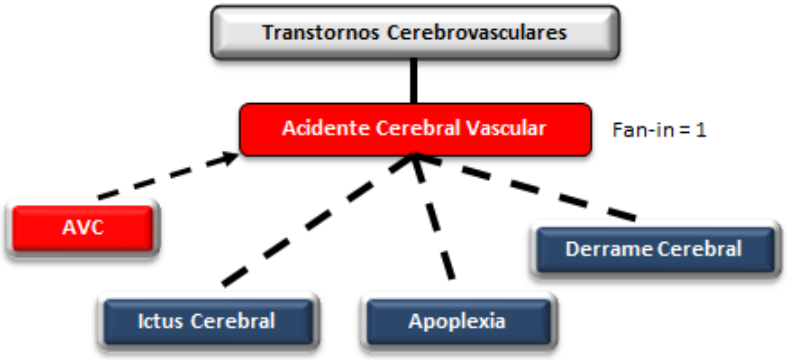


Figura 15: Exemplo de fan-in de um nodo.

- fan-out: este valor determina a quantidade de nodos adjacentes a *i* que não foram visitados, ou seja, quais são os "próximos"nodos visitados na RS. Este valor indica a quantidade de nodos adjacentes a *i* que serão ativados no SA. Este fator determina um dos critérios de parada do SA, pois um nodo *i* com fan-out elevado pode indicar um termo muito genérico na RS, e isto é facilmente obtido pelos tipos das relações de suas arestas. E.g.: se *Acidente Cerebral Vascular* for marcado como semente em uma execução do SA, sendo que *AVC* já foi ativado anteriormente, então o fan-out de *Acidente Cerebral Vascular* é 4.

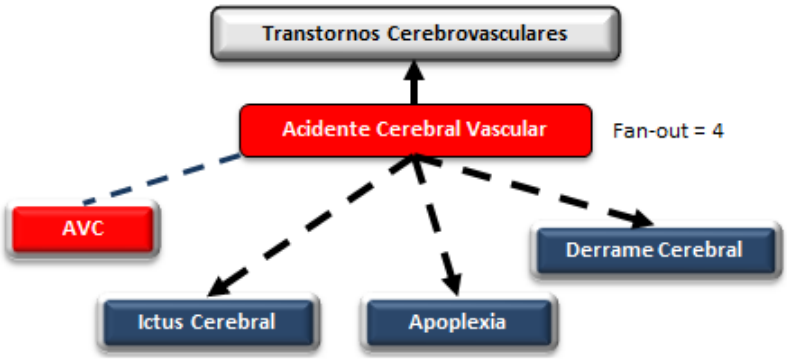


Figura 16: Exemplo de fan-out de um nodo.

4.3 *SPREADING ACTIVATION* ADAPTADO

Como visto no capítulo 3, o algoritmo *Spreading Activation* implementa buscas semânticas em uma Rede Semântica. Conhecendo a Rede Semântica adaptada pode-se apresentar o *Spreading Activation* adaptado para esta RS, de maneira a executar buscas semânticas que recuperem Objetos de Aprendizagem.

O algoritmo desenvolvido executa o SA sobre a RS adaptada, considerando os tipos das associações semânticas entre os nodos, sendo que cada uma delas tem um valor numérico associado. A especialização do SA está situada principalmente na atribuição de pesos distintos para as arestas, de acordo com a associação semântica. Outra característica é ajuste dos valores iniciais dos parâmetros, de maneira controlar os pulsos. A seguir temos a modificações (em negrito) do algoritmo 3:

```
1  TipoRelacao = { ANONIMA, ESPECIFICA, SINONIMO,ANOTACAO};
2  DirecaoRelacao = { GENERALIZACAO, ESPECIALIZACAO};
3  float ThresholdDisparo = 0.85;
4  float fatorDecaimento = 0.85;
5  int maxPulsos = 40;
   /* Inicia o conjunto de sementes K                                    */;
6  ...;
   /* Prepara adjacentes para disparo                                    */;
7  ...;
8  while contPulsos < maxPulsos and nodosMarcadosParaDisparo ≠ ∅ do
9      for n ∈ nodosMarcadosParaDisparo do
           /* Dispara nodo n                                             */;
10         ...;
11         contPulsos ← contPulsos + 1;
12     end
13 end
```

Algorithm 3: Rotina principal do Algoritmo de *Spreading Activation* adaptado

Com a adição da enumeração **TipoRelacao**, cada aresta possui um peso específico deve ser considerado na atualização do valor de ativação dos nodos. Além do peso de cada tipo de relação adicionamos um peso para o sentido desta relação, de maneira a distinguir a relevância semântica entre generalização e especialização, partindo da idéia de que especializar conhecimento é mais importante que generalizar. Desta maneira introduzimos um novo fator de peso de sentido do um link (S) no cálculo do valor de ativação de um nodo, apresentado na **Definição 4**:

Definição 4 (Atualização de um nodo no SA adaptado): A atualização do valor de ativação de um nodo i ativado no SA dada por

$$A_i(p) = A_i(p-1) + \sum_{j}^{n} (A_j(p-1)) * (W_{ji}) * (D) * (S)$$

onde:

- $A_i(p)$ é o valor de ativação do nodo i no pulso p.
- $A_i(p-1)$ é o valor de ativação do nodo i no pulso $p-1$.
- $A_j(p-1)$ é o valor de ativação do nodo j na iteração $p-1$.

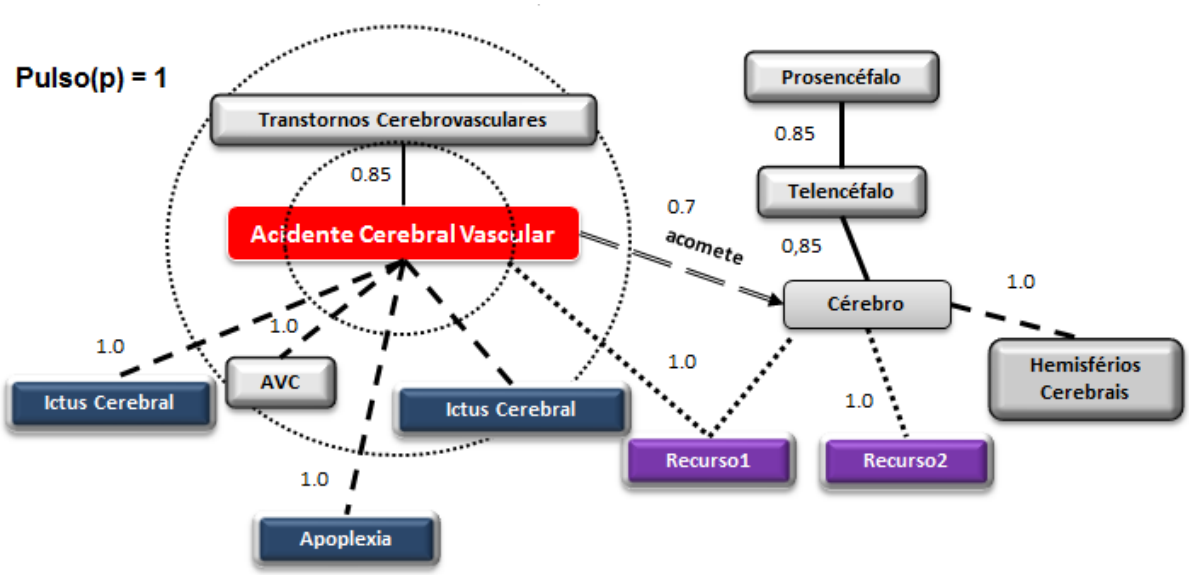
- W_{ij} é o peso associado ao link que conecta o nodo i ao nodo j.
- D é o fator de decaída. O valor de D é um número real $\in [0-1]$.
- **S é o peso associado a direção do link ij. O valor de S é um número real $\in [0-1]$.**
- n é o número de arestas que incidem no nó k

A seguir temos o algoritmo adaptado:

```
1 calcularAtivacao(n);
2 Adjacentes ← retornaAdjacentesDe(n);
3 float somaLinks ← 0;
4 for j ∈ Adjacentes do
5     if notDisparado(j) then
6         somaLinks ← somaLinks + tipoAresta(j,n).peso() * fatorDecaimento *
          sentidoLink(i, j);
7     end
8 end
9 ativacao(i) ← ativacao(i) + somaLinks;
```

Algorithm 4: Rotina de atualização de um nodo i do Algoritmo de *Spreading Activation* adaptado

Este algoritmo implementado retorna uma lista de RIs ordenados de acordo com o valor de ativação final, ou seja, sua relevância semântica. No exemplo a seguir o trecho da RS anteriormente visto recebe os pesos em cada tipo de relação e o SA é executado tendo *AVC* como semente. Com, isso, *AVC* recebe valor de ativação inicial 1 e no primeiro pulso o valor de ativação de *Acidente Vascular Cerebral* é atualizado para 0.76, de acordo com os cálculos apresentados.



$$GENERALIZACAO = 0.8$$
$$ESPECIALIZACAO = 0.9$$
$$i = AcidenteVascularCerebral$$
$$j = \{Apoplexia, Recurso1, Cérebro, TranstornosCerebrovasculares\}$$
$$A_i(p) = A_i(p-1) + \sum_{j}^{n} (A_j(p-1)) * (W_{ji}) * (D) * (S)$$
$$A_1(0) = 0 + (1*1*0.85)*0.9 + (0*0*0.85)*0.9 + (0*0.85*0.85)*0.9 + (0*0.7*0.85)*0.9$$
$$A_1(0) = 0{,}76$$

Figura 17: Exemplo de execução do SA.

Este processo de atualização de valores de relevâncias semânticas dos nodos da RS persiste até que uma das condições de parada (*threshold*, número de ciclos, fan-out) seja alcançada. O resultado é um *ranking* semântico ordenado de forma que os nodos com relevância final mais próximos de 1 estejam no topo, seguidos dos demais até que se chegue na última ocorrência tolerada pelo *threshold*. Entretanto este *ranking* semântico é uma das formas de apresentação de resultados de uma busca, pois o usuário pode utilizar a busca sintática que retornará outro *ranking*. A seção a seguir apresenta este problema.

4.4 CÁLCULO DAS RELEVÂNCIAS GLOBAIS

A implementação *Spreading Activtion* atua sobre uma RS composta por termos de um vocabulário unidos a Objetos de Aprendizagem através de relações semânticas. A relação de anotação consiste no preenchimento de um campo dos metadados de um OA, o campo *keyword*. Entretanto o demais metadados de um OA (autor, título, data, etc.) são contemplados

pelo mecanismo de buscas sintáticas. A implementação do SA independente do mecanismo sintático gera um novo problema ao usuário: ele terá que realizar buscas separadas que retornarão *rankings* distintos. Por exemplo, se um usuário procura OAs informando *keyword* e nome do autor os dois mecanismos são utilizados e dois *rankings* são gerados, como apresenta a figura a seguir:

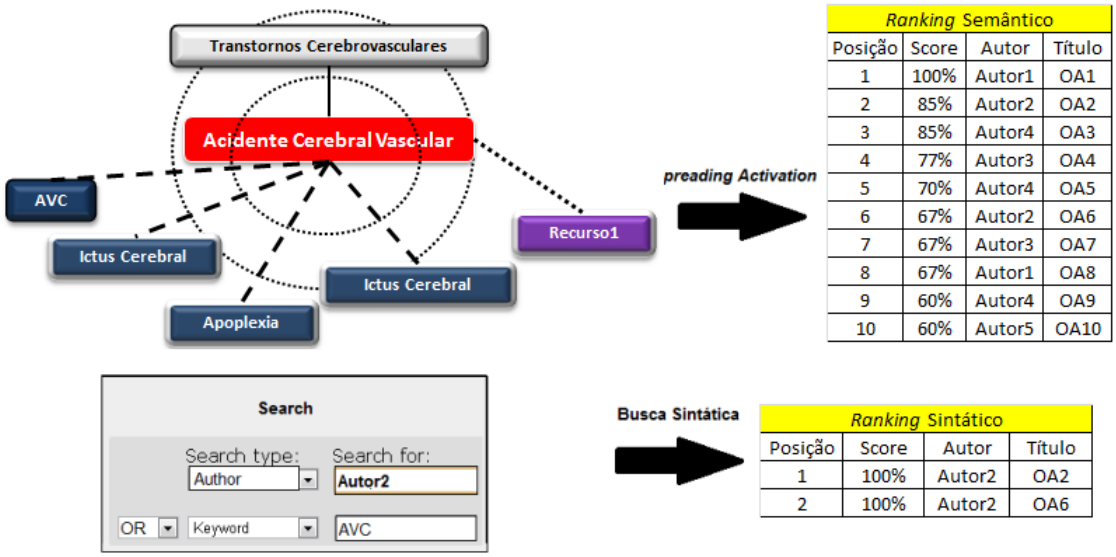


Figura 18: Exemplo de consulta nos dois mecanismos de busca.

Uma solução para este problema é a **composição** das duas buscas. Para realizar tal tarefa, os dois mecanismos de busca são ativados, os resultados dos *rankings* são combinados de acordo com o fator de importância α, resultando em uma única listagem final.

O CIBELE provê funções que padronizam as chamadas a cada uma das buscas e fazem a composição dos *rankings*. Os repositórios de RIs são plataformas *Web*, logo a sua estrutura de páginas pode receber adição destas funções, tornando o processo de buscas transparente ao usuário, ou seja, o usuário não sabe qual busca específica está chamando (sintática ou semântica), ele apenas recebe o *ranking* final de sua consulta. A seguir temos a função verificaConsulta(), que determina e chama as buscas que serão executadas de acordo com os campos de metadados que o usuário especificou.

```
1  verificaConsulta(formSintatico, formSA);
2  List rankingSintatico, rankingSA , rankingFinal = {};
3  float alpha = 0.5;
4  boolean sintatico, semantico;
5  if formSintaticoPreenchido then
6  |   rankingSintatico ← executaBuscaSintatica(formSintatico);
7  |   sintatico = true ;
8  end
9  if formSAPrenchido then /* formSA é o conjunto de sementes K          */
10 |   rankingSA ← executaSA(formSA);
11 |   semantico = true ;
12 end
13 if sintatico and semantico then
14 |   rankingFinal ← compoeRelevancias(alpha, rankingSintatico, rankingSA); break;
15 end
16 if sintatico then
17 |   rankingFinal ← rankingSintatico;
18 end
19 else
20 |   rankingFinal ← rankingSA;
21 end
22 retorna rankingFinal;
```

Algorithm 5: Função de verificação de consultas no CIBELE

Caso ambas as buscas sejam disparadas a função compoeRelevancias() é executada para gerar o *ranking* composto $R_composto$ ponderado pelo fator α, de acordo com a fórmula:

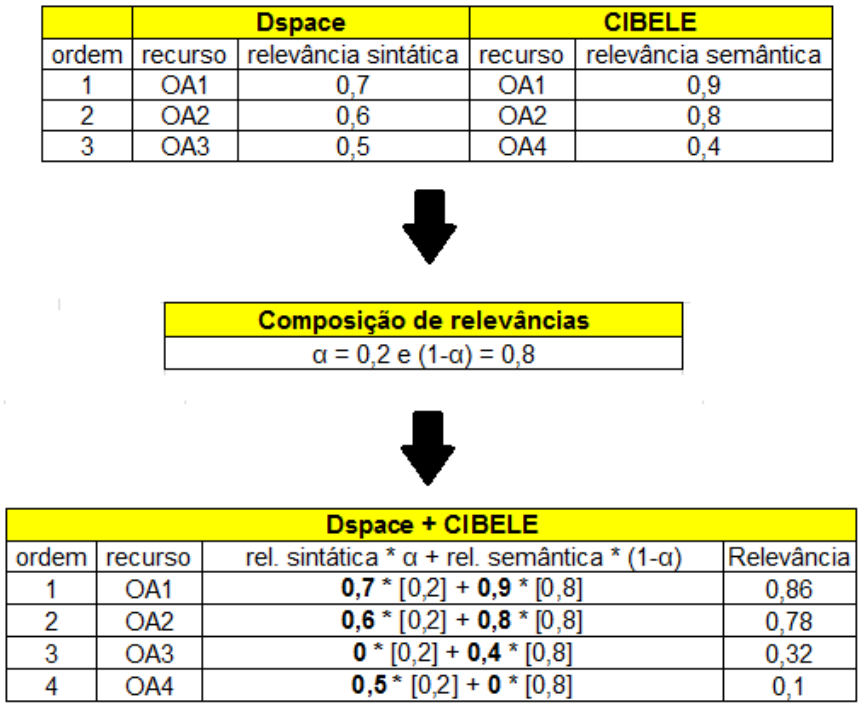
$$R_{composto} = R_{SA} * (\alpha) + R_{sintatico} * (1 - \alpha)$$

A seguir temos a descrição desta função.

1 **compoeRelevancias**(α, *rankingSintatico*, *rankingSA*);
  `/* Multiplica todos os elementos de cada ranking por alpha */`;
2 ponderaRanking(α, *rankingSA*);
3 ponderaRanking($(1-\alpha)$, *rankingSintatico*);
  `/* concatena os rankings */`;
4 var *rankingFinal* = concatena(*rankingSintatico*, *rankingSA*);
  `/* coloca os elementos do rankingFinal em ordem decrescente de relevância */`;
5 *rankingFinal* $\leftarrow$ ordena(*rankingFinal*);
6 retorna *rankingFinal*;

Algorithm 6: Função de composição dos rankings sintatico e semantico

Para compreender melhor esta composição vamos apresentar o seguinte exemplo na **Figura 19**: dados os Recursos de Informação *OA*1, *OA*2, *OA*3 e *OA*4 foram feitas consultas com os mecanismos sintático e semântico e cada uma destas consultas retornou um valor de relevância para estes RIs em questão. Determinamos um fator de relevância $\alpha = 0.2$, ou seja, 20 % do *ranking* final composto tem peso relativo a busca sintática e 80% relativo a busca semântica. Desta maneira o usuário executa uma única consulta e obtém um *ranking* final único e utiliza o potencial dos dois mecanismos de buscas apresentados.



	Dspace		CIBELE	
ordem	recurso	relevância sintática	recurso	relevância semântica
1	OA1	0,7	OA1	0,9
2	OA2	0,6	OA2	0,8
3	OA3	0,5	OA4	0,4

Composição de relevâncias
α = 0,2 e (1-α) = 0,8

Dspace + CIBELE			
ordem	recurso	rel. sintática * α + rel. semântica * (1-α)	Relevância
1	OA1	**0,7** * [0,2] + **0,9** * [0,8]	0,86
2	OA2	**0,6** * [0,2] + **0,8** * [0,8]	0,78
3	OA3	**0** * [0,2] + **0,4** * [0,8]	0,32
4	OA4	**0,5** * [0,2] + **0** * [0,8]	0,1

Figura 19: Exemplo de composição de relevâncias.

5 IMPLEMENTAÇÃO DA PROPOSTA

Implementou-se um protótipo do módulo de buscas semânticas do CIBELE em Java. Este protótipo utiliza o SA adaptado. Embora tal módulo possa ser acoplado a um repositório de conteúdo digital, para facilitar nossos testes, simulamos tal acoplamento. A **Figura 20** apresenta a arquitetura implementada, identificando os pacotes e bibliotecas utilizados. O processo de preparação e utilização do sistema segue os seguintes passos: (1) definição da ontologia utilizada pelo sistema; (2) inserção dos objetos de informação e anotação semântica desses objetos; (3) ajuste dos parâmetros de configuração do *Spreading Activation* (SA); (4) geração da rede semântica (RS), valendo-se da API do Jena; (5) realização de buscas sobre a RS. Cada busca é especificada por um conjunto de palavras-chave fornecidas pelo usuário e processada nas seguintes etapas: (5a) geração da *Spreading Activation Network* (SAN) para processamento da busca sobre a RS usando SA; (5b) ativação de nodos a cada pulso do SA; (5c) cálculo do ranking de relevância semântica dos objetos retornados para o usuário sobre o conjunto de nodos ativados e suas relevâncias após a execução dos pulsos aramazenadas em SAN'.

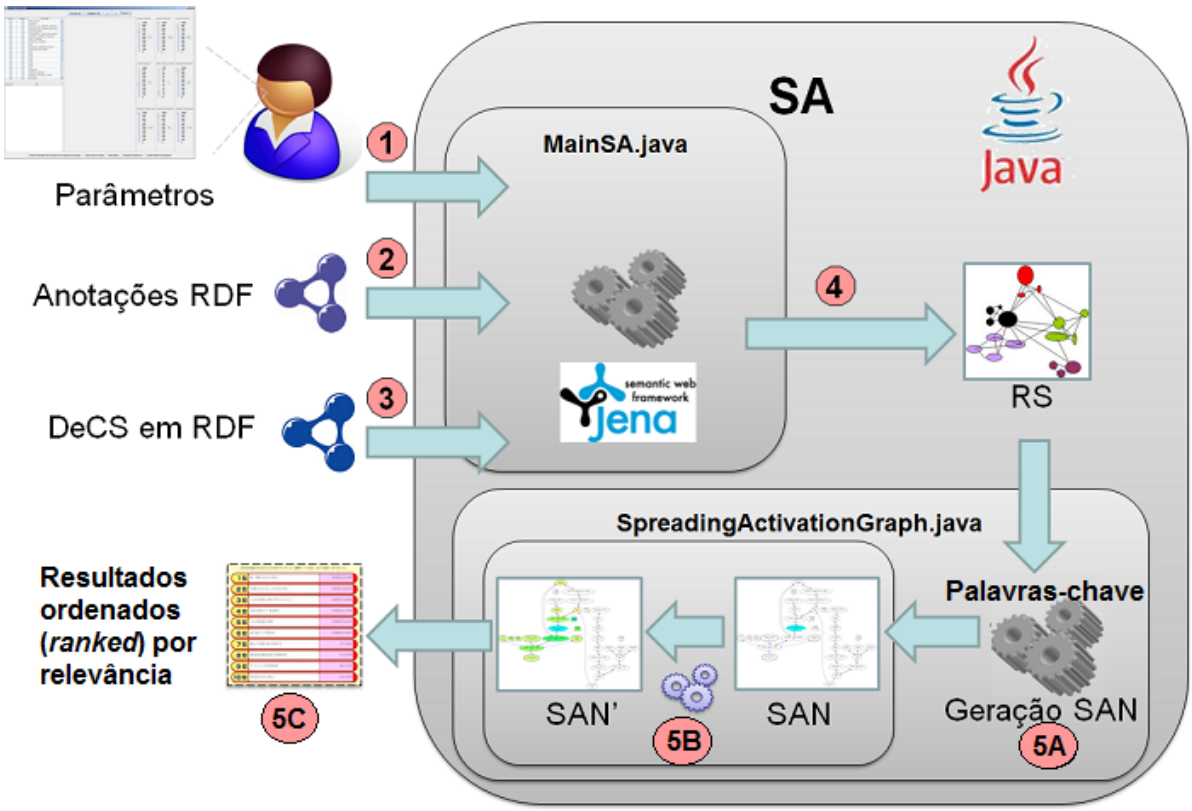


Figura 20: Arquitetura do SA implementado.

5.1 VOCABULÁRIO CONTROLADO DECS

A ontologia utilizada para catalogação e busca de objetos de aprendizagem da UnA-SUS, foi produzida a partir de um extrato do vocabulário controlado (VC) denominado *Descritores em Ciências da Saúde* (DeCS) [1]. O DeCS começou a ser elaborado em 1986 pelo Centro Latino-Americano e do Caribe de Informação em Ciências da Saúde (BIREME) e originário do *Medical Subject Headings* (MeSH), o qual existe desde 1963 e é de responsabilidade da *United States National Library of Medicine*. A BIREME mantém uma equipe voltada para a atualização e o aperfeiçoamento contínuo do DeCS, lançando uma nova versão a cada ano, praticamente. A versão do DeCS utilizada em nossos experimentos foi a de 2010.

O DeCS é um vocabulário estruturado trilíngue (português, espanhol e inglês) baseado em coleções de termos, organizados para facilitar o acesso à informação na área de saúde. Foi criado para servir como uma referência léxica e semântica na descrição de artigos de revistas científicas, livros, anais de congressos, relatórios técnicos, e outros tipos de materiais, assim como para ser usado na pesquisa e recuperação de assuntos da literatura científica nas fontes de

[1] decs.bvs.br

informação disponíveis na Biblioteca Virtual em Saúde (BVS) tais como LILACS, MEDLINE e outras.

Figura 21: Exemplo de consulta no DeCS.

O DeCS apresenta uma estrutura hierárquica de conceitos que permite a execução de pesquisas em termos mais amplos ou específicos. Ele é um vocabulário dinâmico que apresenta um total 30.369 descritores, sendo destes 25.671 do MeSH e 4698 exclusivamente do DeCS. Ele é atualizado anualmente e em 2010 apresenta 20 categorias:

- Anatomia [A]
- Organismos [B]
- Doenças [C]
- Compostos Químicos e Drogas [D]
- Técnicas Analíticas, Diagnósticas e Terapêuticas e Equipamentos [E]
- Psiquiatria e Psicologia [F]
- Fenômenos e Processos [G]
- Disciplinas e Ocupações [H]
- **Homeopatia** [HP]
- Antropologia, Educação, Sociologia e Fenômenos Sociais [I]
- Tecnologia, Indústria, Agricultura [J]
- Ciências Humanas [K]

- Ciência da Informaçõo [L]
- Denominaçõees de Grupos [M]
- Assistência à Saúde [N]
- **Ciência e Saúde** [SH]
- **Saúde Pública** [SP]
- Características de Publicações [V]
- **Vigilância Sanitária** [VS]
- Denominações Geográficas [Z]

As 4 categorias em negrito foram especialmente desenvolvidas no Brasil visando melhor representar a literatura brasileira gerada. Os conceitos do vocabulário DeCS, na versão 2010, estão assim distribuídos:

- 25,8% referem-se a **Compostos Químicos e Drogas** (categoria D);
- 20,4% do total são da área de **Anatomia** (categoria A), de organismos (categoria B) e de **Fenômenos e Processos** (categoria G);
- 12,9% do total são referentes a **Doenças** (categoria C);
- 21,6% são representados pelas áreas de **Técnicas e Equipamentos** (categoria E), ciências afins (categorias F, H, I, J, K, L, M, N), **Características de publicações** (categoria V) e **Denominações Geográficas** (categoria Z)
- 10,2% do total de conceitos referem-se a área de **Saúde Pública** (categoria SP)
- 5,7% do total de conceitos referem-se a **Homeopatia** (categoria HP)
- 2,4% do total de conceitos referem-se a **Vigilância Sanitária** (categoria VS)
- 0,6% do total de conceitos referem-se a **Ciência e Saúde** (categoria SH)

A obtenção do DeCS deu-se através do trabalho do pesquisador Dr. Divino Ignácio Ribeiro Júnior, que a partir do Serviço DeCS/XML [2], capturou e converteu o VC em uma base de dados relacional. Este processo deu-se em duas etapas:

[2] disponibilizado em http://decs.bvsalud.org/vmx.htm

a) desenvolvimento de um *crawler* para captura dos descritores, sinônimos, definições, taxonomia e relacionamentos horizontais nos seus 3 idiomas, com o qual cada descritor é salvo em um arquivo XML. A validação da captura é realizada por amostragem das categorias, examinando-se o conteúdo obtido e a informação correspondente fornecida no site do Serviço DeCS/XML. O *crawler* captura os descritores guiando-se pelos códigos hierárquicos de tais descritores.

b) conversão dos arquivos XML salvos, por meio de um *script* que popula um banco de dados MySQL previamente preparado para este fim.

Esta base dados é convertida em um modelo RDF através de um utilitário produzido pelo mestrando Wanderson Rigo, o qual implementa um algoritmo valendo-se da biblioteca JENA para realizar esta conversão. Os arquivos de cada categoria do DeCS são carregados no *Spreading Activation*, juntamente com as anotações de RIs catalogados, caracterizando a Rede Semântica onde o SA atuará. A figura a seguir apresenta esquematicamente este trabalho de geração da RS:

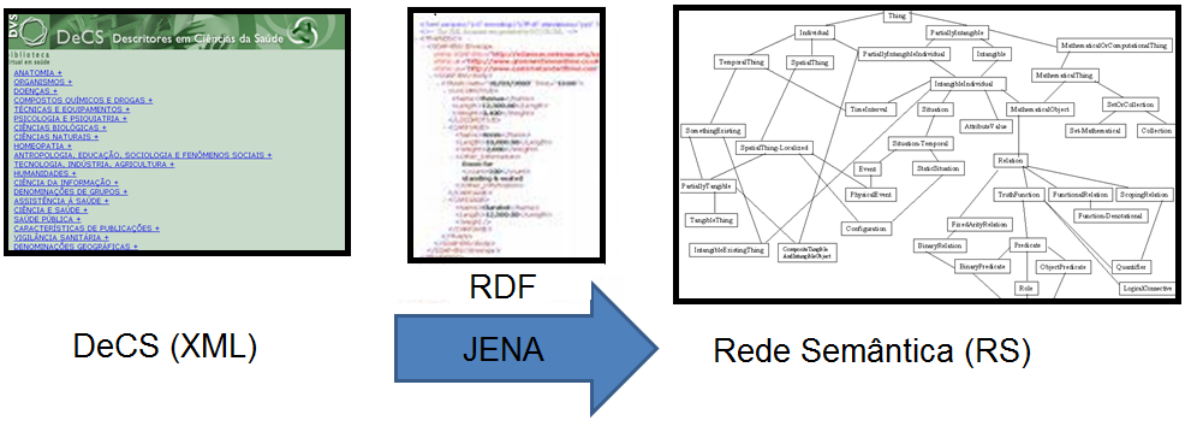


Figura 22: Processo de geração da RS a partir do DeCS.

O *workflow* de geração da *SAN* para a execução do SA é detalhado na **Figura 23**, partindo-se de uma Base de Dados Relacional do DeCS, via um algoritmo implementado por uma classe utilitária (OntologyGenerator) são gerados arquivos RDF que são interpretados pelo JENA. Em seguida, os artefatos gerados entram no fluxo de informações descritos pela **Figura 20**.

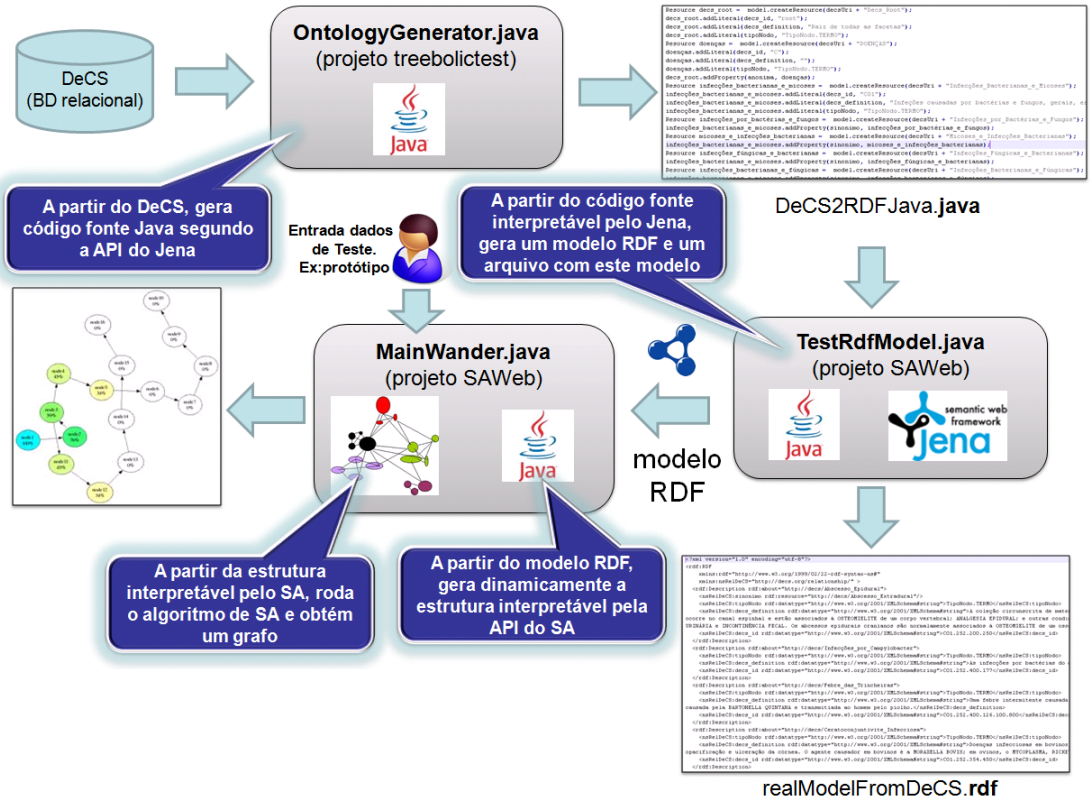


Figura 23: *Workflow* de geração da *SAN*.

5.2 REPOSITÓRIO DE CONTEÚDO

A implementação de repositório de conteúdo selecionada foi a ferramenta *DSpace* [3], um gerenciador de conteúdos digitais de código aberto que permite as personalizações propostas pelo CIBELE. A **Figura 24** ilustra a tela inicial do repositório DSpace UnA-SUS:

[3]http://dspace.org

Figura 24: Repositório UnA-SUS.

O DSpace (Bass et al. 2003) é um repositório digital desenvolvido em Java pelo *Massachusetts Institute of Technology - MIT* e *Hewlett-Packard - HP* e objetiva armazenar, gerir e disseminar conteúdo digital. Ele pode ser livremente adaptado e expandido funcionalmente, nos termos da *BSD Open source license*. O DSpace possibilita a criação de coleções de recursos, as quais aglutinam um conjunto de recursos afins. Permite a definição de *workflows* específicos para o preenchimento de metadados descrevendo os recursos de cada coleção. O DSpace faz uso do PostgreSQL para a persistência dos dados. O DSpace é flexível quanto aos tipos de interface de acesso, disponibilizando:

JSPUI: não é modular, porém mais fácil de se trabalhar, admite alterações pontuais. Foi a interface adotada neste trabalho.

XMLUI: segue o paradigma de *templates* de interface, componentizando os elementos gráficos. Porém alterações pontuais exigem grande esforço, sobretudo se elementos não nativos precisam ser incorporados, como os módulos do CIBELE.

As anotações semânticas são gravadas em uma Base de Dados à parte e não na base do DSpace, justamente para manter o menor acoplamento possível entre o repositório e o CIBELE.

5.3 MECANISMO DE BUSCAS SEMÂNTICAS NO CIBELE

O algoritmo de *Spreading Activation* implementado utiliza a biblioteca *Texai Spreading Activation* [4]. Ela fornece uma API para a construção programática de uma *Spreading Activation Network - SAN*, uma implementação de Rede Semântica que permite a execução do algoritmo de SA. Esta biblioteca utiliza o modelo do SA puro e disponibiliza métodos de processamento que recebem parâmetros de configuração, possibilitando assim a execução personalizada do SA.

Para a implementação do SA adaptado procedemos desta forma:

- **familiarização com a biblioteca *Texai Spreading Activation***: este processo requer o entendimento das classes Java desta biblioteca, através do estudo do código-fonte disponibilizado e a execução do algoritmo de SA em seu estado original. A **Figura ??** apresenta um diagrama com as 3 principais classes desta biblioteca: *Link*, *Node* e *SpreadingActivationGraph*. A classe *Node* simplesmente modela um nodo da Rede Semântica. A classe *Link* determina as arestas entre os nodos e o peso associado a cada aresta. Finalmente, a classe *SpreadingActivationGraph* implementa o algoritmo de SA descrito no capítulo 3.

- **implementação do *Spreading Activation* adaptado**: algumas melhorias foram implementadas na biblioteca citada no item anterior:

  a) configuração dos parâmetros do SA: número de ciclos, *threshold*, pesos das relações, fator de decaída;

  b) adição de tipos de associações semânticas, através da criação do tipo *Enum TipoLink*, sendo que cada tipo de aresta recebe um valor de peso distinto.

Criou-se também uma classe *Normalizer* que tem a tarefa de fazer a normalização dos valores de ativação dos nodos, de maneira que este valor sempre fique no intervalo [0,1]. Além destas alterações de parâmetros, foi implementada a ordenação do *ranking* semântico e a criação de relações assimétricas (generalização e especialização, as quais, em nossa abordagem, possuem pesos distintos). A **Figura 25** apresenta esta classe e as enumerações adicionadas ao modelo original marcadas em vermelhos.

[4] http://sourceforge.net/projects/texai/files

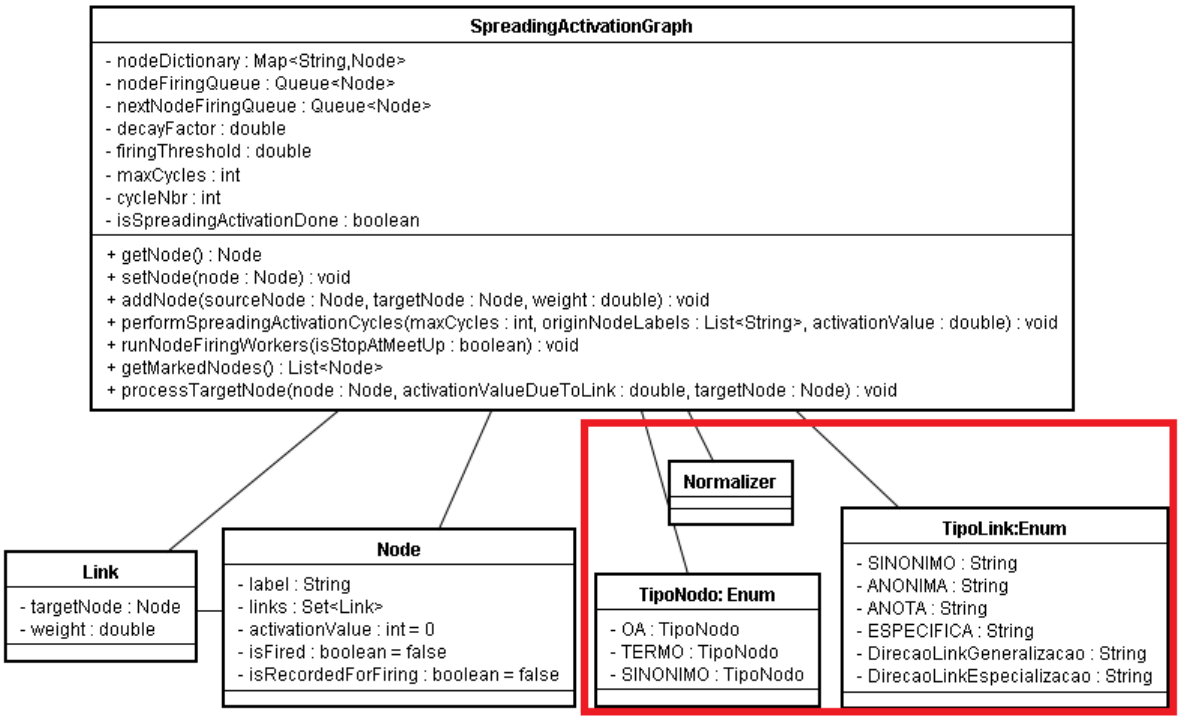


Figura 25: Diagrama de classes do módulo de busca do CIBELE.

O primeiro artefato obtido pela implementação do SA adaptado foi um protótipo do SA com interface gráfica em Java. A interface gráfica de tal protótipo disponibiliza *sliders* para ajustar o valor de cada um dos parâmetros do SA, de maneira a facilitar os testes, a determinação de um conjunto *default* de valores dos parâmetros e a depuração dos erros de execução.

A **Figura 26** apresenta este protótipo. Os *sliders* situados do lado direito do protótipo (1) definem os valores dos parâmetros do SA. Após a escolha dos parâmetros o usuário seleciona dentre os nodos disponíveis no lado esquerdo e no topo (2) o conjunto de sementes do SA. A execução é iniciada ou pausada quando o usuário pressiona o botão “Executar SA” (3). Com isso o SA é executado e figuras relativas a cada um dos pulsos são geradas e apresentadas no centro do protótipo (4). Ao final da execução do algoritmo o *ranking* dos nodos ordenado por suas relevâncias é apresentado em (5).

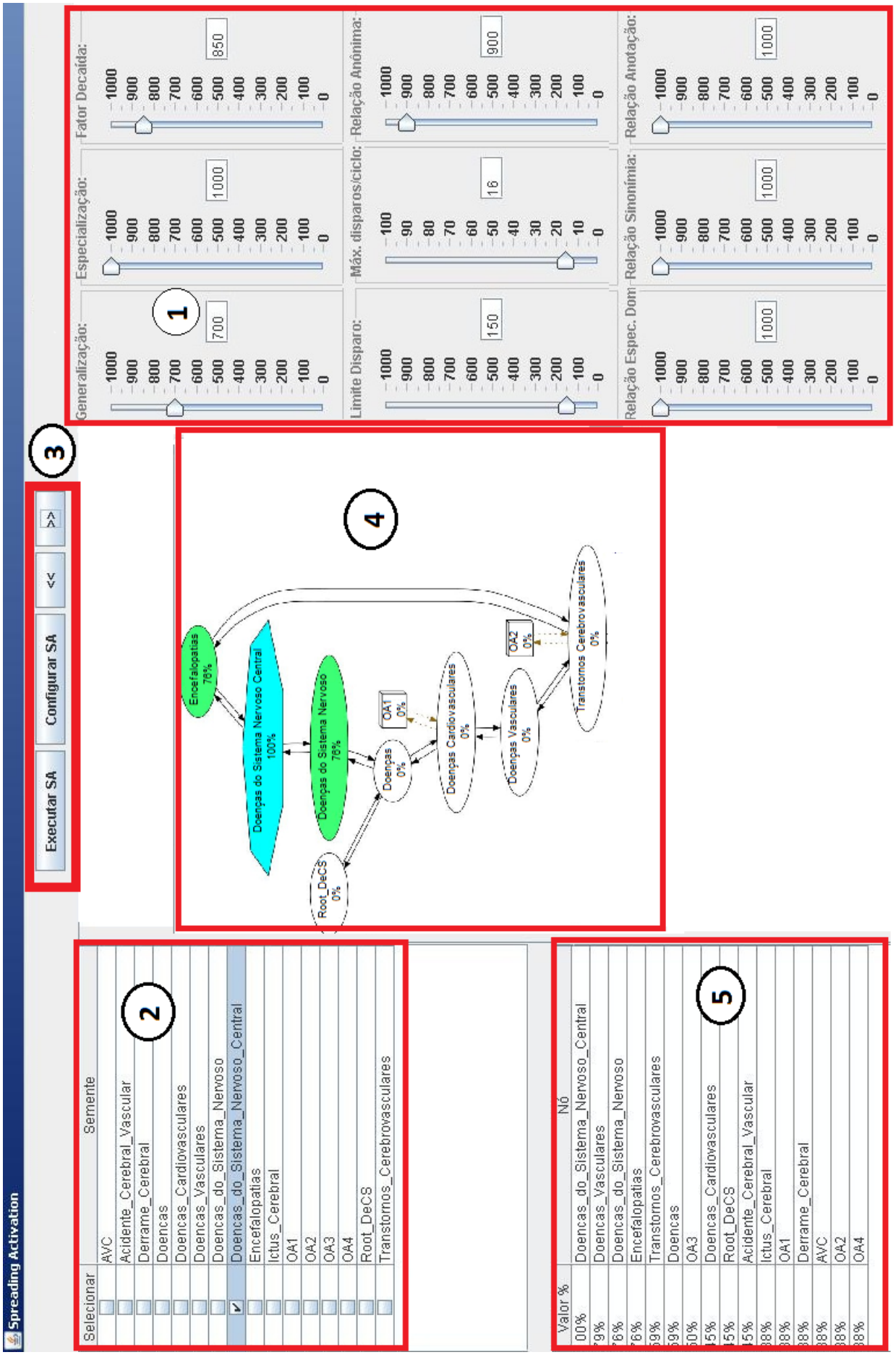


Figura 26: Protótipo do SA em Java.

5.4 INTEGRAÇÃO COM O DSPACE

Como dito anteriormente, os módulos do CIBELE foram concebidos com o objetivo de prover um mecanismo de busca semântica aos usuários de repositórios de RIs e manter um fraco acoplamento com tais repositórios. Este acoplamento é realizado via *HTTP Request*, como mostra a **Figura 27**. Nesta figura temos a apresentação dos 8 passos compreendidos entre a consulta do usuário e a obtenção da resposta e também os módulos envolvidos. Este processo funciona da seguinte maneira:

1. o usuário utiliza um módulo de interface auto-completar de consulta para selecionar o conjunto *K* de palavras-chave do vocabulário controlado (no caso o extrato do DeCS);
2. este conjunto *K* selecionado é encaminhado ao DSpace, para possível verificação e confirmação do usuário;
3. o conjunto *K* é submetido ao SA adaptado, via chamada *HTTP Request* do módulo implementado
4. o SA retorna a listagem dos resultados, com as respectivas relevâncias semânticas;
5. a listagem de resultados é ordenada pelas relevâncias semânticas;
6. a resposta é enviada ao DSpace;
7. o DSpace realiza buscas por outros campos de metadados que possam ter sido especificados (e.g., autor, data de publicação) e retorna os resultados ordenados pela relevância para tais campos;
8. as listas de resultados ordenadas (por relevância semântica em relação as palavras-chaves e relevância sintática frente aos outros campos de metadados, respectivamente) são combinadas usando os algoritmos 5 e 6 descritos na seção 4.4.

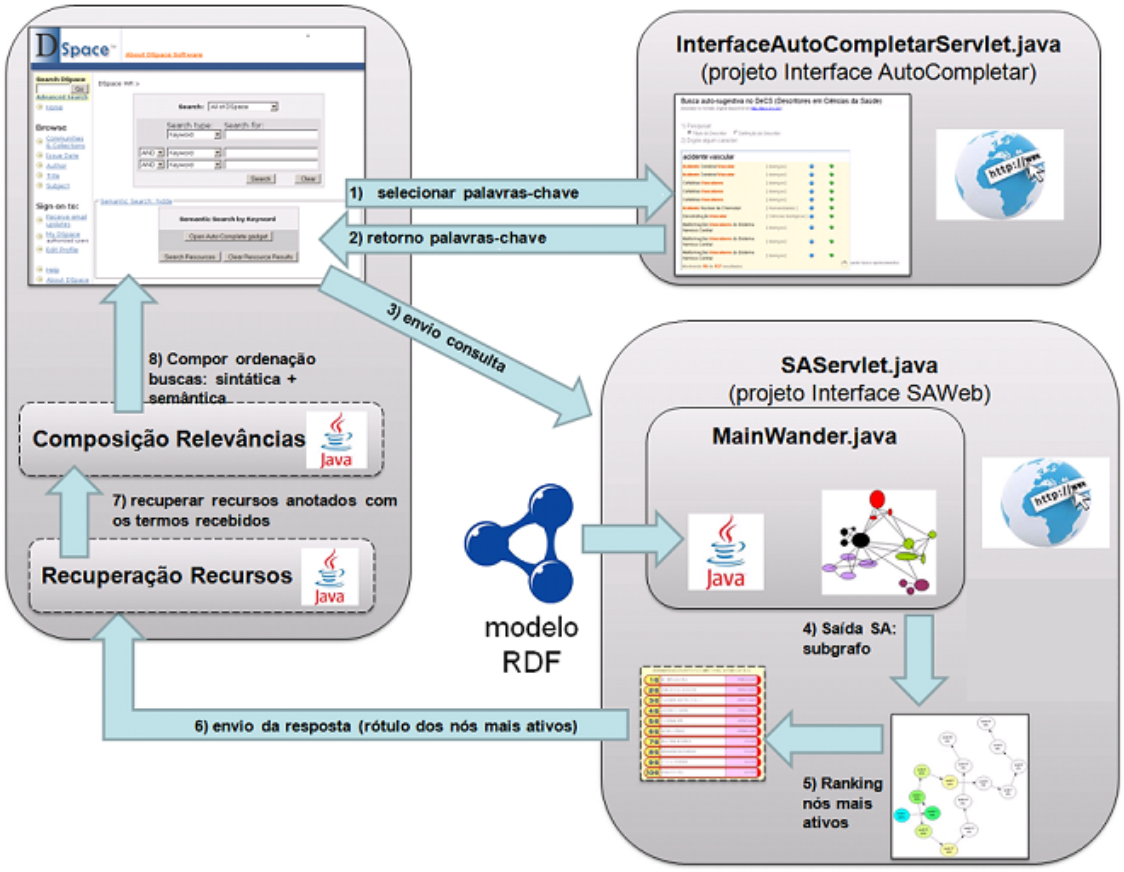


Figura 27: Integração do SA ao DSpace.

Os passos 3) e 6) enfatizam a modularidade do SA em relação ao repositório DSpace, pois a busca semântica é realizada sem qualquer dependência com o DSpace. Para isso, foram utilizadas tecnologias *Web*, com a criação de páginas *JSP* [5] e *scripts JavaScript* que utilizam a metodologia *Asynchronous Javascript and XML (AJAX)* [6] para acoplar o módulo Web do SA ao repositório DSpace, que é implementado em Java, JavaScript, JSP e HTML. A instalação padronizada do DSpace já produz a hierarquia de páginas e monta todo o repositório. Nosso foco está nas buscas, então apenas a página de buscas *default* "simple-search" é alterada e uma página adicional para o SA "advanced-search" é criada. A seguir temos uma ilustração do *link* adicionado na página inicial do DSpace:

[5] http://www.oracle.com/technetwork/java/javaee/jsp/
[6] api.jquery.com/category/ajax

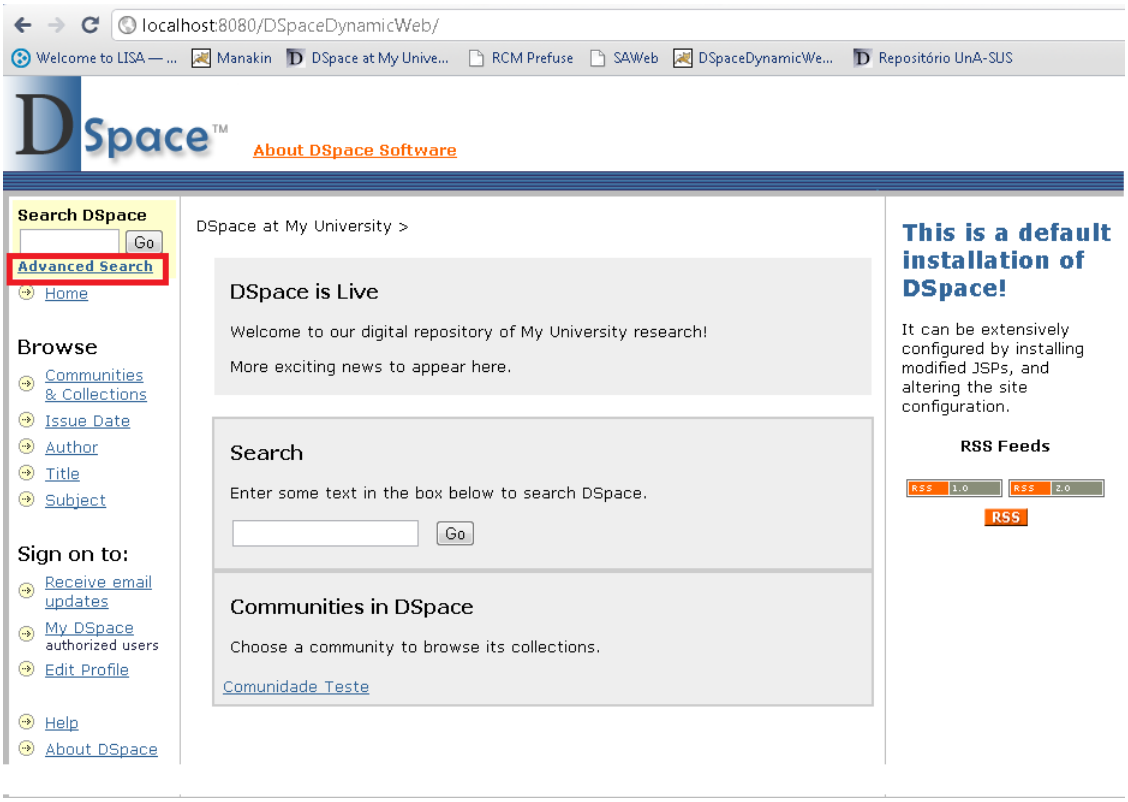


Figura 28: Página inicial do DSpace.

Quando a opção “busca avançada” é selecionada o usuário é direcionado para a página de busca sintática avançada, entretanto esta página foi modificada para comportar também buscas semânticas. A figura abaixo mostra esta modificação:

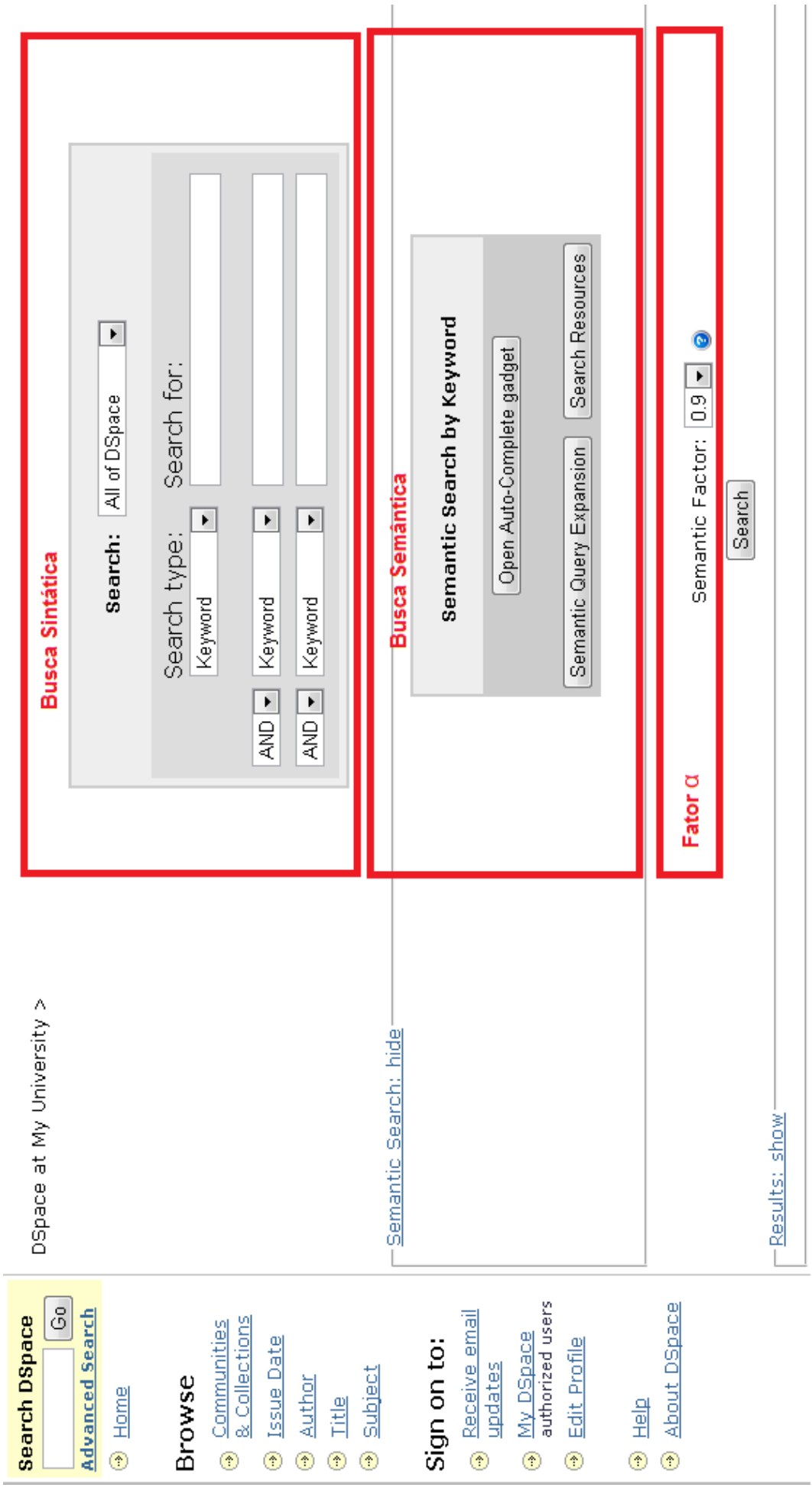


Figura 29: Busca Avançada no DSpace.

O usuário deve selecionar as palavras-chave para compor o conjunto de sementes do SA. Para isso utiliza-se uma interface de auto-completar, outro módulo do CIBELE (para maiores detalhes ver o trabalho de (Rigo et al. 2010)). A interface de auto-completar é apresentada em destaque na **Figura 30**:

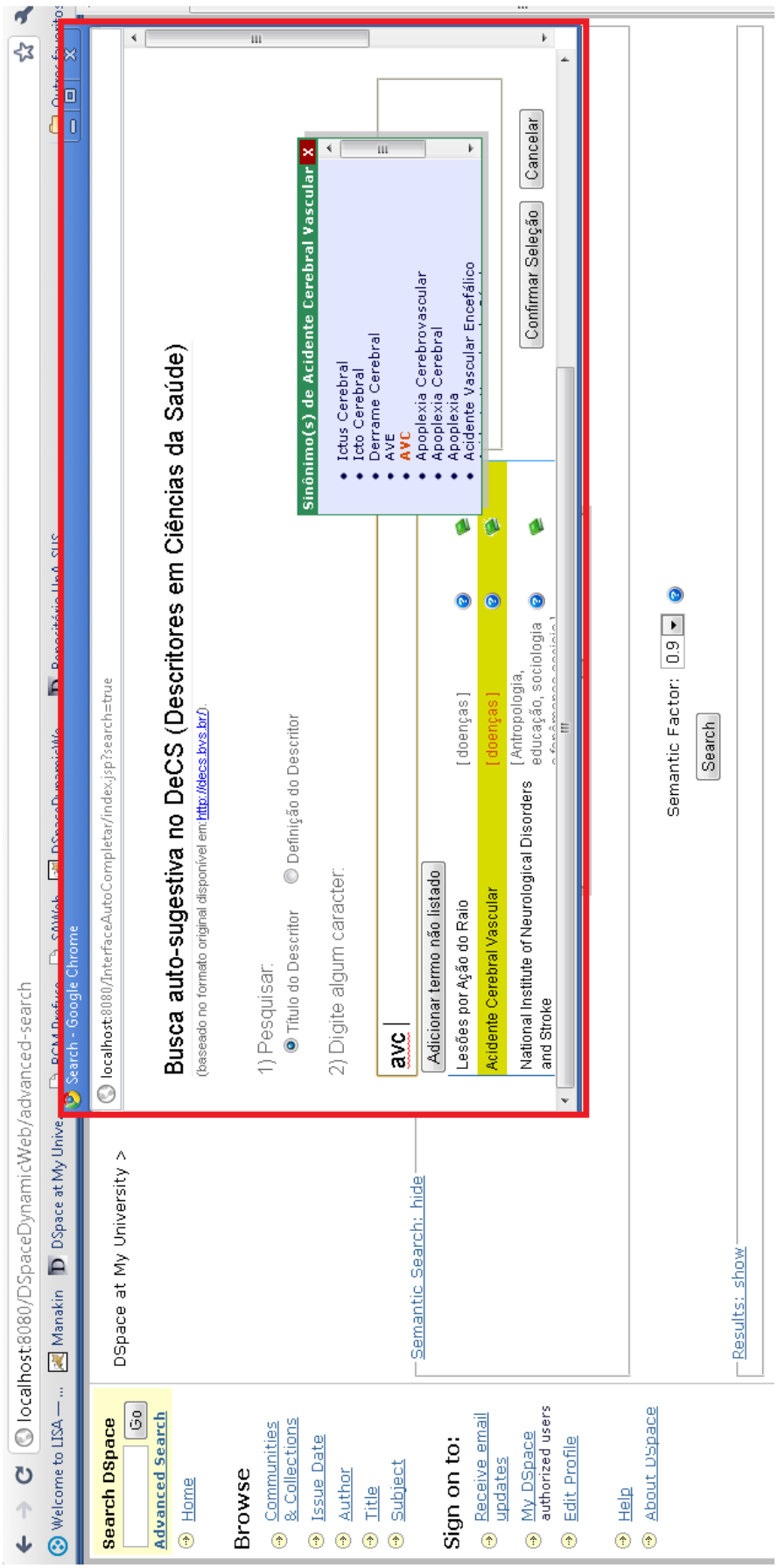


Figura 30: Interface de Auto-Completar no DSpace.

Após selecionar as sementes, o conjunto de sementes é enviado ao SA, que então executa, retorna os resultados que são ponderados pelo fator α definido no campo "Semantic Factor". O SA proporciona buscas semânticas no repositório, as quais ocorrem em paralelo a buscas sintáticas nativas do repositório. Este fato motivou a idéia de se **compor** estas 2 buscas, de maneira que o usuário possa executá-las e obter uma resposta única, potencializando a efetividade do sistema de buscas do repositório. Para isso, foi criada uma rotina que formata os resultados das 2 buscas em *JSON - JavaScript Object Notation* [7] e faz a composição. O JSON foi escolhido por apresentar facilidades na manipulação de dados, sendo muito utilizado principalmente para tráfego de informações em ambientes heterogêneos via HTTP e possuir implementações para mais de 20 linguagens, dentre elas o *Javascript*.

5.5 ESTUDO DE CASO

Os Recursos de Informação podem ser tratados de forma mais específica, quando voltados ao ensino e capacitação de profissionais da saúde. Neste escopo eles são categorizados como **Objetos de Aprendizagem** de acordo com a **Definição 5** a seguir:

Definição 5 (Objeto de Aprendizagem - OA): um Objeto de Aprendizagem é qualquer material que pode ser usado no processo de ensino e aprendizado (Hodgins 2002). Quando em formato digital um OA pode ser organizado na forma de material educativo a ser usado na formulação de cursos suportados por tecnologia ou Ensino a Distância - EaD.

[7] http://www.json.org

6 EXPERIMENTOS

A realização de experimentos compara o desempenho do mecanismo de busca implementado, com soluções desenvolvidas em trabalhos relacionados, especificamente buscas com índices meramente léxicos e utilizando SA convencional. Para isso, a proposta de SA adaptado foi implementada e utilizada com um repositório de conteúdo. Os objetos de informação e a ontologia utilizados nesses experimentos referem-se a um estudo de caso na área de saúde, desenvolvido junto à UnA-SUS. Infelizmente, não foi possível realizar experimentos empíricos devido à falta de objetos devidamente anotados no atual estágio de evolução do programa UnA-SUS. Assim, focamos este trabalho na implementação do primeiro protótipo dos mecanismos de buscas e na avaliação do seu desempenho computacional, com diferentes configurações de parâmetros. Este descreve a implementação, o plano de experimentos, a sua realização e relata os resultados obtidos.

6.1 DESCRIÇÃO DOS EXPERIMENTOS

Os experimentos de recuperação de informação foram realizados sobre o Protótipo do SA desenvolvido, visando aferir o desempenho do algoritmo de *Spreading Activation*. Estes experimentos com o protótipo também visam valores de parâmetros do SA apropriados para uma execução mais eficaz do método de busca no repositório UnA-SUS da UFSC.

6.1.1 AMBIENTE DE REALIZAÇÃO DOS EXPERIMENTOS

Todos os trabalhos de implementação e experimentos foram realizados em máquinas do Laboratório para Integração de Sistemas e Aplicações Avançadas - LISA. A configuração das máquinas é a seguinte:

- **processador:** Intel Core 2 Duo E6550 @ 2.33 GHz
- **memória principal (RAM):** 4,00 GB

- **sistema operacional:** Microsoft Windows 7 Professional

O código do protótipo foi instrumentado e executado na *IDE Eclipse3.5* [1], sob a plataforma *Java 6*.

6.1.2 OBJETIVOS

Os objetivos dos experimentos com o Protótipo de SA desenvolvido são:

1. a partir de variações nos valores dos parâmetros de configuração do SA, avaliar qual o **impacto** de tais parâmetros no desempenho do SA e na qualidade dos resultados das buscas.
2. a partir de variações nos valores dos parâmetros de configuração do SA, descobrir possíveis **correlações** que ocorram entre os parâmetros de configuração do SA.
3. medir a **quantidade** e a relevância semântica dos OAs obtidos em cada consulta.
4. detectar possíveis **erros** de execução no protótipo.

Sob a luz destes objetivos e a partir de consultas pré-definidas executadas com o algoritmo de SA, comparações entre as diversas configurações de parâmetros do SA são feitas.

6.1.3 MÉTRICAS

Para cumprir cada um dos objetivos traçados é necessária a captura de métricas durante a execução dos experimentos:

- **tempo de execução**: medida que define quanto tempo o SA levou para executar uma consulta, em milissegundos.
- **consumo de CPU**: define o percentual de processador exigido para a execução do SA.
- **quantidade de OAs obtidos**: determina a quantidade de OAs obtidos em cada consulta. Esta métrica afere de maneira simples a cobertura do SA de acordo com os valores dos parâmetros, pois estes parâmetros interferem na quantidade de nodos disparados e também no cálculo dos valores de ativação de cada nodo.

[1] http://www.eclipse.org/

- **relevância semântica dos OAs obtidos**: define valor de relevância semântica dos Objetos de Aprendizagem obtidos pelo SA.

- **percentual de acerto dos OAs obtidos**: esta medida é definida no trabalho de (Baeza-Yates e Ribeiro-Neto 1999). Em nosso trabalho, cada consulta q_i visa recuperar um conjunto de OAs $E_{qi} \subset R$. Este conjunto E_{qi} corresponde aos OAs que empiricamente são esperados de se obter após uma consulta com q_i. Com isto foi possível aferir o percentual de acerto da busca realizada com q_i em relação a E_{qi}. Esta métrica foi comparada entre as 4 categorias do SA e também na busca sintática nativa do DSpace.

6.2 DADOS DOS EXPERIMENTOS

Foram definidos dois tipos de experimentos: i) experimentos com uma porção pequena do DeCS e OAs reais anotados na RS e ii) experimentos com uma grande porção do DeCS e OAs sintéticos com anotações geradas aleatoriamente, de acordo com alguma distribuição de frequência. Esta divisão justifica-se pela dificuldade de se obter uma grande quantidade de anotações reais no repositório oficial.

6.2.1 E1 - EXPERIMENTOS SIMPLES

Para a recuperação de OAs é necessária a catalogação destes objetos com termos do DeCS. O repositório UnA-SUS já apresenta OAs catalogados, contudo estes OAs não foram catalogados corretamente. Esta catalogação inicial foi feita pelos desenvolvedores de OAs, que ainda não possuíam o conhecimento necessário para fazer a catalogação com termos do vocabulário controlado. Assim, foi necessário efetuar anotações desses objetos com termos do DeCS da seguinte maneira: o conteúdo de cada OA foi estudado e o conjunto de palavras-chave foi obtido destes conteúdos, então consultou-se o DeCS para se obter os termos necessários à catalogação destes Objetos. O conjunto de objetos de aprendizagem inseridos no repositório de recursos de informação e as anotações desses objetos com um subconjunto de termos do DeCS são apresentados na **Figura 31**.

Recurso	Título	Anotados com
r1	Arritmias Cardíacas	k01, k02, k03
r2	Hipertensão Arterial	k01, k04, k05, k06, k07
r3	Doenças do Coração - Miocardiopatias	k01, k08, k09, k10
r4	Insuficência Cardíaca	k01, k11, k12
r5	Acidente Vascular Cerebral	k05, k13, k14, k15, k16
r6	Infecções do Cérebro	k15, k17, k18, k19, k20
r7	Lesões causadas pelo desporto	k41, k42, k23, k43
r8	Miopatia miotônica	k21, k22, k23, k24, k25
r9	Paralisia Periódica	k49, k22, k23,
r10	Osteoporose	k26. k27
r11	Artrose	k31, k30, k29, k28, k27
r12	Bronquite	k32, k33, k34, k35, k36
r13	Doenças Alérgicas dos Pulmões	k44, k34, k33, k45
r14	Pneumonias	k46, k47,k48, k34
r15	Fibrose quística	k34, k37, k38, k39
r16	Câncer do Pulmão	k34, k40
r17	Embolia pulmonar	k34, k50, k13, k05
r18	Tumores ósseos	k27, k26, k30, k28
r19	Síndrome de insuficiência respiratória aguda	k33, k34, k12, k40
r20	Delírio e demência	k51, k15, k05

Conjunto de termos	
id.K	Termo DeCS
k01	Coração
k02	Valvas Cardiacas
k03	Arritmias Cardíacas
k04	Doenças Vasculares
k05	Acidente Cerebral Vascular
k06	Sistema Nervoso Central
k07	Síndrome Neurologica de Alta Pressao
k08	Isquemia Miocardica
k09	Ventriculos do Coracao
k10	Miocardio
k11	Sistema de Conducao Cardiaco
k12	Baixo Debito Cardíaco
k13	Doencas Vasculares
k14	Doencas Arteriais Intracranianas
k15	Cerebro
k16	Arterias
k17	Infeccoes do Sistema Nervoso Central
k18	Meningite
k19	Encefalite Viral
k20	Medula Espinhal
k21	Transtornos Miotonicos
k22	Doencas Musculares
k23	Musculos
k24	Espasticidade Muscular
k25	Síndromes da Dor Miofascial
k26	Doencas osseas
k27	Osso e Ossos
k28	Artropatias
k29	Doencas das Cartilagens
k30	Articulacoes
k31	Cartilagem Articular
k32	Bronquios
k33	Bronquite
k34	Pulmao
k35	Bronquiolite Viral
k36	Pneumonia Bacteriana
k37	Fibrose Cística
k38	Doencas do Sistema Digestorio
k39	Doencas Congenitas e Hereditarias
k40	Doencas Pulmonares Intersticiais
k41	Tendoes
k42	Ferimentos e Lesões
k43	Cotovelo de Tenista
k44	Pneumopatias
k45	Hipersensibilidade Respiratoria
k46	Pneumonia
k47	Pneumonia Bacteriana
k48	Pneumonia Viral
k49	Paralisias Periodicas Familiares
k50	Embolia Pulmonar
k51	Transtornos Relacionadosao Uso de Substâncias

Figura 31: Conjunto de OAs e anotações dos experimentos.

A Rede Semântica (RS) dos experimentos foi gerada com os objetos e anotações apresentados na **Figura 31**. Todos os termos usados nas anotações desses objetos estão restritos a de 3 categorias do DeCS: *Anatomia*, *Organismos* e *Doenças*. Assim, foram efetuados recortes de termos e relações semânticas destas 3 categoria para compor a rede semântica utilizada nos testes. Estas categorias foram escolhidas por apresentarem muitos inter-relacionamentos entre seus termos e serem de fácil compreensão. A geração de uma RS a partir de recortes do DeCS

facilita a visualização da execução do SA. Os nodos que compõem esta RS compreendem os 3 primeiros níveis das 3 categorias supracitadas, de maneira a abarcar todos os termos mais gerais e também termos mais específicos, os quais são usados para anotar os recursos. A RS considerada nestes experimentos apresenta em torno de 1200 nodos.

Feita a estruturação da RS é necessária a definição de um conjunto S de valores de parâmetros que foi considerado durante os testes com o algoritmo de SA. Os parâmetros são os seguintes:

- **threshold**: este parâmetro determina se um nodo pode ser ativado em um pulso do SA.
- **generalização**:valor de peso associado a direção de um *link* do nodo. O cálculo do valor de ativação leva em conta este valor quando o nodo ativado está generalizando o nodo anterior.
- **especialização**: valor de peso associado a direção de um *link* do nodo. O cálculo do valor de ativação leva em conta este valor quando o nodo ativado está especializando o nodo anterior.
- **anônima**: valor de peso relativo a relações entre dois nodos que representam termos do DeCS.
- **fator de decaída**: valor que representa a perda de força do SA a medida em que os pulsos se afastam da semente.
- **máximo de nodos por ciclo**: quantidade máxima de nodos ativados em um pulso do SA.

Agrupamos as configurações de parâmetros em quatro categorias: *restrita, média, aberta* e *muito aberta*. A justificativa de escolha dos nomes está baseada na natureza de cada uma destas categorias, de maneira que uma configuração restrita faça com que o SA execute poucos pulsos, tenha limiar de ativação alto e decaimento rápido. Esta idéia se estende às outras 3 categorias. A tabela a seguir apresenta os valores definidos em cada uma destas categorias:

Categoria	Id.	*Threshold*	Generalização	Especialização	Anônima	Fator Decaída	Máx.Nodos por Ciclo
Restrita	S1	0.7	0.8	0.85	0.85	0.8	15
Média	S2	0.5	0.85	0.9	0.9	0.85	20
Aberta	S3	0.45	0.9	0.95	0.95	0.9	40
Mais Aberta	S4	0.4	0.95	1.0	1.0	0.95	60

Figura 32: Categorias de configuração dos parâmetros do SA.

Além destes parâmetros variáveis foram definidos valores constantes, elencados a seguir:

- **sinônimo**: esta relação recebeu peso = 0.9, pois termos sinônimos apresentam correspondência semântica no DeCS.
- **anotação**: esta relação recebeu peso=1.0, de maneira que o valor de ativação de um OA diminua muito pouco em relação a um termo que o anota.

6.2.1.1 CONSULTAS DE E1

A execução do SA no protótipo requer um conjunto de classes de consultas Q. Este conjunto foi definido em função da quantidade e da natureza (sinônimos, termos genéricos, termos especializados) dos termos, gerando assim 6 classes de consultas:

- **c1 - 1 termo**: consulta com 1 termo que anotou um dos OAs;
- **c2 - 2 termos**: consulta com 2 termos que anotaram um dos OAs;
- **c3 - 3 termos**: consulta com 3 termos que anotaram um dos OAs;
- **c4 - 1 sinônimo**: consulta com 1 termo sinônimo de um determinado termo que anota um dos OAs;
- **c5 - termo especializado**: consulta com 1 termo mais especializado em relação a um determinado termo que anota um dos OAs;
- **c6 - termo mais geral**: consulta com 1 termo mais geral em relação a um determinado termo que anota um dos OAs;

São consideradas 6 consultas para cada um dos 20 OAs, sendo que cada uma delas diz respeito a uma das 6 categorias definida previamente ($q1$, ..., $q6$). Por exemplo, para o OA $r2$ que versa sobre "Hipertensão Arterial" temos as seguintes consultas:

- q1 = (Coração).
- q2 = (Doenças Vasculares, Acidente Cerebral Vascular).
- q3 = (Doenças Vasculares, Acidente Cerebral Vascular, Coração).
- q4 = (AVC).
- q5 = (Doenças do Sistema Nervoso Central).
- q6 = (Encéfalo).

Outros termos são utilizados para gerar consultas para recuperar outros objetos da base, escolhendo palavras-chaves apropriadas cada objeto em cada classe de consulta $c1$ a $c6$. Sabe-se que cada consulta possui um conjunto de OAs esperados E_{qi}. Para as 6 consultas supracitadas os OAs esperados são os seguintes:

- E_{q1} = (r2).
- E_{q2} = (r2, r5, r17, r20).
- E_{q3} = (r2, r5, r6, r17, r20).
- E_{q4} = (r2).
- E_{q5} = (r2, r5, r6, r17, r20)
- E_{q6} = (r5, r6, r20).

Cada uma das 6 classes de consultas foi adaptada para encontrar cada um dos 20 OAs e executada com cada uma das 4 configurações de parâmetros (restrita, média, aberta, muito aberta) de parâmetros do SA. Em vista disso, 480 é o número de vezes que o algoritmo de SA foi executado já que temos:

- 20 OAs (R);
- 6 consultas por OA (Q);
- 4 configurações de parâmetros por consulta (S).

6.2.2 E2 - EXPERIMENTOS COM DADOS SINTÉTICOS

A realização de experimentos com grande quantidade de nodos e anotações na RS visa avaliar o desempenho e a escalabilidade do SA, com variações dos seus parâmetros. Para isso, a RS foi gerada com uma porção do DeCS chamada *SubDeCS* através da composição das categorias Anatomia, Doenças e Organismo, totalizando 9694 nodos, aproximadamente 30% do total de nodos do DeCS.

As anotações foram geradas aleatoriamente utilizando a ferramenta *DBGen* [2], que produziu 10000 números distribuídos no intervalo [1,10000] segundo a distribuição Gaussiana (normal) com média em 4798 (valor médio da quantidade de termos de *SubDeCS*). Esta distribuição

[2] http://www.gbdi.icmc.usp.br/download

determinou a quantidade de anotações que cada OA vai recebeu, situadas no intervalo de [1,7] anotações. Assim foram criadas 7 partições nesta distribuição e cada OA recebeu um número aleatório de anotações de acordo com a partição em que ele está situado. A **Figura 33** apresenta a distribuição das anotações sobre os 10000 OAs. Por exemplo, um OA *i* situado em col_2 receberá anotações de 2 termos aleatórios de *SubDeCS*.

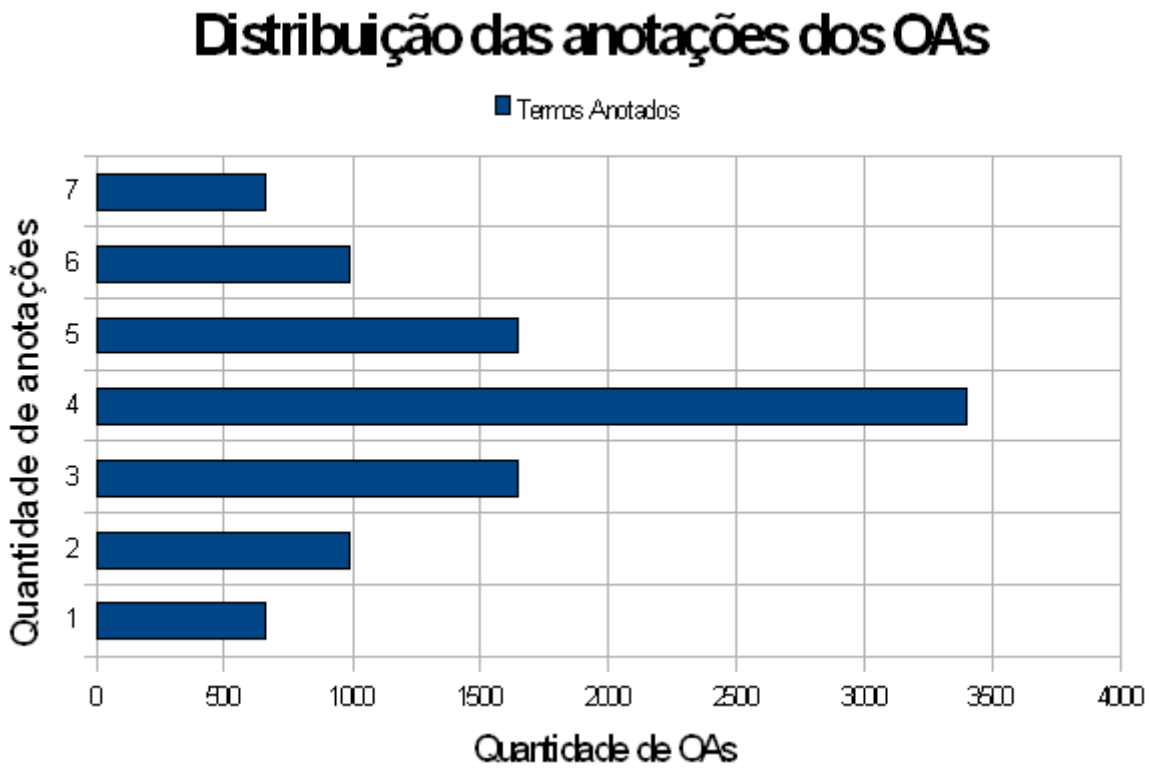


Figura 33: Distribuição das anotações dos OAs de E2.

Como mostrado na **Figura 33**, os OAs foram particionados em 7 categorias Col_i de acordo com a distribuição Normal e as anotações foram realizadas da seguinte maneira:

- col_1: os 660 OAs situados no intervalo [0,660] receberam anotação com 1 termo;
- col_2: os 990 OAs situados no intervalo [661,1650] receberam 2 anotações;
- col_3: os 1650 OAs situados no intervalo [1651,3300] receberam 3 anotações;
- col_4: os 3400 OAs situados no intervalo [3301,6700] receberam 4 anotações;
- col_5: os 1650 OAs situados no intervalo [6701,8350] receberam 5 anotações;
- col_6: os 990 OAs situados no intervalo [8351,9340] receberam 6 anotações;

- col_7: os 660 OAs situados no intervalo [9341,10000] receberam 7 anotações;

Para a coleta das métricas procedeu-se da seguinte maneira:

1. selecionou-se um parâmetro *p* dentre os parâmetros avaliados (número máximo de nodos por ciclo, fator de decaída e *threshold*). Ao parâmetro *p* atribuiu-se um valor inicial mínimo (0,05 para fator de decaída e *threshold* e 5 para número máximo de nodos por ciclo);

2. variou-se este parâmetro *p* com um passo pequeno (0,05 ao *threshold* ou fator de decaída e 5 para número máximo de nodos por ciclo). Os demais parâmetros receberam um valor médio fixo (0,5 para fator de decaída e *threshold* e 50 para número máximo de nodos por ciclo); ;

3. a cada novo valor de *p* foram executadas 2 consultas no SA: i) consulta com 15 sementes aleatórias e ii) consulta i) acrescida de 5 sementes com termos mais gerais (Anatomia, Doenças, Organimos, Sistema Nervoso e Sistema Cardiovascular);

Assim foram feitas 20 iterações para cada um dos parâmetros *p*, avaliando o consumo de CPU e a quantidade de nodos obtidas em cada execução do SA.

6.3 ANÁLISE DOS RESULTADOS

6.3.1 ANÁLISE QUANTITATIVA

6.3.1.1 E1 - EXPERIMENTOS COM OBJETOS REAIS ANOTADOS MANUALMENTE

A análise quantitativa avalia métricas colhidas pela instrumentação do código (tempo de execução, uso de CPU, quantidade de OAs e relevância semântica. Para colher a métrica de consumo de CPU foi utiliza a ferramenta *YourKit Java Profiler 9.5* [3] integrada à IDE $Eclipse3.5$.

O gráfico da **Figura 34** apresenta o **tempo médio de execução** do SA com cada categoria de parâmetros $s_i \in S$. A categoria que consumiu mais tempo executando o SA foi a categoria $s3$ *(aberto)*, seguida de perto pela categoria $s4$ *(muito aberto)* e a categoria que menos consumiu tempo executando foi a categoria $s1$ *(restrito)*. Esperava-se que a categoria $s4$ *(muito aberto)* apresentasse o maior consumo de tempo de execução.

[3] http://www.yourkit.com/

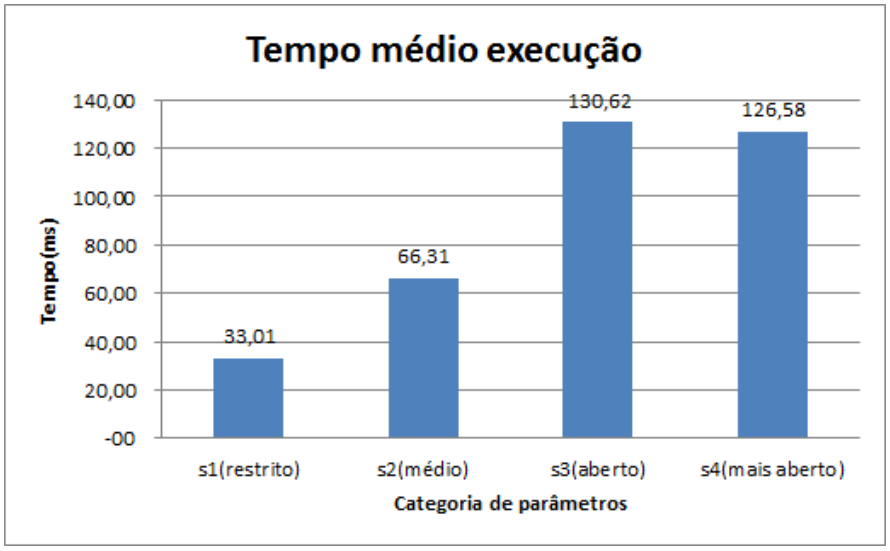


Figura 34: Comparação do tempo médio de execução entre as categorias de parâmetros.

A **Figura 35** mostra o consumo médio de CPU (em %) para cada categoria de configuração de parâmetros. Esta métrica é complementar a métrica de tempo de consumo de CPU, porém os resultados não foram análogos. A categoria que apresentou o maior consumo médio de CPU foi *s*2 *(média)* e as demais categorias apresentaram consumos muito semelhantes.

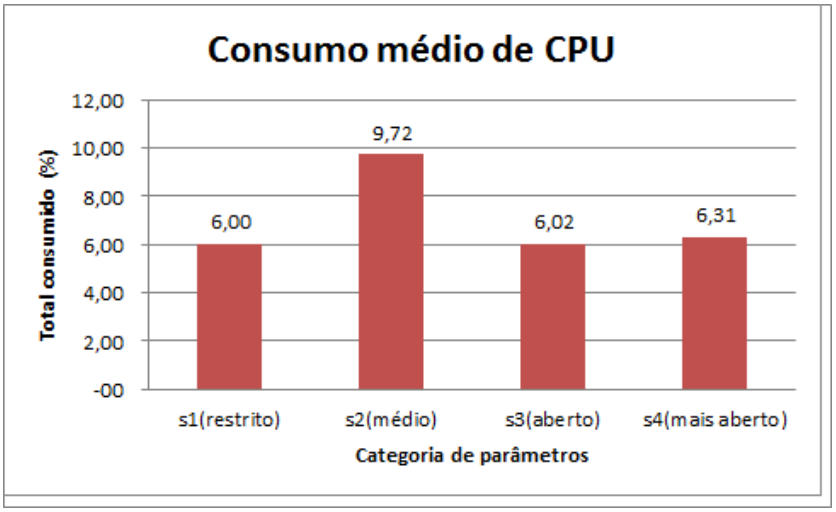


Figura 35: Comparação do consumo médio de CPU entre as categorias de parâmetros.

A quantidade média de OAs obtidos para cada categoria de parâmetros é apresentada na **Figura 36**. Os resultados foram de acordo com o esperado: as categorias que proporcionavam uma execução do SA de forma mais abrangente (*s*3 e *s*4) alcançaram mais OAs.

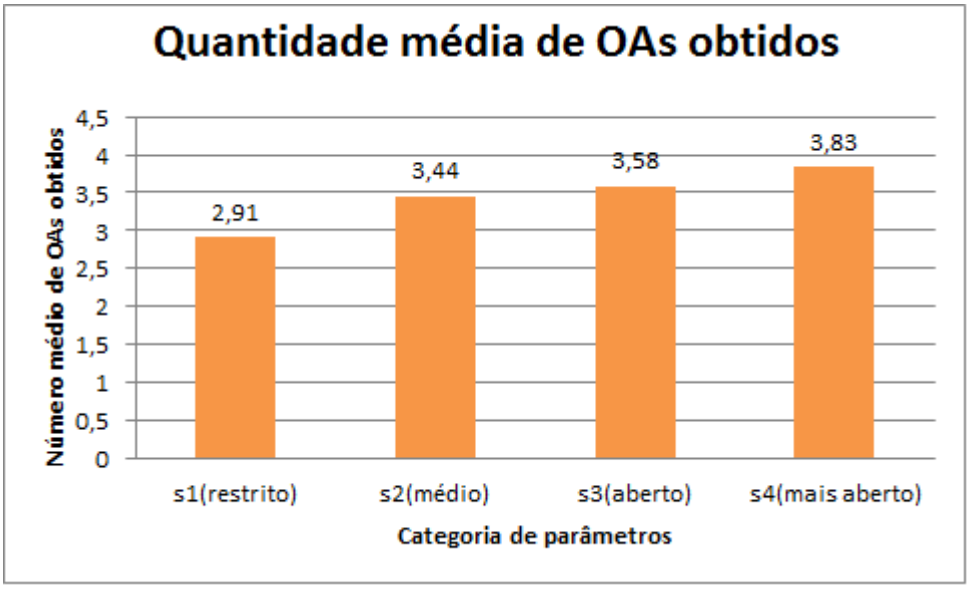


Figura 36: Comparação dos OAs recuperados entre os conjuntos de parâmetros.

Outra medida aferida foi o **tempo médio de execução** para cada uma das 6 classes de consultas $q_i \in Q$, apresentado na **37**. O objetivo foi determinar o desempenho da execução das diferentes classes de consulta. Observou-se que a classe de consulta $q6$ (termo mais geral) destaca-se das demais classes em uma ordem de grandeza, comprovando o fato de que nodos com grau muito elevado consomem muitos recursos da máquina que está executando o SA. Nodos genéricos possuem muitas ligações a outros nodos, consequetemente provocam o espalhamento da ativação do SA em uma área muito grande da RS. A restrição de fan-out tenta evitar isso. Porém não se pode garantir que consultas com termos mais gerais serão pouco submetidas por usuários. Assim surgiram propostas de trabalhos futuros para atenuar este problema:

- atualização dos parâmetros do SA em tempo de execução: quando o SA expandir um nodo com muita associatividade ele pode diminuir o espalhamento do algoritmo, aumentando o valor de *threshold*, por exemplo;
- interface sugestiva: uma interface que sugira aos usuário termos mais específicos como sementes do SA em relação às sementes submetidas. Com isso o obtém-se um processamento mais rápido e respostas mais curtas e específicas.

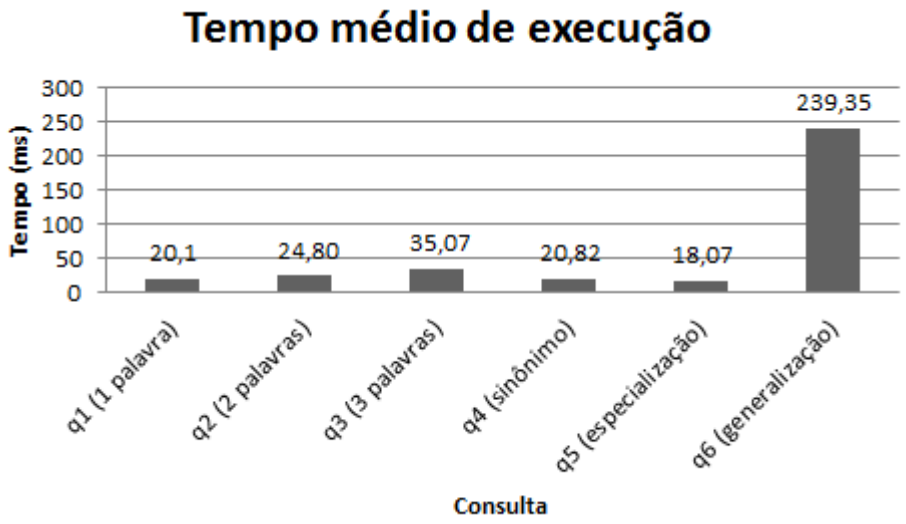


Figura 37: Comparação do tempo de execução entre as classes de consultas.

6.3.1.2 E2 - EXPERIMENTOS COM DADOS SINTÉTICOS

A **Figura 38** apresenta a comparação do consumo de CPU de acordo com a variação dos parâmetros *p* do SA avaliados. Similarmente às medições de E1, o consumo de CPU aumentou de acordo com a diminuição do *threshold* ou o aumento do fator de decaída. Esta medida confirmou a relação inversa entre ***threshold*** e **fator de decaída** observada inicialmente em E1. O **número máximo de nodos por ciclo** não apresentou relevância no SA, pois sua variação não afetou o consumo de CPU do SA.

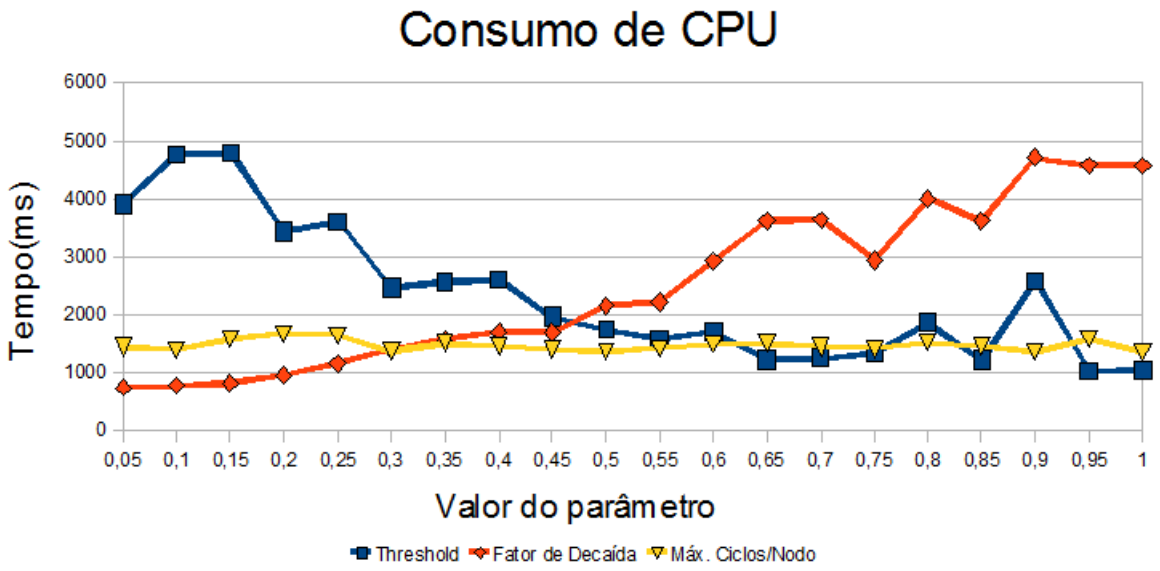


Figura 38: Comparação do tempo de execução entre os parâmetros do SA avaliados.

Na **Figura 39** aferiu-se a quantidade média de OAs recuperados em cada execução do SA de acordo com a variação dos parâmetros p do SA avaliados. Esta medida apresentou comportamento análogo ao consumo de CPU, ou seja, a quantidade de OAs obtidas eleva-se com a diminuição do valor do ***threshold*** ou o aumento do **fator de decaída**. Novamente o **número máximo de nodos por ciclo** não impactou nos resultados, pois a quantidade de OAs obtidas permaneceu em um intervalo muito restrito.

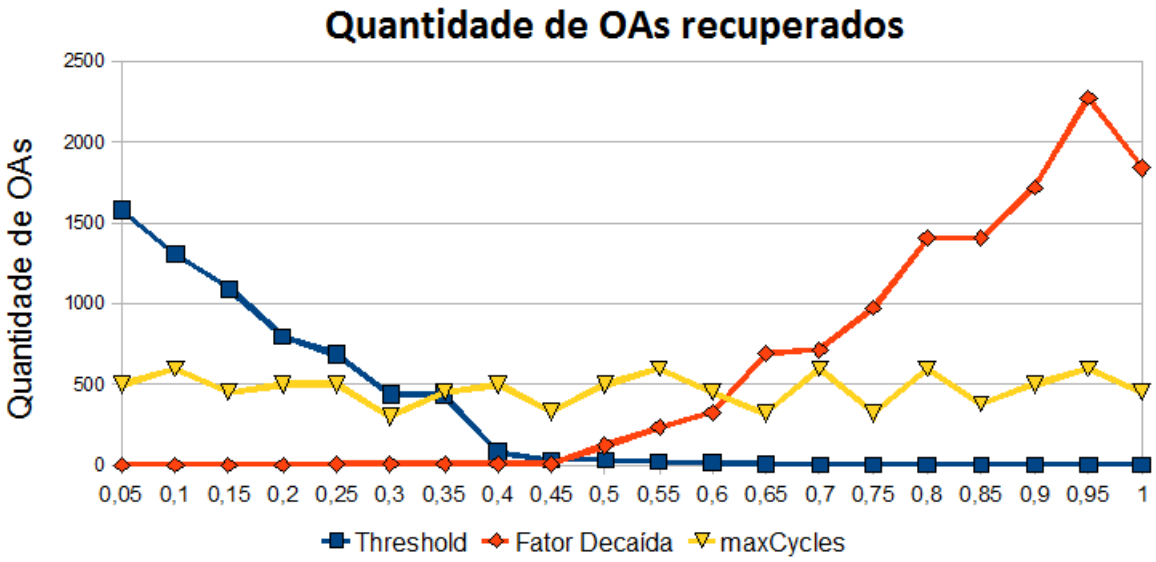


Figura 39: Comparação dos OAs recuperados entre os parâmetros do SA avaliados

6.3.2 ANÁLISE QUALITATIVA

O percentual **de OAs corretamente recuperados** foi aferido de duas maneiras:

1. em relação à **cobertura**: para uma consulta qi foi determinada a relação entre o número de OAs corretos obtidos (r_{qi}) e o número de OAs esperados para a esta consulta ($|E_{qi}|$).

$$cobertura = \frac{r_{qi}}{|E_{qi}|}$$

onde:

- r_{qi} é a quantidade de OAs esperados que são **efetivamente** recuperados pela consulta qi.
- $|E_{qi}|$ é a quantidade de OAs que se espera recuperar com a consulta qi.

2. em relação à **precisão**: esta medida determinou-se pela relação entre o total de OAs obtidos (t_{qi}) e o número de OAs corretos recuperados (r_{qi}), desta maneira:

$$precisão = \frac{r_{qi}}{|t_{qi}|}$$

onde:

- r_{qi} é a quantidade de OAs esperados que são **efetivamente** recuperados pela consulta qi.
- $|t_{qi}|$ é a quantidade total de OAs recuperados com a consulta qi.

A categoria com maior cobertura foi s_4, seguida de s_3, s_2 e s_1 e por fim temos o mecanismo sintático, que apresentou cobertura muito inferior. Quanto à precisão aconteceu exatamente o oposto, pois a categoria com maior precisão aferida foi o mecanismo sintático, seguido por s_1, s_2, s_3 e finalmente s_4. Estes resultados estão de acordo com o esperado, pois as categorias apresentaram cobertura e precisão correlacionadas com sua natureza de configuração de parâmetros. Já o mecasnimo sintático apresenta cobertura nulo com consultas que não apresentem *matching* exato entre as palavras-chave das consultas e os termos que catalogaram os OAs. Dessa maneira as consultas $q4$ (1 sinônimo), $q5$ (generalização) e $q6$ (especialização) nunca retornam Objetos de Aprendizagem. Entretanto o mecanismo sintático apresenta precisão máxima, pois recupera somente o que foi esperado.

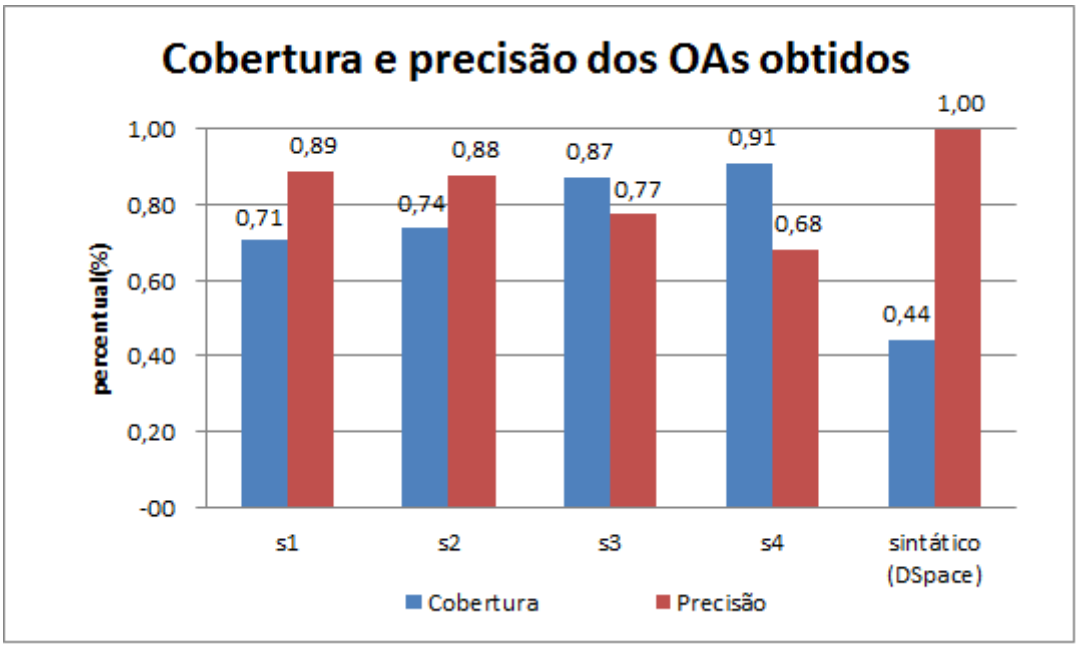


Figura 40: Comparação de cobertura e precisão de OAs recuperados de acordo com as categorias de parâmetros e com a busca sintática do DSpace.

6.4 DISCUSSÃO

Os experimentos permitiram focaram na comparação de desempenho do SA configurado com valores de parâmetros distintos. Observou-se que o consumo de CPU foi maior na categoria s_2 (*média*), de maneira não esperada. As demais categorias apresentaram consumo de CPU muito similares, de tal maneira que não se consegue encontrar explicações para o consumo elevado de s_2. Entretanto a medida tempo médio de execução do SA já apresentou valores dentro do esperado, de maneira que este tempo de execução aumentava de acordo com o aumento da abertura "tolerância"de cada categoria, numa ordem crescente de grandeza começando em s_1 (*restrita*) e terminando em s_4 (*mais aberta*).

Em relação à quantidade de Objetos de Aprendizagem recuperados em cada uma das consultas o comportamento observado também foi o esperado: consultas mais restritas (s_1 e s_2) apresentaram número menor de OAs obtidos que consultas mais abertas (s_3 e s_4). Contudo medir apenas a quantidade de OAs recuperados não é o suficiente, com isso partiu-se para uma análise qualitativa destes OAs recuperados. Definiu-se a **cobertura** e **precisão** média de cada uma das categorias e observou-se uma relação inversamente proporcional entre **cobertura** e **precisão**, como por exemplo na categoria s_1 (*restrita*), que apresentou a menor **cobertura** e a maior **precisão** dos OAs obtidos. Este fato é explicado pela natureza desta categoria, que expande menos nodos (menor cobertura) mas recupera apenas OAs que são esperados (maior precisão). Uma configuração extrema do SA pode degenerá-lo para uma busca sintática, ou seja, apenas o nodo semente é executado, nenhum adjacente é expandido e assim esta semente tem a mesma função de uma palavra-chave da busca sintática.

Com estes experimentos podemos realizar as seguintes análises:

1. **relação entre *threshold* e tempo de execução**: em E1 o gráfico **tempo médio de execução** apresenta fortes indícios de uma relação inversamente proporcional entre *threshold* e tempo de execução, pois um *threshold* menor torna o SA mais "tolerante" e este então passa a considerar mais nodos ativos. Consequentemente tais nodos podem disparar os nodos adjacentes, expandindo ainda mais o grafo, aumentando assim o tempo de execução. O fator de decaída tem uma relação diretamente proporcional com esta métrica apresentada, pois valores mais altos no fator de decaída fazem com que as ondas de ativação enfraqueçam a cada pulso, assim se espalham menos pelo grafo. Estas observações foram comprovadas com o experimento E2, pois este apresentou execuções do SA em uma RS mais ampla.

2. **relação entre *threshold* e consumo de CPU**: durante a execução dos experimentos foi

possível acompanhar o consumo de CPU através da ferramenta *YourKit Java* e verificou-se uma relação inversamente proporcional entre uso de CPU e *threshold*. Quanto menor o valor de *threshold*, maior a tolerância do SA e com isso mais nodos são ativados, aumentando assim o consumo de CPU.

3. **relação entre *associatividade* de um nodo e consumo de CPU**: em consultas que envolvam nodos com *associatividade* elevada o consumo de CPU é maior, ou seja, consultas com termos mais gerais da RS consomem mais CPU.

4. **degeneração do SA pelo *threshold***: valores de *threshold* muitos baixos degeneram o SA quando este ativa nodos que possuem associatividade alta. Concluímos que o *threshold* é um fator crítico para o desempenho do SA principalmente em consultas que apresentem termos mais gerais. Este fato agrava-se em E2, pois a RS maior apresenta nodos com mais adjacentes, podendo ocorrer ainda mais degeneração do SA em consultas com estes nodos.

5. **número de nodos ativados**: a quantidade de nodos ativada em um pulso do SA depende diretamente do *threshold* e do **fator de decaída**. Quanto menor o *threshold* mais nodos são ativados e quanto maior o fator de decaída mais lenta é a diminuição da força da propagação

6. **escolha de categoria de parâmetros**: o experimento E1 sugere a categoria $s1$ (default) como a categoria com o maior equilíbrio entre consumo de recursos (tempo e CPU) e resultados (OAs obtidos). Este fato justifica a escolha do nome desta categoria $s1$ ($default$). Ela foi o primeira a ser elaborada, em decorrência da experiência e intuição adquiridas durante o processo de familiarização e implementação do *Spreading Activation*.

7. **acerto com buscas semânticas**: em E1 observou-se um incremento substancial na quantidade de OAs recuperados via o mecanismo de SA em relação à busca sintática. O SA consegue obter resultados valendo-se de consultas formuladas com palavras-chaves que não estão diretamente ligadas às palavras-chave que anotam os OAs, permitindo assim o uso de sinônimos e termos mais gerais ou especializados na elaboração das consultas. O experimento E2 mostrou o comportamento do SA perante uma RS grande e com a carga de valores de parâmetros distintos.

7 CONCLUSÕES

Este trabalho apresentou uma implementação de um protótipo de um mecanismo de buscas semânticas com o algoritmo de *Spreading Activation*(SA). Para implementação do SA foi necessário primeiramente se definir formalmente um estrutura de Rede Semântica e então adaptar o SA. Os experimentos realizados para testar o protótipo desenvolvido, mostraram ganhos de cobertura com o uso do mecanismo de busca proposto, com baixo comprometimento da precisão.

7.1 OBJETIVOS CUMPRIDOS

Os objetivos traçados para este trabalho foram:

1. a partir de variações nos valores dos parâmetros de configuração do SA, avaliar qual o **impacto** de tais parâmetros no desempenho do SA e na qualidade dos resultados das buscas.
2. a partir de variações nos valores dos parâmetros de configuração do SA, descobrir possíveis **correlações** que ocorram entre os parâmetros de configuração do SA.
3. medir a **quantidade** e a relevância semântica dos OAs obtidos em cada consulta.
4. detectar possíveis **erros** de execução no Protótipo.

Os objetivos **1)** e **2)** foram cumpridos parcialmente através da definição das categorias de configuração de parâmetros. Os gráficos e análises geradas e discutidas apresentam o desempenho quantitativo e qualitativo do SA, além de observações realizadas de correlações entre os parâmetros.

O objetivo **3)** foi cumprido através de uma análise qualitativa experimentos. Esta análise deu-se com coleta da quantidade de OAs recuperados em cada categoria de parâmetros de SA e também na busca sintática do DSpace.

O cumprimento do objetivo **4)** deu-se durante a execução dos experimentos, com o auxílio do *console* de *IDE Eclipse 3.5*, que apresenta as mensagens dos possíveis erros na implementação, facilitando a depuração.

7.2 TRABALHOS FUTUROS

O trabalho também apresentou outras idéias promissoras a serem exploradas em trabalhos futuros:

1. Refinar a implementação, de maneira a otimizar as configurações dos parâmetros de acordo com o histórico de consultas realizadas pelos usuários, definido assim perfis de usuários.

2. Realizar testes com o mecanismo de buscas compostas no DSpace com usuários especialistas, de maneira a determinar um valor ideal para o fator de relevância α.

3. Adicionar outros vocabulários controlados que complementem o DeCS, de maneira a aumentar o potencial do SA.

4. Realizar testes com muitos usuários em rede, para aferir a escalabilidade do SA executando um grande volume de consultas simultâneas.

5. Utilizar algoritmos de detecção de relações semânticas entre termos do vocabulário e os Objetos de Aprendizagem, para gerar anotações de OAs.

6. Refazer tais testes sobre o repositório completo usando o DeCS inteiro para criar a RS.

REFERÊNCIAS

Androutsopoulos, Tsatsaronis e Vazirgiannis 2007 ANDROUTSOPOULOS, I.; TSATSARONIS, G.; VAZIRGIANNIS, M. Word sense disambiguation with spreading activation networks generated from thesauri. In: *Proceedings of the 20th international joint conference on Artifical intelligence*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc, 2007. p. 1725–1730.

Aragao M. P.; Rocha e Schwabe 2004 ARAGAO M. P.; ROCHA, C.; SCHWABE, D. A hybrid approach for searching in the semantic web. *Proceedings of the 13th international conference on World Wide Web*, 2004.

Aswath D.; Ahmed e Davulcu 2005 ASWATH D.; AHMED, S. T. D. J.; DAVULCU, H. Boosting item keyword searching with spreading activation. *Proceedings of the 2005 IEEE/WIC/ACM International Conference on Web Intelligence*, p. 704–707, 2005.

Baeza-Yates e Ribeiro-Neto 1999 BAEZA-YATES, R.; RIBEIRO-NETO, B. *Modern Information Retrieval*. [S.l.]: New York : ACM Press, 1999.

Bass et al. 2003 BASS, M. et al. The dspace institutional digital repository system: current functionality. In: *Proceedings of the 3rd ACM/IEEE-CS joint conference on Digital libraries*. [S.l.]: IEEE Computer Society, 2003. (JCDL '03), p. 87–97.

Berners-Lee T.; Hendler e Lassila 2001 BERNERS-LEE T.; HENDLER, J.; LASSILA, O. The semantic web. *Scientific American*, v. 284, n. 5, p. 34–43, May 2001.

Breitman, Casanova e Truszkowski 2007 BREITMAN, K.; CASANOVA, M.; TRUSZKOWSKI, W. *Semantic Web: Concepts, Technologies and Applications*. [S.l.]: Springer, 2007. (NASA Monographs in Systems and Software Engineering).

Chang E; Hai Dong; Hussain 2008 CHANG E; HAI DONG; HUSSAIN, F. A survey in semantic search technologies. *2nd IEEE International Conference on Digital Ecosystems and Technologies*, p. 403–408, 2008.

Ciravegna et al. 2005 CIRAVEGNA, F. et al. Semantic annotation for knowledge management: Requirements and a survey of the state of the art. *Journal of Web Semantics*, v. 4(1), p. 14–28, 2005.

Collins e Quillian 1968 COLLINS, A. M.; QUILLIAN, M. Semantic memory. *Journal of Verbal Learning and Verbal Behavior,*, p. 227–270, 1968.

Crestani 1997 CRESTANI, F. Application of spreading activation techniques in information retrieval. *Artificial Intelligence Review*, v. 11, p. 453–482, 1997.

Crestani e Rijsbergen 1993 CRESTANI, F.; RIJSBERGEN, C. Modeling adaptive information retrieval. *Journal of Intelligent Information Systems*, v. 8, p. 29–56, 1993.

D'Agostini C.S.; Fileto e Gauthier 2008 D'AGOSTINI C.S.; FILETO, R.; GAUTHIER, F. A. Contextual semantic search - capturing, using the user's context to direct semantic search. *10th International Conference on Enterprise Information Systems (ICEIS), Barcelona, Spain*, SAIC, p. 154–159, 2008.

Davies, Studer e Warren 2006 DAVIES, J.; STUDER, R.; WARREN, P. *Semantic Web Technologies: trends and research in ontology-based Systems*. [S.l.]: John Wiley & Sons, 2006.

Guarino 1998. GUARINO, N. Formal ontology and information systems. *FOIS98, Trento, Italy.*, p. 3–15., 1998.

Guha, McColl e Miller 2003 GUHA, R.; MCCOLL, R.; MILLER, E. Semantic search. *Proc. of the 12th international conference on World Wide Web*, A435, p. 700–709, 2003.

Han e Reeve 2005 HAN, H.; REEVE, L. Survey of semantic annotation platforms. *Proceedings of the ACM symposium on Applied computing*, p. 1634–1638, 2005.

Hodgins 2002 HODGINS, W. *Learning Object Metadata (LOM)*. 2002. Acessado em 3 de Fevereiro de 2011.

Leon e Perojo 2005 LEON, R. R.; PEROJO, K. R. Web semántica: un nuevo enfoque para la organización y recuperación de información en la web. *Acimed*, v. 16, 2005.

Mangold 2007 MANGOLD, C. A survey and classification of semantic search approaches. *Int. J. Metadata, Semantics and Ontology*, v. 2, p. 23–24, 2007.

Manning, Raghavan e Schtze 2008 MANNING, C. D.; RAGHAVAN, P.; SCHTZE, H. *Introduction to Information Retrieval*. New York, NY, USA: Cambridge University Press, 2008.

Nilas N.; Nilas e Masakul 2007 NILAS N.; NILAS, P.; MASAKUL, K. A spreading activation approach for e-commerce site selection system. 2007.

O'Reilly 2005 O'REILLY, T. *What Is Web 2.0. Design Patterns and Business Models for the Next Generation of Software*. September 2005. Http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html.

Rigo et al. 2010 RIGO, W. et al. Interfaces web baseadas em conhecimento para anotação de recursos de informação e gerenciamento de repositórios. *XXI Simpósio Brasileiro de Informática na Educação (SBIE), João Pessoa, PB.*, 2010.

Rumelhart D.E; Norman 1983 RUMELHART D.E; NORMAN, D. Representation in memory. *Center for Human Information Processing Technical Report no. 116*, 1983.

Schiel 1989 SCHIEL, U. Abstractions in semantic networks: axiom schemata for generalization, aggregation and grouping. *SIGART Bull.*, p. 25–26, 1989. Disponível em: <http://doi.acm.org/10.1145/65751.65752>.

Wikipedia 2011 WIKIPEDIA. Partially ordered set. 2011. Disponível em: <http://en.wikipedia.org/wiki/Poset>.