Edberto Ferneda

Recuperação de Informação:

Análise sobre a contribuição da Ciência da Computação para a Ciência da Informação

Tese apresentada à Escola de Comunicação e Artes da Universidade de São Paulo como exigência parcial para obtenção do título de Doutor em Ciências da Comunicação.

Área de concentração: Ciência da Informação e Documentação.

Orientadora: Profª Drª Johanna Wilhelmina Smit

São Paulo
2003

A

ÉLCIO FERNEDA
e
ELZA FERNEDA

Meus pais.

Agradecimentos

À Prof[a] Dr[a] Johanna Smit,
por me propiciar a oportunidade de realizar este trabalho,
pela indicação de rumos e pelo constante incentivo.

—

À Prof[a] Dr[a] Nair Kobashi,
pelo apoio e preciosas dicas.

—

À Cristina Ortega,
pela amizade e apoio.

—

Ao amigo Guilherme Ataíde Dias,
parceiro nestes anos de lutas e angústias,
mas também de muitas realizações.

—

À Eliany Alvarenga de Araújo,
"culpada" por minha intromissão na Ciência da Informação,
pela amizade e inspiração.

—

Aos meus irmãos, Edilson e Edmir,
pela ajuda e apoio constantes.

—

À Valquiria, companheira desde tempos imemoriáveis.

POESIA

Gastei uma hora pensando um verso
que a pena não quer escrever.
No entanto ele está cá dentro
inquieto, vivo.
Ele está cá dentro
e não quer sair.
Mas a poesia deste momento
inunda minha vida inteira.

Carlos Drummond de Andrade

Resumo

Desde o seu nascimento, a Ciência da Informação vem estudando métodos para o tratamento automático da informação. Esta pesquisa centrou-se na Recuperação de Informação, área que envolve a aplicação de métodos computacionais no tratamento e recuperação da informação, para avaliar em que medida a Ciência da Computação contribui para o avanço da Ciência da Informação. Inicialmente a Recuperação de Informação é contextualizada no corpo interdisciplinar da Ciência da Informação e são apresentados os elementos básicos do processo de recuperação de informação. Os modelos computacionais de recuperação de informação são analisados a partir da categorização em "quantitativos" e "dinâmicos". Algumas técnicas de processamento da linguagem natural utilizadas na recuperação de informação são igualmente discutidas. No contexto atual da Web são apresentadas as técnicas de representação e recuperação da informação desde os mecanismos de busca até a Web Semântica. Conclui-se que, apesar da inquestionável importância dos métodos e técnicas computacionais no tratamento da informação, estas se configuram apenas como ferramentas auxiliares, pois utilizam uma conceituação de "informação" extremamente restrita em relação àquela utilizada pela Ciência da Informação.

Palavras-chave: Informação, Ciência da Informação, Ciência da Computação, Recuperação de Informação, Modelos de recuperação de informação.

Abstract

Since its birth, Information Science has been studying methods for the automatic treatment of information. This research has focused on Information Retrieval, an area that involves the application of computational methods in the treatment and retrieval of information, in order to assess how Computer Science contributes to the progress of Information Science. Initially, Information Retrieval is contextualized in the interdisciplinary body of Information Science and, after that, the basic elements of the information retrieval process are presented. Computational models related to information retrieval are analyzed according to "quantitative" and "dynamic" categories. Some natural language processing techniques used in information retrieval are equally discussed. In the current context of the Web, the techniques of information retrieval are presented, from search engines to the Semantic Web. It can be concluded that in spite of the unquestionable importance of the computational methods and techniques for dealing with information, they are regarded only as auxiliary tools, because their concept of "information" is extremely restrict in relation to that used by the Information Science.

Keywords: Information, Information Science, Computer Science, Information Retrieval, Information Retrieval Models.

Sumário

Lista de Figuras

1 Introdução

O acelerado desenvolvimento tecnológico e a premência de métodos adequados para o tratamento da informação em grandes repositórios como a Internet impõem uma aproximação mais efetiva entre a Ciência da Computação e a Ciência da Informação. Porém, há de se observar as diferenças entre essas duas ciências que, embora compartilhem alguns interesses comuns, estão posicionadas em campos científicos bastantes distintos.

Nos últimos anos a palavra "*informação*" tem sido muito utilizada não só na constituição de discursos, mas também na criação de disciplinas ligadas à Ciência da Computação ou à Informática, além da Ciência da Informação. Com o imperativo tecnológico da sociedade contemporânea, o conceito de "informação" que se impõe é aquele que permite sua operacionalização através do computador ou outros dispositivos digitais.

1.1 Hipótese de pesquisa

Apesar do objetivo comum que motivou o nascimento quase contemporâneo da Ciência da Informação e da Ciência da Computação, observa-se uma grande distância teórica entre estas ciências. Esta distância é justificada inicialmente pelo fato de se tratarem de dois campos científicos bastante distintos. Em uma análise mais aprofundada verifica-se que a informação, objeto de comum interesse de ambas as ciências, é paradoxalmente o que mais as distancia. Na Ciência da Informação o conceito de informação está associado à semântica:

> "[...] *enquanto objeto da Ciência da Informação, a informação aparece como produto de um processo intencional, como algo construído, portanto, cujo propósito é o de promover a adequação significativa dos conteúdos.*" (Tálamo, 1997, p.11);

> "*A informação comporta um elemento de sentido. É um significado transmitido a um ser consciente por meio de uma mensagem inscrita em um suporte espaço-temporal: impresso, sinal elétrico, onda sonora, etc.*" (Le Coadic, 1996, p.5).

Na Ciência da Computação a definição de informação se aproxima à de Shannon e Weaver (1949), mais adequada à construção de sistemas informáticos nos quais *dados* podem ser totalmente descritos através de representações formais, podendo ser quantificados, armazenados em um computador e processados por ele:

> *"*[...] *não é possível processar informação diretamente em um computador. Para isso é necessário reduzi-la a dados."* (Setzer, 2001, p.242-243).

Em vista das diferenças entre a noção de informação utilizada pela Ciência da Informação, cujo componente semântico é evidente, e o conceito de informação empregado pela Ciência da Computação, a hipótese que se levanta neste trabalho é que, *no que se refere ao processo de recuperação de informação, as técnicas e procedimentos provenientes da Ciência da Computação apresentam-se apenas como mecanismos auxiliares no tratamento da informação, tal como definida no contexto da Ciência da Informação*.

1.2 Objetivos da pesquisa

O objetivo geral desta pesquisa pode ser enunciado da seguinte maneira:

> *Avaliar em quais aspectos a Ciência da Computação contribui para o avanço da Ciência da Informação, no que diz respeito ao processo de recuperação da informação.*

Este objetivo pode ser desdobrado nos seguintes objetivos específicos:

- *Analisar os recursos oriundos da Ciência da Computação mobilizados pelos sistemas de recuperação de informação;*

- *Analisar os impactos dos recursos oriundos da Ciência da Computação no processo de recuperação de informação;*
- *Verificar, face ao contexto atual da Web, como interagem os conceitos e processos da Ciência da Computação e da Ciência da Informação no que diz respeito aos mecanismos de recuperação de informação.*

1.3 Desenvolvimento da pesquisa

O presente trabalho inicia pela contextualização da Recuperação de Informação como produto da interdisciplinaridade da Ciência da Informação (Capítulo 2). No Capítulo 3 serão apresentados os elementos básicos do processo de recuperação de informação. Em seguida serão descritos os principais modelos de Recuperação de Informação empregados em sistemas automatizados, iniciando pelos clássicos modelos quantitativos (Capítulo 4) e avançando para os modelos dinâmicos (Capítulo 5). No Capítulo 6 serão vistas algumas técnicas de processamento da linguagem natural utilizadas na recuperação de informação. Para finalizar será feita uma análise da recuperação de informação na Web (Capítulo 7). A contribuição representada pela Ciência da Computação na Recuperação de Informação deverá ser dimensionada, a título de conclusão (Capítulo 8).

2
A Ciência da Informação

O nascimento da Ciência da Informação pode ser visto como conseqüência de uma sucessão de técnicas relacionadas com o registro físico do conhecimento, principalmente a escrita. A escrita permitiu registrar, estocar e recuperar o conhecimento, gerando uma espiral cumulativa de textos cujo potencial foi amplificado quando Johann Gutenberg inventou o tipo móvel e apresentou a primeira prensa na Europa.

O sucesso do invento de Gutenberg só não foi mais imediato pelo fato de que naquela época poucas pessoas sabiam ler. Em uma sociedade basicamente agrária, os camponeses nada tinham a ganhar com a alfabetização, e em geral não aspiravam a ela. Porém, a Revolução Industrial iniciada em meados do século XVIII provocou o êxodo das populações do campo para a cidade e deu impulso à procura por mais informação e à qualificação necessária para sua interpretação e utilização. A construção de estradas e o surgimento das estradas de ferro facilitaram a expansão do comércio e a distribuição de livros e jornais. A velocidade das mensagens passou da velocidade do cavalo para a da locomotiva e desta para a eletricidade.

Em 1822, Niépce apresentou a primeira fotografia, seguido por Louis Daguerre em 1839. A fotografia, que começou como diversão de amadores, em pouco tempo aliou-se à impressão nas técnicas de ilustração de livros e jornais. Assim como a palavra falada, a imagem pôde então ser preservada e transmitida entre gerações. Na década de 1840, John Benjamin Dancer combinou a fotografia com a microscopia e se tornou o pioneiro da

microfotografia e da microfilmagem. Em 1842, Alexander Bain "escaneou" uma imagem e enviou o resultado pelo telégrafo, criando o primeiro *fac-simile* da história. Novas invenções se seguiram durante a segunda metade do século XIX, a maioria delas ligadas à transmissão de informação. Em 1876 Alexander Graham Bell, que em 1844 havia inventado o telégrafo, estendeu o alcance da voz humana ao inventar o seu "telégrafo falante", o telefone. No ano seguinte Thomas Edison criou a primeira máquina de gravar sons e em 1879 projetou a lâmpada elétrica. (McGarry, 1999, p. 90-93).

Segundo Castells (1999, p. 53), esse período de transformações tecnológicas em aceleração marca uma descontinuidade histórica irreversível na base material da espécie humana. O repentino aumento de aplicações tecnológicas transformou os processos de produção e distribuição de bens e serviços, criou uma grande quantidade de novos produtos e mudou de maneira decisiva a localização das riquezas e do poder no mundo, que ficou ao alcance dos países e elites capazes de comandar esse sistema tecnológico.

No início do século XX o termo "Documentação" foi cunhado por Paul Otlet, que também a sistematizou e previu tecnologias que seriam úteis para sua operacionalização. Otlet, em seu "*Traité de Documentation*" (1934), mostra-se interessado em toda novidade tecnológica que permita condensar e organizar a informação de acordo com suas necessidades e objetivos. Otlet e Henri La Fontaine entraram para a história da biblioteconomia como autores da Classificação Decimal Universal (CDU). Em 1895 fundam em Bruxelas, na Bélgica, o *International Institute for Bibliography* - IIB, marco no desenvolvimento do que veio a se chamar Documentação e posteriormente Ciência da Informação. O primeiro objetivo do IIB era a elaboração do Repertório Bibliográfico Universal (RBU), que tinha a pretensão de sintetizar toda a produção bibliográfica internacional em fichas padronizadas. Para Otlet as fichas rompiam a linearidade do texto escrito, permitindo a livre associação entre as informações nelas registradas. Devidamente conectadas através dos códigos da CDU, essa rede de fichas pode ser vista como um prenúncio do hipertexto. As solicitações de pesquisa nesse grande banco de dados eram feitas através do correio e sua operacionalização era bastante demorada. Em uma época na qual não existiam fotocopiadoras ou computadores, era necessário remover as fichas do arquivo, copiá-las à mão e recolocá-las de volta no arquivo. Além da execução das "buscas", era também tarefa dos funcionários sintetizar e copiar nas fichas os materiais enviados por colaboradores de toda a parte do mundo (Rayward, 1997).

Otlet era um homem com imensa curiosidade em relação às inovações tecnológicas que pudessem ser úteis no processo de condensação e registro da informação. Fez diversas experimentações com a microfilmagem e previu um futuro promissor para uma invenção surgida na época: a televisão. Anteviu vários equipamentos tecnológicos como o fax, os microcomputadores, as *work-stations*, a Internet (Otlet, 1934, p. 389-391). Paul Otlet morreu em 1944, às vésperas do final da Segunda Guerra.

Após a Segunda Guerra Mundial, o entusiasmo na busca de soluções para os problemas advindos da explosão informacional pode ser resumido pelo artigo de Vannevar Bush (1945) intitulado "*As We May Think*". Nesse artigo, Bush define o problema do gerenciamento da informação e propõe como solução uma máquina, denominada Memex, que agregava as mais modernas tecnologias de informação existentes na época. O Memex nunca foi construído, mas as idéias que inspiraram sua idealização ainda fazem parte das aspirações de pesquisadores e cientistas da atualidade. Em uma escala muito maior, enfrenta-se hoje os mesmos problemas apontados por Otlet, e, como Bush, busca-se na tecnologia a solução para tais problemas.

2.1 A Ciência da Informação e o conceito de informação

Segundo Shera e Cleveland (1977), a década de 60 forneceu um clima favorável para o desenvolvimento da Ciência da Informação. Os problemas relacionados com o tratamento da informação começavam a ser abordados por parte da comunidade científica mundial, ao mesmo tempo em que se vivia um período de acelerado desenvolvimento tecnológico.

A primeira formulação do que seria a Ciência da Informação surgiu como resultado das conferências do *Georgia Institute of Technology* (ou simplesmente "*Georgia Tech*"), realizadas entre 1961 e 1962:

> "[Ciência da Informação é] *a ciência que investiga as propriedades e comportamento da informação, as forças que regem o fluxo da informação e os meios de processamento da informação para uma acessibilidade e usabilidade ótimas. Os processos incluem a origem, disseminação, coleta, organização, recuperação, interpretação e uso da informação. O campo deriva de ou relaciona-se com a matemática, a lógica, a lingüística, a psicologia, a tecnologia da computação, a pesquisa operacional, as artes*

gráficas, as comunicações, a biblioteconomia, a administração e alguns outros campos" (Shera e Cleveland, 1977, p. 265).

Em 1968, Harold Borko formulou uma definição complementar, ressaltando suas características tanto de ciência pura como de ciência aplicada.

"*Ciência da Informação é a disciplina que investiga as propriedades e o comportamento da informação, as forças que regem o fluxo da informação e os meios de processamento da informação para acessibilidade e usabilidade ótimas. Está relacionada com o corpo de conhecimento que abrange a origem, coleta, organização, armazenamento, recuperação, interpretação, transmissão, transformação e utilização da informação. Isto inclui a investigação das representações da informação nos sistemas naturais e artificiais, o uso de códigos para a transmissão eficiente de mensagem, e o estudo dos dispositivos e técnicas de processamento de informação tais como computadores e seus sistemas. É uma ciência interdisciplinar derivada de e relacionada a vários campos tais como matemática, lógica, lingüística, psicologia, tecnologia da computação, pesquisa operacional, artes gráficas, comunicações, biblioteconomia, administração e outros campos similares. Possui um componente de ciência pura, que investiga o assunto sem considerar suas aplicações, e um componente de ciência aplicada, que desenvolve serviços e produtos.*" (Borko, 1968, p. 3).

Saracevic (1996, p. 47), aponta que:

"*a Ciência da Informação é um campo dedicado às questões científicas e à prática profissional voltadas para os problemas da efetiva comunicação do conhecimento e de seus registros entre os seres humanos, no contexto social, institucional ou individual do uso e das necessidades de informação. No tratamento destas questões são consideradas de particular interesse as vantagens das modernas tecnologias informacionais*".

O componente tecnológico, principalmente a "tecnologia da computação", aparece em várias definições de Ciência da Informação. Alguns autores inserem a tecnologia em uma

posição central, outros a colocam como resultado da interdisciplinaridade da Ciência da Informação.

A natureza interdisciplinar da Ciência da Informação propicia o surgimento de diferentes correntes e estimula discussões sobre o seu objeto de estudo, a informação. Nesse ambiente, onde se juntam conceitos de áreas diversas, a construção de conceitos interdisciplinares apresenta-se como um grande desafio. De todo modo, a sistematização da Ciência da Informação deve passar obrigatoriamente pela definição do conceito de "informação".

Segundo McGarry (1999, p. 3), a palavra "informação" tornou-se popular logo após a invenção da imprensa no século XV, quando normalmente se utilizava uma palavra em latim para expressar uma nova idéia ou conceito. A raiz do termo vem de *formatio* e *forma*, ambos transmitindo a idéia de "moldar algo" ou dar "forma a" algo indeterminado.

Claude Shannon define informação como:

> "*O que acrescenta algo a uma representação* [...] *Recebemos informação quando o que conhecemos se modifica. Informação é aquilo que logicamente justifica alteração ou reforço de uma representação ou estado de coisas. As representações podem ser explicitadas como num mapa ou proposição, ou implícitas como no estado de atividade orientada para um objetivo do receptor*". (Shannon e Weaver,1949, p. 3, citado em McGarry, 1999, p. 3)

Na visão de Shannon, a informação não depende de um suporte material, mas de um emissor, um receptor e um canal, podendo ser facilmente quantificada. Esta definição de informação, base da Teoria da Informação, foi fundamental na construção dos primeiros computadores eletrônicos, e ainda desempenha um papel importante no estudo da informação em diversos contextos.

Numa abordagem pragmática, Buckland (1991b) identifica três principais usos do termo "informação":

- **Como processo** - o ato de informar ou a comunicação do conhecimento ou notícias sobre um fato ou ocorrência;

- **Como conhecimento** - o que é percebido pela informação enquanto processo, o conhecimento comunicado. Sua principal característica é a intangibilidade;
- **Como coisa -** aquilo que é visto como informativo: objetos, documentos, textos, dados ou eventos. A sua principal característica é a sua tangibilidade, sua materialidade.

Nos dois primeiros usos a informação para ser comunicada precisa estar "*expressa, descrita ou representada em algum modo físico*", em uma forma tangível, que seria a informação como coisa. Buckland define a "*informação como coisa*" em termos de potencial para o processo de informar, e defende o papel fundamental desta definição em sistema de recuperação de informação por este ser o único sentido com o qual tais sistemas podem lidar diretamente.

Hayes (1986), associando *dados* e *informação*, apresenta a seguinte definição:

> "*Informação é uma propriedade dos dados resultante de ou produzida por um processo realizado sobre os dados. O processo pode ser simplesmente a transmissão de dados (em cujo caso são aplicáveis a definição e medida utilizadas na teoria da comunicação); pode ser a seleção de dados; pode ser a organização de dados; pode ser a análise de dados*"

Ruyer, (1972, p. 3) apresenta a seguinte definição:

> "*A palavra 'informação', em seu sentido usual, parece comportar, necessariamente, um elemento de consciência e de sentido.* [...] *A informação, no sentido habitual do termo, é a transmissão a um ser consciente de uma significação, de uma noção, por meio de uma mensagem com base em um suporte espaço-temporal: imprensa, mensagem telefônica, onda sonora, etc.*"

Robredo (2003, cap. 1) apresenta e avalia diversos conceitos de informação. Inerente a quase todas as definições de informação analisadas no contexto da Ciência da Informação está evidenciado o seu caráter semântico.

2.2 A Ciência da Computação e sua relação com a Ciência da Informação

Pode-se apontar a Segunda Guerra Mundial como o marco inicial da Ciência da Computação, quando efetivamente se construíram os primeiros computadores digitais. Diferentemente da Ciência da Informação, é raro encontrar na literatura uma enunciação que defina o seu corpo teórico.

Denning et al (1989, p. 12) definem Ciência da Computação como:

> "[…] *o estudo sistemático de processos algorítmicos que descrevem e transferem informação: sua teoria, análise, projeto, eficiência, implementação e aplicação*. *A questão fundamental de toda a computação é: 'O que pode ser (eficientemente) automatizado?'* ".

De acordo com essa definição, a Ciência da Computação trata apenas dos processos que podem ser executados através de um conjunto seqüencial de instruções: os algoritmos.

Na introdução do livro intitulado "História da Computação – teoria e tecnologia", Fonseca Filho (1999, p. 13) define a Ciência da Computação como:

> "[...] *um corpo de conhecimento formado por uma infra-estrutura conceitual e um edifício tecnológico onde se materializam o* hardware *e o* software. *A primeira fundamenta a segunda e a precedeu.*"

De fato, a história da computação é formada por uma sucessão de personagens e suas idéias, direta ou indiretamente materializadas em programas (*software*) ou dispositivos (*hardware*). Essa história pode ser contada a partir de diversos referenciais, desde a criação do conceito abstrato de *número* até a criação dos primeiros computadores totalmente eletrônicos no início do século XX.

Na década de 50, vários cientistas, engenheiros e bibliotecários se empenharam na busca de soluções para os problemas enfrentados por Otlet no início do século e atualizados por Bush após a Segunda Guerra. Os primeiros resultados significativos no tratamento computacional da informação surgiram com os experimentos de Hans Peter Luhn na indexação automática e na elaboração automática de resumos. Engenheiro pesquisador da IBM, Luhn foi durante vários anos o criador de inúmeros projetos que visavam modificar radicalmente métodos tradicionais de armazenamento, tratamento e recuperação de

informação. Em 1961 Luhn já acumulava cerca de 80 patentes nos Estados Unidos (Schultz, 1968).

Em 1951, Calvin Mooers criou o termo "*Information Retrieval*" (Recuperação de Informação) e definiu os problemas a serem abordados por esta nova disciplina.

> "*A Recuperação de Informação trata dos aspectos intelectuais da descrição da informação e sua especificação para busca, e também de qualquer sistema, técnicas ou máquinas que são empregadas para realizar esta operação.*" (Mooers, 1951)

A Recuperação de Informação se firmou como uma área de pesquisa autônoma no seio da Ciência da Informação, com um acelerado desenvolvimento. Para Saracevic (1999), a Recuperação de Informação pode ser considerada a vertente tecnológica da Ciência da Informação e é resultado da relação desta com a Ciência da Computação.

O termo "recuperação de informação" atribuído a sistemas computacionais é ainda hoje bastante questionado, sendo que muitos autores preferem o termo "recuperação de documento" ("*document retrieval*") ou "recuperação de textos" ("*text retrieval*"). De fato, os sistemas não recuperam "informação", mas sim documentos ou referências cujo conteúdo poderá ser relevante para a necessidade de informação do usuário. Neste trabalho será utilizada a designação original "recuperação de informação", ficando subentendido que se trata de uma "informação" potencial, uma probabilidade de informação contida nos documentos ou textos recuperados pelo sistema, e que "*só vai se consubstanciar a partir do estímulo externo-documento, se também houver uma identificação (em vários níveis) da linguagem desse documento, e uma alteração, uma reordenação mental do receptor-usuário*" (Braga, 1995, p. 86).

A década de 60 foi um período bastante fértil de novas idéias relacionadas à Recuperação de Informação. Maron e Kuhns (1960) lançam os princípios básicos do modelo probabilístico para a recuperação de informação, que mais de quinze anos depois seria formalmente definido por Robertson e Jones (1976). Em meados dos anos 60 inicia-se uma longa série de experimentos que constitui um marco na Recuperação de Informação: o projeto SMART. Resultado da vida de pesquisa de Gerard Salton, este projeto produziu em mais de duas décadas, além de inúmeros artigos científicos, um modelo de recuperação de informação,

a criação e o aprimoramento de diversas técnicas computacionais e o sistema SMART (Salton, 1971).

Os primeiros sistemas de recuperação de informação baseavam-se na contagem de freqüência das palavras do texto e na eliminação de palavras reconhecidamente de pouca relevância. Nos trabalhos de Luhn e Salton observa-se inicialmente uma crença de que os métodos puramente estatísticos seriam suficientes para tratar os problemas relacionados à recuperação de informação. Porém, no transcorrer de suas pesquisas, percebe-se uma busca por métodos de análise semântica mais sofisticada. Desde os seus primeiros trabalhos, Salton se mostra interessado pela utilização de processos de tratamento da linguagem natural na recuperação de informação. Em livro de 1983, Salton e McGill apresentam em um capítulo intitulado "*Future directions in Information Retrieval*" a aplicação do processamento da linguagem natural e da lógica *fuzzy* na recuperação de informação, apontando a direção de futuras pesquisas para a Inteligência Artificial.

Embora a utilização de técnicas da Inteligência Artificial tenha surgido em conseqüência de uma natural evolução dos modelos matemáticos na busca de um aprofundamento semântico no tratamento textual, as pesquisas utilizando modelos estatísticos continuaram gerando novos modelos e aperfeiçoando antigas idéias. É o caso do modelo booleano estendido e de diversos outros modelos que foram atualizados tendo em vista a premência de métodos de recuperação para a Web.

A aproximação da Ciência da Informação com a Inteligência Artificial deu-se inicialmente através da automação de alguns processos documentários como a indexação e a elaboração de resumos. É através do Processamento da Linguagem Natural que esta aproximação se opera, tendo como objetivo a representação da semântica do texto, como será descrito no Capítulo 6.

Além do Processamento da Linguagem Natural, outras áreas da Inteligência Artificial são empregadas na solução dos problemas da recuperação de informação. É o caso dos sistemas especialistas, das redes neurais e dos algoritmos genéticos, apresentados detalhadamente no Capítulo 5. Na Ciência da Computação a pesquisa em redes neurais está inserida na vertente conexionista da Inteligência Artificial, que visa a modelagem da inteligência humana através da simulação dos componentes do cérebro. As redes neurais introduzem nos sistemas de recuperação a capacidade de se *adaptarem* ao "meio ambiente", isto é, às buscas dos usuários. Já os algoritmos genéticos implementam uma representação dos

mecanismos da evolução natural e dos processos genéticos da reprodução humana. Os sistemas de recuperação baseados nos algoritmos genéticos possuem a capacidade de *evoluírem*, alterando progressivamente as representações (código genético) dos documentos. Estes potenciais modelos de recuperação podem ser vistos como possíveis soluções para a urgência de métodos que consigam não só lidar com a quantidade de informação, mas também que possibilitem uma melhor qualidade da informação recuperada em relação à necessidade de informação específicas e individuais.

A aplicação de técnicas típicas da Inteligência Artificial na recuperação de informação geralmente se dá através de pesquisadores ligados à Ciência da Computação, que se "aventuram" na Ciência da Informação com o objetivo de verificar a aplicabilidade de tais técnicas em outros campos. Após o desenvolvimento de pequenos protótipos e de alguns resultados práticos, retornam às pesquisas em sua ciência de origem, sem consolidar avanços significativos na Recuperação de Informação. Essa natural divergência de interesses nas pesquisas é pelo menos parcialmente rompida com o surgimento da Internet e da Web. A Web promoveu um rápido direcionamento nos esforços de pesquisa dos mais variados campos científicos para os problemas relacionados à recuperação de informação. Se muitas vezes a obra de Paul Otlet é criticada por seu centralismo autoritário e seu monumentalismo, o que vemos na Web são problemas gerados por uma exagerada "democracia informacional" em uma dimensão que supera o "monumental".

3
A Recuperação de Informação

No contexto da Ciência da Informação, o termo "recuperação de informação" significa, para uns, a operação pela qual se seleciona documentos, a partir do acervo, em função da demanda do usuário. Para outros, "recuperação de informação" consiste no fornecimento, a partir de uma demanda definida pelo usuário, dos elementos de informação documentária correspondentes. O termo pode ainda ser empregado para designar a operação que fornece uma resposta mais ou menos elaborada a uma demanda, e esta resposta é convertida num produto cujo formato é acordado com o usuário (bibliografia, nota de síntese, etc.). Há ainda autores que conceituam a recuperação de informação de forma muito mais ampla, ao subordinar à mesma o tratamento da informação (catalogação, indexação, classificação). Como apresentado no capítulo anterior, o termo Recuperação de Informação (*Information Retireval*) designa também uma área de pesquisa fundada por Calvin Mooers em 1951.

Este trabalho optou por uma abordagem que enfatiza os processos de busca de informação, excluindo, portanto, o tratamento documental que, embora complementar, mobiliza uma outra bibliografia.

O processo de recuperação de informação consiste em identificar, no conjunto de documentos (*corpus*) de um sistema, quais atendem à necessidade de informação do usuário. O usuário de um sistema de recuperação de informação está, portanto, interessado em recuperar "informação" sobre um determinado assunto e não em recuperar dados que

satisfazem sua expressão de busca, nem tampouco documentos, embora seja nestes que a informação estará registrada. Essa característica é o que diferencia os sistemas de recuperação de informação dos Sistemas Gerenciadores de Bancos de Dados (ou simplesmente "bancos de dados"), estudados e implementados desde o nascimento da Ciência da Computação.

Os sistemas de banco de dados têm por objetivo a recuperação de todos os objetos ou itens que satisfazem precisamente às condições formuladas através de uma expressão de busca. Em um sistema de recuperação de informação essa precisão não é tão estrita. A principal razão para esta diferença está na natureza dos objetos tratados por estes dois tipos de sistema. Os sistemas de recuperação de informação lidam com objetos lingüísticos (textos) e herdam toda a problemática inerente ao tratamento da linguagem natural. Já um sistema de banco de dados organiza itens de "informação" (dados), que têm uma estrutura e uma semântica bem definidas. Os sistemas de informação podem se aproximar do padrão que caracteriza os bancos de dados na medida em que sejam submetidos a rígidos controles, tais como vocabulário controlado, listas de autoridades, etc.

Os sistemas de recuperação de informação devem representar o conteúdo dos documentos do *corpus* e apresentá-los ao usuário de uma maneira que lhe permita uma rápida seleção dos itens que satisfazem total ou parcialmente à sua necessidade de informação, formalizada através da uma expressão de busca. Uma representação simplificada do processo de recuperação de informação é apresentada na Figura 1.

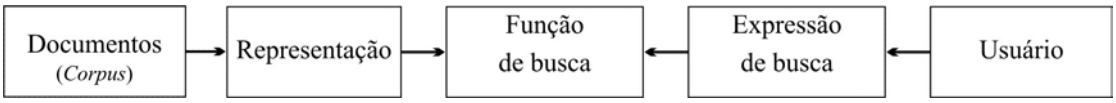


Figura 1 Representação do processo de recuperação de informação

A fim de se tentar esboçar um esquema do processo de recuperação de informação, será utilizado o conceito de "*informação como coisa*" definido por Buckland (1991b), para quem os itens que formam os sistemas de informação seriam registros relacionados a *coisas* ou objetos. Para o referido autor, o termo *informação* é utilizado na maioria das vezes vinculado a um objeto que contém informação: um documento. Assim, o termo *informação* poderia também designar "*algo atribuído a um objeto, tal como dado e documento que se referem à informação, porque deles se espera que sejam informativos*". Por sua vez o termo *documento*, entendido como coisa informativa, incluiria, por exemplo, objetos, artefatos, imagens e sons.

Suzanne Briet (1951, p. 7, citado por Buckland,1997, p. 806) define documento como "*qualquer signo físico ou simbólico, preservado ou registrado, com a intenção de representar, reconstruir ou demonstrar um fenômeno físico ou abstrato*". Esta definição generaliza ainda mais o conceito de documento a qualquer tipo de suporte, seja ele material ou digital.

No ambiente digital que vem se configurando nas últimas décadas, os acervos de objetos digitais se multiplicam tanto no que se refere à sua tipologia quanto à sua complexidade. Nesse novo cenário, textos, imagens, sons, vídeos, páginas Web e diversos outros objetos digitais requerem diferentes tipos de tratamento e representação para uma recuperação de informação eficaz (Burke, 1999). Particularmente no contexto da Web, uma das principais mudanças é a desterritorialização do documento e a sua desvinculação de uma forma física tradicional como o papel, possibilitando uma integração entre diferentes suportes (texto, imagem, som) e uma ruptura na linearidade do acesso aos documentos através do imenso hipertexto da Web, cujas características, no que se refere à recuperação de informação, são detalhadas no Capítulo 7.

Com as mudanças do conceito de *documento* advindas dos meios digitais, o tratamento da informação envolve elementos relacionados a diversas disciplinas, ampliando o campo de pesquisa da Ciência da Informação e reforçando ainda mais sua característica interdisciplinar, principalmente no seu relacionamento com a Ciência da Computação, mais notadamente no contexto da Web.

O processo de *representação* busca descrever ou identificar cada documento do *corpus* através de seu conteúdo. Tal representação geralmente é realizada através do processo de indexação. Durante a indexação são extraídos conceitos do documento através da análise de seu conteúdo e traduzidos em termos de uma linguagem de indexação, tais como cabeçalhos de assunto, tesauros, etc. Esta representação identifica o documento e define seus pontos de acesso para a busca e pode também ser utilizada como seu substituto.

A analise de um documento pode envolver uma interpretação de seu conteúdo com a finalidade de agregar assuntos que não estão diretamente explicitados em sua superfície textual, mas que podem ser facilmente abstraídos por um indexador humano. A indexação de um documento pode também ser efetuada tendo em vista a sua recuperação. Nesse caso a análise do documento é feita com a preocupação de tornar o seu conteúdo visível para os usuários de um sistema de informação.

A automação do processo de indexação só é possível através de uma simplificação na qual se considera que os assuntos de um documento podem ser derivados de sua estrutura textual através de métodos algorítmicos. A principal vantagem da automação está no seu baixo custo, considerando o crescente barateamento dos computadores e dos softwares.

Os métodos automáticos de indexação geralmente utilizam "filtros" para eliminar palavras de pouca significação (*stop words*), além de normalizar os termos reduzindo-os a seus radicais, processo conhecido como *stemming*. Essa forma de indexação seleciona formas significantes (termos ou frases) dos documentos, desconsiderando os significados que os mesmos podem possuir de acordo com os contextos. Embora esta forma de indexação seja amplamente utilizada, suas falhas e limitações se evidenciam pela simplificação da dimensão semântica da linguagem.

Com o aumento da capacidade de armazenamento dos computadores, muitos sistemas conseguem manter disponíveis os textos dos documentos. Nesses sistemas, chamados sistemas de *texto completo* ou *texto integral*, não há de fato uma *representação* ou poder-se-ia considerar que tal representação é feita pelo conjunto formado por todas as palavras de seu texto. Com um aumento da quantidade de documentos, mesmo os computadores modernos podem não comportar o armazenamento dos textos dos documentos, tendo que limitar a representação a um conjunto limitado de termos.

A necessidade de informação do usuário é representada através de sua *expressão de busca*, que pode ser especificada em linguagem natural ou através de uma linguagem artificial, e deve resultar na recuperação de um número de documentos que possibilite a verificação de cada um deles a fim de selecionar os que são úteis. A principal dificuldade do usuário está em predizer, por meio de uma expressão de busca, as palavras ou expressões que foram usadas para representar os documentos e que satisfarão sua necessidade. As estratégias que podem ou devem ser utilizadas na formulação de buscas são tema de diversas pesquisas da Ciência da Informação. Com o aumento da quantidade de documentos disponibilizados nos sistemas de informação este processo de predição, que nunca é tão preciso como nos sistemas de banco de dados, é dificultado pelo número elevado de documentos resultantes das buscas. Assim, não é suficiente predizer um ou mais termos utilizados para indexar os documentos desejados, é necessário também evitar a recuperação de documentos não relevantes, minimizando o esforço em verificar a relevância de tais documentos.

O *usuário* de um sistema de informação tem que traduzir a sua necessidade de informação em uma expressão de busca através de uma linguagem fornecida pelo sistema. Geralmente a expressão de busca é composta de um conjunto de palavras que tentam exprimir a semântica da necessidade de informação do usuário. A subjetividade do processo de recuperação de informação faz com que muita da responsabilidade de sua eficácia seja transferida para o usuário.

A partir de meados da década de 70 iniciou-se um debate sobre um novo paradigma teórico denominado "abordagem centrada no usuário" (Ferreira, 95). Segundo essa perspectiva, a recuperação de informação é um processo de produção de sentido por parte do usuário, o qual utiliza a informação para construir conhecimento. Para Morris (1994), a informação é parcialmente construída pelo usuário durante esse processo de produção de sentido, e só existe fora dele de maneira incompleta. Portanto, segundo essa perspectiva os sistemas de informação deveriam ser modelados conforme a natureza das necessidades de informação do usuário, levando-se em conta os seus padrões de comportamento na busca da informação.

Embora exista um consenso sobre a importância de se estabelecer o usuário como o centro do processo de recuperação de informação, a abordagem centrada no usuário carece de definições e pressupostos claros para identificar variáveis e gerar questões de pesquisa, além de metodologias específicas e rigor científico.

No centro do processo de recuperação de informação está a *função de busca*, que compara as representações dos documentos com a expressão de busca dos usuários e recupera os itens que supostamente fornecem a informação que o usuário procura. Porém, o fato de um termo utilizado na expressão de busca aparecer na representação de um documento não significa que o documento seja relevante para a necessidade do usuário. Em primeiro lugar, a busca provavelmente contém mais do que um termo e, portanto, a recuperação de um documento deve considerar a totalidade dos termos de busca. Em segundo lugar, o termo presente na representação de um documento pode estar em um contexto que não é apropriado à necessidade do usuário. Por último, um documento, mesmo que fortemente relacionado com uma busca, pode não ser relevante para o usuário, simplesmente por ser muito antigo ou por já ter sido recuperado anteriormente pelo mesmo.

A eficiência de um sistema de recuperação de informação está diretamente ligada ao modelo que o mesmo utiliza. Um modelo, por sua vez, influencia diretamente no modo de

operação do sistema. Apesar de alguns desses modelos terem sido criados nos anos 60 e 70 ("modelos quantitativos", Capítulo 4) e aperfeiçoados nos anos 80, as suas principais idéias ainda estão presentes na maioria dos sistemas de recuperação atuais e nos mecanismos de busca da Web. Alguns outros modelos, chamados aqui de "modelos dinâmicos" (Capítulo 5), resumem propostas mais recentes que utilizam métodos derivados da Inteligência Artificial e representam alternativas promissoras a serem estudadas e desenvolvidas futuramente.

4
Modelos quantitativos

A grande maioria dos modelos de recuperação de informação é de natureza quantitativa, baseados em disciplinas como a lógica, a estatística e a teoria dos conjuntos. Em um estudo sobre os modelos de recuperação de informação, Robertson (1977) justifica esse predomínio pelo fato de que a determinação de um modelo matemático geralmente pressupõe uma cuidadosa análise formal do problema e especificações de hipóteses, além de uma formulação explícita da forma como o modelo depende das hipóteses.

Nos modelos de recuperação de informação apresentados neste capítulo os documentos são representados por um conjunto de termos de indexação. Um termo de indexação é geralmente uma palavra que representa um conceito ou significado presente no documento. Porém, os termos de indexação associados a um documento não são igualmente úteis para descrever o seu conteúdo. Existem termos mais representativos do assunto principal do documento e outros termos que representam assuntos periféricos à temática do mesmo. Decidir a importância de um termo para a descrição do conteúdo de um documento não é uma tarefa simples, mesmo para pessoas experientes. Alguns sistemas computacionais utilizam propriedades que facilitam a mensuração do potencial representativo de um termo de indexação. Por exemplo, em um *corpus* com milhares de documentos uma palavra que aparece em todos os documentos não seria um bom termo de indexação. Por outro lado, uma palavra que aparece em apenas três documentos possivelmente seria de grande utilidade como termo de indexação, pois reduziria consideravelmente o número de documentos que poderiam

ser de interesse para uma determinada necessidade de informação do usuário. Portanto, diferentes termos de indexação possuem graus de relevância distintos, de acordo com os documentos e os objetivos do sistema de informação.

4.1 Modelo booleano

A Lógica como ciência começou a se desenvolver com o filósofo Aristóteles. Através da leitura dos diálogos de Platão, Aristóteles descobriu que existe uma lei que rege o pensamento para que se atinja o conhecimento de algo, a verdade, sem cair em contradição. Para Aristóteles, a lógica seria um instrumento para a ciência e a filosofia. A lógica aristotélica estava a serviço de uma explicação da realidade e baseava-se na distinção entre *verdadeiro* e *falso*.

Investigando os tipos de raciocínio, Aristóteles construiu uma teoria cujo núcleo é a caracterização e análise dos *silogismos*. Um exemplo típico de silogismo é:

Todo homem é mortal
Sócrates é homem
Logo, Sócrates é mortal

Uma característica importante da silogística aristotélica é a utilização de símbolos que representam expressões substantivas e possibilitam estabelecer um certo rigor nas demonstrações matemáticas.

Apesar das limitações para representar todos os tipos de inferências, o domínio da lógica silogística prevaleceu até o século XIX, quando George Boole concebeu um sistema de símbolos e regras aplicável desde números até enunciados. Com esse sistema é possível codificar proposições em linguagem simbólica e manipulá-las quase da mesma maneira como se faz com os números. Com o trabalho de Boole, a Lógica afasta-se da Filosofia e aproxima-se da Matemática.

A álgebra booleana é um sistema binário no qual existem somente dois valores possíveis para qualquer símbolo algébrico: 1 ou 0, *verdadeiro* ou *falso*. Essa teoria revelou-se ideal para o funcionamento de circuitos eletrônicos e foi fundamental na idealização da arquitetura dos computadores modernos.

4.1.1 Operadores booleanos

No modelo booleano um documento é representado por um conjunto de termos de indexação que podem ser definidos de forma intelectual (manual) por profissionais especializados ou automaticamente, através da utilização de algum tipo de algoritmo computacional. As buscas são formuladas através de uma expressão booleana composta por termos ligados através dos operadores lógicos AND, OR e NOT (E, OU e NÃO)[1], e apresentam como resultado os documentos cuja representação satisfazem às restrições lógicas da expressão de busca.

Uma expressão conjuntiva de enunciado t_1 **AND** t_2 recuperará documentos indexados por ambos os termos (t_1 e t_2). Esta operação equivale à *interseção* do conjunto dos documentos indexados pelo termo t_1 com o conjunto dos documentos indexados pelo termo t_2, representado pela área cinza na Figura 2

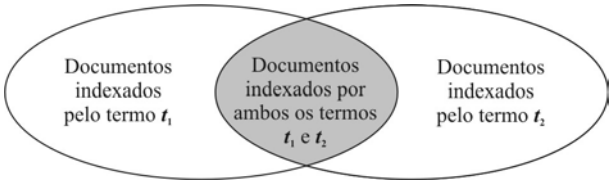


Figura 2 Representação do resultado de uma expressão booleana conjuntiva (AND)

Uma expressão disjuntiva t_1 **OR** t_2 recuperará o conjunto dos documentos indexados pelo termo t_1 ou pelo termo t_2. Essa operação equivale à *união* entre o conjunto dos documentos indexados pelo termo t_1 e o conjunto dos documentos indexados pelo termo t_2, como ilustrado na Figura 3.

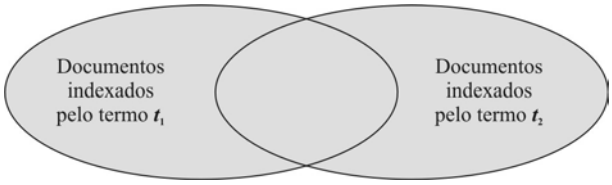


Figura 3 Resultado de uma busca booleana disjuntiva (OR)

[1] Será utilizada a terminologia em inglês em função de sua ampla disseminação.

Uma expressão que utiliza apenas um termo t_1 terá como resultado o conjunto de documentos indexados por t_1. A expressão **NOT** t_1 recuperará os documentos que não são indexados pelo termo t_1, representados pela área cinza da Figura 4

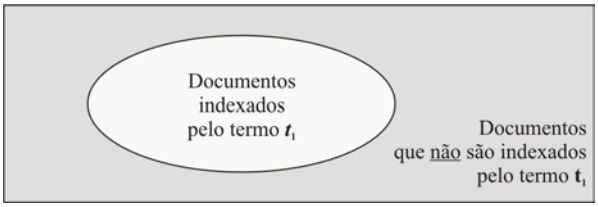


Figura 4 Resultado de uma busca negativa (NOT)

As expressões t_1 **NOT** t_2 ou t_1 **AND NOT** t_2 terão o mesmo resultado: o conjunto dos documentos indexados por t_1 e que não são indexados por t_2 (Figura 5). Neste caso o operador NOT pode ser visto como um operador da diferença entre conjuntos. Assim, a área cinza da Figura 5 representa o conjunto dos documentos indexados pelos termo t_1 *menos* (*subtraído de*) o conjunto dos documentos indexados por t_2.

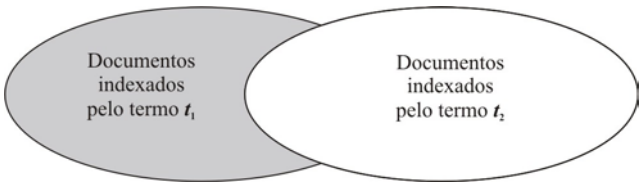
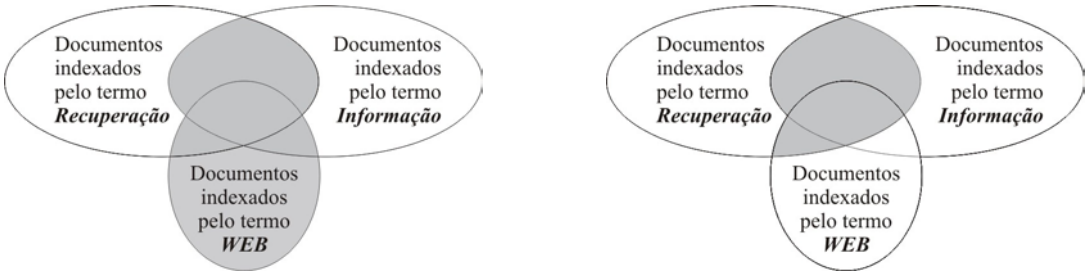


Figura 5 Resultado de uma busca booleana com o operador NOT

Termos e operadores booleanos podem ser combinados para especificar buscas mais detalhadas ou restritivas. Como a ordem de execução das operações lógicas de uma expressão influencia no resultado da busca, muitas vezes é necessário explicitar essa ordem delimitando partes da expressão através de parênteses. Na ausência de parênteses, a expressão booleana será interpretada de acordo com o padrão utilizado pelo sistema, que pode ser a execução da expressão da esquerda para a direita ou em uma ordem pré-estabelecida, geralmente NOT - AND - OR.



(a) (*Recuperação* AND *Informação*) OR *WEB* **(b)** *Recuperação* AND (*Informação* OR *WEB*)

Figura 6 Resultado de uma expressão de busca booleana utilizando parênteses

As áreas cinzas da Figura 6 representam o resultado de duas expressões de busca que utilizam os mesmos termos e os mesmos operadores, mas diferem na ordem de execução. Na primeira expressão (a) inicialmente é executada a operação AND entre os termos "Recuperação" e "Informação". Com o resultado obtido é executada a operação OR com o termo "WEB". A segunda expressão (b) executa a operação OR entre os termos "Informação" e "WEB" e com o resultado é efetuada a operação AND com o termo "Recuperação".

Expressões complexas exigem um conhecimento profundo da lógica booleana e evidenciam a importância da elaboração de uma estratégia de busca adequada para garantir a qualidade da informação recuperada. O conhecimento da lógica booleana é importante também para entender e avaliar os resultados obtidos em uma busca.

4.1.2 Operadores de proximidade

Até a década de 60 os sistemas de recuperação de informação utilizavam apenas pequenos resumos ou algumas palavras-chave para representar o conteúdo dos documentos. Os recursos computacionais existentes não permitiam o armazenamento de todo o texto dos documentos.

Durante os anos 70, a diminuição gradual do custo e o aumento na capacidade dos computadores permitiram aos sistemas armazenar o texto completo dos documentos e não apenas sua representação. Nesse período surgiram também os primeiros editores de texto, o que permitiu um aumento na disponibilidade de documentos digitais (Lesk, 1995).

Em um sistema de recuperação de texto completo (*full-text*) cada documento é representado pelo conjunto de todas as palavras de seu texto. Tais sistemas possuem recursos que permitem recuperar documentos através da avaliação da proximidade entre palavras do

texto do documento como um todo ou em unidades textuais específicas como sentença ou parágrafo. Durante o processo de busca o usuário tenta predizer palavras ou frases que podem aparecer no texto dos documentos e que são relevantes em relação à sua necessidade de informação. Os operadores de proximidade permitem especificar condições relacionadas à distância e à posição dos termos no texto.

O formato genérico de um operador de proximidade pode ser representado como:

$$\mathbf{t_1} \quad \underline{\boldsymbol{n}\ \textit{unidades de}} \quad \mathbf{t_2}$$

A distância ***n*** é um número inteiro e "unidades" podem ser palavras, sentenças ou parágrafos.

No sistema STAIRS, desenvolvido pela IBM, por exemplo, a expressão de busca t_1 **WITH** t_2 permite recuperar documentos cujos termos t_1 e t_2 apareçam no mesmo parágrafo. A expressão t_1 **SAME** t_2 recuperará documentos onde o termo t_1 e o termo t_2 apareçam em uma mesma sentença.

Um outro operador de proximidade bastante comum nos sistemas de recuperação de informação e nos mecanismos de busca da Web é o operador ADJ. Este operador permite pesquisar duas palavras adjacentes no texto de um documento, na ordem especificada na expressão de busca. Por exemplo, a expressão ***pronto*** **ADJ** ***socorro*** terá como resultado os documentos que tiverem a palavra "pronto" seguida da palavra "socorro", isto é, recuperará documentos que contêm a expressão "pronto socorro". Em muitos sistemas é possível utilizar diretamente um termo composto delimitando as suas palavras com aspas. Assim, a expressão de busca ***pronto*** **ADJ** ***socorro*** equivale à expressão ***"pronto socorro"***. Uma variação do operador ADJ permite selecionar documentos que possuem em seu texto duas palavras específicas em uma mesma frase, separadas por um número máximo de palavras e na ordem especificada na expressão de busca. Por exemplo, a expressão ***política*** **ADJ5** ***saúde*** terá como resultado um conjunto de documentos que possuem em uma mesma sentença as palavras "política" e "saúde", nessa ordem, separadas por no máximo 5 palavras.

Outro operador bastante comum é o operador NEAR. No mecanismo de busca Lycos (www.lycos.com), por exemplo, a expressão de busca ***política*** **NEAR/10** ***social*** recuperará documentos nos quais a palavra "política" apareça a no máximo 10 palavras de distância da palavra "social", não importando a ordem em que elas se encontram.

Os operadores booleanos podem ser combinados com os operadores de proximidade a fim de formar expressões de busca mais restritivas ou mais genéricas. Por exemplo, a expressão ***"Recuperação de"* ADJ (*informação* OR *documentos*)** recuperará o conjunto dos documentos que contenham o termo "Recuperação de informação" ou o termo "Recuperação de documentos".

Blair (1990, p.47-53) apresenta um resumo crítico sobre os sistemas de recuperação de texto completo. Segundo o autor, a riqueza e flexibilidade da linguagem natural dificultam sensivelmente a predição de palavras ou frases que aparecem nos textos de documentos relevantes e ao mesmo tempo não ocorrem em documentos não relevantes.

Mesmo utilizando operadores de proximidade, o resultado de uma busca booleana será um conjunto de documentos que respondem verdadeiramente à expressão de busca e presumivelmente serão considerados relevantes pelo usuário. Apesar de os operadores de proximidade agregarem novos recursos aos sistemas de texto completo, tais operadores não alteram substancialmente as vantagens e limitações do modelo booleano.

O modelo booleano, apesar de bem formalizado, possui limitações que diminui sua atratividade. Algumas dessas limitações são:

- Sem um treinamento apropriado, o usuário leigo será capaz de formular somente buscas simples. Para buscas que exijam expressões mais complexas é necessário um conhecimento sólido da lógica booleana.
- Existe pouco controle sobre a quantidade de documentos resultante de uma busca. O usuário é incapaz de predizer quantos registros satisfarão a restrição lógica de uma determinada expressão booleana, sendo necessárias sucessivas reformulações antes que seja recuperado um volume aceitável de documentos;
- O resultado de uma busca booleana se caracteriza por uma simples partição do *corpus* em dois subconjuntos: os documentos que atendem à expressão de busca e aqueles que não atendem. Presume-se que todos os documentos recuperados são de igual utilidade para o usuário. Não há nenhum mecanismo pelo qual os documentos possam ser ordenados;
- Não existe uma forma de atribuir importância relativa aos diferentes termos da expressão booleana. Assume-se implicitamente que todos os termos têm o mesmo peso.

Um erro freqüente na formulação de expressões booleanas é a interpretação equivocada do significado dos operados AND e OR. Na linguagem coloquial, quando se diz "gatos **e** cachorros", intuitivamente imagina-se uma *união* entre o conjunto dos "gatos" e o conjunto dos "cachorros". Em um sistema de recuperação de informação a expressão t_1 **AND** t_2 resultará na *interseção* entre o conjunto dos documentos indexados pelo termo t_1 e o conjunto dos documentos indexados por t_2. Na linguagem cotidiana, quando se diz "café **ou** chá" expressa-se uma escolha ou seleção cujo resultado será apenas um dos elementos. Em um sistema de recuperação de informação, a expressão t_1 **OR** t_2 resultará uma *união* do conjunto de documentos indexados por t_1 com o conjunto de documentos indexados por t_2 (Smith, 1993).

Apesar de suas limitações, o modelo booleano está presente em quase todos os sistemas de recuperação de informação, seja como a principal maneira de formular as expressões de busca, seja como um recurso alternativo. Uma razão para isso é que para usuários experientes este modelo oferece um certo controle sobre o sistema. Se o conjunto de documentos resultante é muito grande ou muito pequeno, é fácil saber quais os operadores necessários para diminuir ou aumentar a quantidade de documentos até atingir um resultado satisfatório.

Uma das maiores desvantagens do modelo booleano é a sua inabilidade em ordenar os documentos resultantes de uma busca. Por esta razão o modelo não seria adequado aos modernos sistemas de texto integral, como os mecanismos de busca da Web, onde o ordenamento dos documentos é de extrema importância face ao volume de documentos que geralmente é recuperado. Apesar disso, muitos desses sistemas se desenvolveram utilizando o modelo booleano como ponto de partida para a implementação de novos recursos de recuperação. Neste sentido o modelo booleano pode ser considerado o modelo mais utilizado não só nos sistema de recuperação de informação e nos mecanismos de busca da Web, mas também nos sistemas de banco de dados, onde o seu poder se expressa através da linguagem SQL.

4.2 Modelo vetorial

O modelo vetorial propõe um ambiente no qual é possível obter documentos que respondem parcialmente a uma expressão de busca. Isto é feito através da associação de pesos tanto aos termos de indexação como aos termos da expressão de busca. Esses pesos são

utilizados para calcular o grau de similaridade entre a expressão de busca formulada pelo usuário e cada um dos documentos do *corpus*. Como resultado, obtém-se um conjunto de documentos ordenado pelo grau de similaridade de cada documento em relação à expressão de busca.

4.2.1 Representação vetorial

No modelo vetorial um documento é representado por um vetor onde cada elemento representa o peso, ou a relevância, do respectivo termo de indexação para o documento. Cada vetor descreve a posição do documento em um espaço multidimensional, onde cada termo de indexação representa uma dimensão ou eixo. Cada elemento do vetor (peso) é normalizado de forma a assumir valores entre zero e um. Os pesos mais próximos de um (1) indicam termos com maior importância para a descrição do documento. A Figura 7 apresenta a representação gráfica de um documento $\mathbf{DOC_1}$ com termos de indexação t_1 e t_3, com pesos 0.3 e 0.5, respectivamente.

	t_1	t_3
DOC_1	0.3	0.5

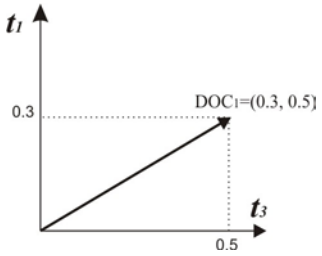


Figura 7 Representação vetorial de um documento com dois termos de indexação

A Figura 8 representa graficamente um documento $\mathbf{DOC_2}$ = (0.5, 0.4, 0.3) em um espaço tridimensional.

	t_1	t_2	t_3
DOC_2	0.5	0.4	0.3

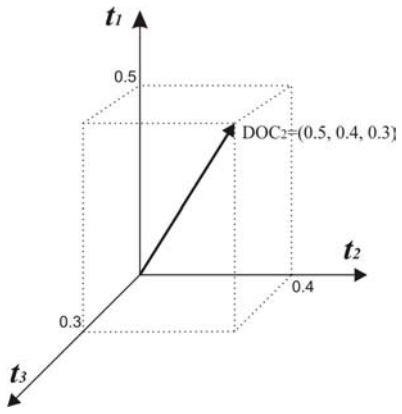


Figura 8 Representação vetorial de um documento com três termos de indexação

A Figura 9 mostra os dois documentos **DOC_1** e **DOC_2** representados em um mesmo espaço vetorial. Os números positivos representam os pesos de seus respectivos termos. Termos que não estão presentes em um determinado documento possuem peso igual a zero.

	t_1	t_2	t_3
DOC_1	0.3	0.0	0.5
DOC_2	0.5	0.4	0.3

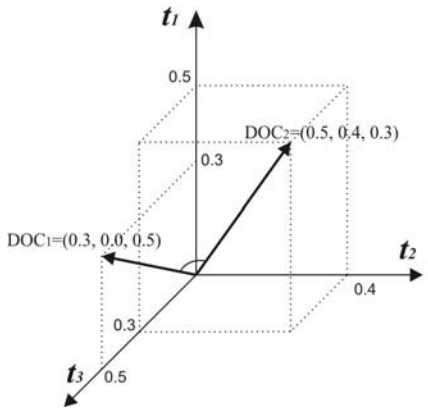


Figura 9 Espaço vetorial contendo dois documentos

Da mesma forma que os documentos, no modelo vetorial uma expressão de busca também é representada por um vetor numérico onde cada elemento representa a importância (peso) do respectivo termo na expressão de busca. A Figura 10 mostra a representação da expressão de busca **$eBUSCA_1$**=(0.2, 0.35, 0.1) juntamente com os documentos **DOC_1** e **DOC_2** em um espaço vetorial formado pelos termos t_1, t_2 e t_3.

	t_1	t_2	t_3
$eBUSCA_1$	0.2	0.35	0.1

	t_1	t_2	t_3
DOC_1	0.3	0.0	0.5
DOC_2	0.5	0.4	0.3

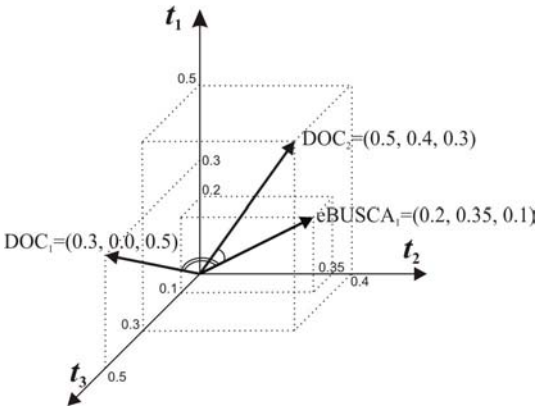


Figura 10 Representação de uma expressão de busca em um espaço vetorial

Para que fosse possível apresentar visualmente um espaço vetorial contendo documentos e expressões de buscas, nos exemplos acima foram utilizados apenas três termos de indexação na representação dos documentos. Obviamente, um sistema real contém um grande número de termos de indexação e documentos. Um *corpus* contendo um número

indefinido de documentos e termos de indexação pode ser representado através de uma matriz onde cada linha representa um documento e cada coluna representa a associação de um determinado termo aos diversos documentos. Um *corpus* contendo ***n*** documentos e ***i*** termos de indexação pode ser representado da seguinte forma:

	t_1	t_2	t_3	…	t_i
DOC$_1$	$w_{1,1}$	$w_{2,1}$	$w_{3,1}$	…	$w_{i,1}$
DOC$_2$	$w_{1,2}$	$w_{2,2}$	$w_{3,2}$	...	$w_{i,2}$
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.
DOC$_n$	$w_{1,n}$	$w_{2,n}$	$w_{3,n}$	...	$w_{i,n}$

onde $w_{i,n}$ representa o peso do *i*-ésimo termo do *n*-ésimo documento.

4.2.2 Cálculo da similaridade

A utilização de uma mesma representação tanto para os documentos como para as expressões de busca permite calcular o grau de similaridade entre dois documentos ou entre uma expressão e cada um dos documentos do *corpus*. Em um espaço vetorial contendo ***t*** dimensões a similaridade (***sim***) entre dois vetores ***x*** e ***y*** é calculada através do *co-seno* do ângulo formado por estes vetores, utilizando a seguinte fórmula:

$$sim(x, y) = \frac{\sum_{i=1}^{t}(w_{i,x} \times w_{i,y})}{\sqrt{\sum_{i=1}^{t}(w_{i,x})^2} \times \sqrt{\sum_{i=1}^{t}(w_{i,y})^2}}$$

onde $w_{i,x}$ é o peso do *i*-ésimo elemento do vetor *x* e $w_{i,y}$ é o peso do *i*-ésimo elemento do vetor *y*.

O grau de similaridade entre o documento **DOC$_1$** e o documento **DOC$_2$**, representados na Figura 9, é calculado como:

$$sim(\mathrm{DOC}_1, \mathrm{DOC}_2) = \frac{(0.3 \times 0.5) + (0.0 \times 0.4) + (0.5 \times 0.3)}{\sqrt{0.3^2 + 0.0^2 + 0.5^2} \times \sqrt{0.5^2 + 0.4^2 + 0.3^2}} = \frac{0.15 + 0.0 + 0.15}{\sqrt{0.34} \times \sqrt{0.5}} = 0.73$$

Portanto, o grau de similaridade entre os documentos **DOC$_1$** e **DOC$_2$** é igual a 0.73 ou 73%.

Utilizando a mesma fórmula, pode-se calcular a similaridade entre a expressão **eBUSCA$_1$** e cada um dos documentos **DOC$_1$** e **DOC$_2$**, representados na Figura 10:

$$sim(\mathrm{DOC}_1, \mathrm{eBUSCA}_1) = 0.45 \quad (45\%)$$

$$sim(DOC_2, eBUSCA_1) = 0.92 \quad (92\%)$$

Portanto, a expressão $eBUSCA_1$ possui um grau de similaridade de 45% com o documento DOC_1 e de 92% com o documento DOC_2.

Os valores da similaridade entre uma expressão de busca e cada um dos documentos do *corpus* são utilizados no ordenamento dos documentos resultantes. Portanto, no modelo vetorial o resultado de uma busca é um conjunto de documentos ordenados pelo grau de similaridade entre cada documento e a expressão de busca. Esse ordenamento permite restringir o resultado a um número máximo de documentos desejados. É possível também restringir a quantidade de documentos recuperados definindo um limite mínimo para o valor da similaridade. Utilizando um limite de 0.5, por exemplo, uma expressão de busca obterá como resultado apenas os documentos cujo valor da similaridade for maior ou igual a 0.5 (50%).

Diferentemente do modelo booleano, o modelo vetorial utiliza pesos tanto para os termos de indexação quanto para os termos da expressão de busca. Esta característica permite o cálculo de um valor numérico que representa a relevância de cada documento em relação à busca.

Uma característica do modelo vetorial é que os termos de indexação são independentes, isto é, não são considerados os relacionamentos existentes entre eles. Embora alguns autores apontem essa característica como uma desvantagem, segundo Baeza-Yates e Ribeiro-Neto (1999, p. 30), não há evidências conclusivas que apontem que tais dependências afetam significativamente o desempenho de um sistema de recuperação de informação. Uma importante limitação do modelo vetorial é não permitir a formulação de buscas booleanas, o que restringe consideravelmente sua flexibilidade.

Um dos maiores méritos do modelo vetorial é a definição de um dos componentes essenciais de qualquer teoria científica: um modelo conceitual. Este modelo serviu como base para o desenvolvimento de uma teoria que alimentou uma grande quantidade de pesquisas e resultou no sistema SMART (Salton, 1971).

4.2.3 O sistema SMART

O projeto SMART (***S**istem for the **Ma**nipulation and **R**etrieval of **T**ext*) teve início em 1961 na Universidade de Harvard e mudou-se para a Universidade de Cornell após 1965. O

sistema SMART é o resultado da vida de pesquisa de Gerard Salton e teve um papel significativo no desenvolvimento de toda a área da Recuperação de Informação. O SMART é uma implementação do modelo vetorial, proposto pelo próprio Salton nos anos 60.

No sistema SMART cada documento é representado por um vetor numérico. O valor de cada elemento desse vetor representa a importância do respectivo termo na descrição do documento. Estes pesos podem ser atribuídos manualmente, o que necessitaria de pessoal especializado trabalhando durante certo tempo. No entanto, o sistema SMART fornece um método automático para o cálculo dos pesos não só dos vetores que representam os documentos, mas também para os vetores das expressões de busca. A forma de calcular esses pesos é descrita por Salton e McGill (1983, p.204-207). Inicialmente define-se a freqüência de um termo ("*term frequency*" - *tf*) como sendo o número de vezes que um determinado termo ***t*** aparece no texto de um documento ***d***.

$$tf_{t,d} = freq_{t,d}$$

Essa medida (*tf*) não faz distinção entre termos que ocorrem em todos o documentos do *corpus* e termos que ocorrem somente em alguns documentos. Sabe-se intuitivamente que um termo que aparece em todos os documentos terá provavelmente pouca utilidade em identificar a relevância dos documentos. Portanto, para um cálculo preciso do peso de um determinado termo de indexação é preciso uma estatística global que caracterize o termo em relação a todo o *corpus*. Esta medida, chamada "*inverse document frequency*" (*idf*), mostra como o termo é distribuído pelo *corpus*, e é calculada da seguinte forma:

$$idf_t = \frac{N}{n_t}$$

onde ***N*** é o número de documentos no *corpus* e $\boldsymbol{n_t}$ é o número de documentos que contém o termo ***t***.

Quanto menor o número documentos que contêm um determinado termo, maior o *idf* desse termo. Se todos os documentos do *corpus* contiverem um determinado termo, o *idf* desse termo será igual a um (1).

Finalmente, o peso de um termo ***t*** em relação a um documento ***d*** ($w_{t,d}$) é definido através da multiplicação da medida *tf* pela medida *idf*. Essa nova medida é conhecida como *tf*idf* e possui a seguinte fórmula:

$$w_{t,d} = tf_{t,d} \times idf_t$$

A medida *tf*idf* é utilizada para atribuir peso a cada elemento dos vetores que representam os documento do *corpus*. Os melhores termos de indexação (os que apresentarão maior peso) são aqueles que ocorrem com uma grande freqüência em poucos documentos.

Assim como os documentos, uma expressão de busca também é representada por um vetor. Isso permite ao usuário atribuir a cada termo da expressão um número que representa a importância relativa do termo para a sua necessidade de informação. Porém, o que aparentemente é um recurso bastante útil, por outro lado pode ser confuso para um usuário inexperiente. Em Salton e Buckley (1988) são descritas algumas formas alternativas para calcular automaticamente os pesos não só para os termos de indexação, mas também dos termos de busca. O peso de cada termo ***t*** de uma expressão de busca **eBUSCA** ($w_{\text{t,eBUSCA}}$) pode ser calculado através da seguinte fórmula:

$$w_{\text{t,eBUSCA}} = \left(0.5 + \frac{tf_{\text{t,d}}}{2} \right) \times idf_t$$

Através da utilização desta fórmula os pesos dos termos utilizados na expressão de busca serão calculados automaticamente, simplificando a tarefa de formular expressões de buscas.

Antes de se atribuir pesos aos temos de indexação dos documentos é necessário definir quais serão esses termos. O sistema SMART, desde a sua concepção, já incorporava algumas ferramentas de análise lingüística para a extração automática de termos de indexação a partir de seu *corpus*. Os primeiros resultados mostraram que algumas técnicas lingüísticas, que inicialmente se acreditava serem essenciais para um bom desempenho do sistema, não se mostraram eficazes na prática. Por esta razão o sistema SMART foi baseado em processos "lingüísticos" mais simples, que eram bem conhecidos na época (Salton e Lesk, 1968; Salton, 1972 e 1973). O processo de indexação do sistema SMART é feito através das seguintes etapas:

1. Identificar e isolar cada palavra do texto do documento ou de sua representação (Resumo, palavras-chave);
2. Eliminar palavras com grande freqüência e pouco valor semântico (*stop-words*) tais como preposições, artigos, etc.;

3. Remover afixos (prefixos e sufixos) das palavras restantes, reduzindo-as ao seu radical (processo conhecido como *stemming*);

4. Incorporar os radicais (termos) aos vetores dos documentos e atribuir-lhes um peso, calculado através da medida *tf*idf;*

Após esse processo, alguns termos podem apresentar pesos com valor muito abaixo da média. Ao invés de simplesmente excluir esses termos, eles são agrupados a outros termos formando termos compostos mais específicos.

No sistema SMART um termo composto é formado pelos radicais de duas ou mais palavras que não fazem parte da lista de *stop words* (*stop list*), seus componentes ocorrem na mesma frase e pelo menos um desses componentes possui freqüência superior a um determinado limite. Um método mais complexo e preciso de se identificar termos compostos considera a distância (número de palavras) e a ocorrência dos componentes do termo no texto. Um termo composto representa de forma mais precisa o assunto tratado pelo documento e, portanto, o peso associado a ele deve ser maior do que o peso médio dos termos simples. O processo de identificação de termos compostos pode ser resumido da seguinte forma:

1. Eliminar *stop words* do texto dos documentos e reduzir cada palavra restante ao seu radical eliminando prefixos e sufixos;

2. Para cada par de radicais verificar a distância entre seus componentes, que não pode ultrapassar um determinado número de palavras. Pelo menos um componente de cada termo composto deve ter uma freqüência relativamente alta.

3. Eliminar termos compostos que possuem termos idênticos;

4. O peso de um termo composto é uma função dos pesos de seus componentes, e deve ser superior ao peso de cada componente tomado isoladamente.

Outra técnica pioneira desenvolvida pelo sistema SMART é a reformulação da expressão de busca do usuário com o propósito de obter melhores resultados na recuperação. Essa reformulação pode ser feita automaticamente ou através da interação do usuário, em um processo conhecido como "*Relevance Feedback"*. Esse processo visa construir uma nova expressão de busca a partir dos documentos identificados como relevantes no conjunto de documentos resultantes de uma busca anterior. No sistema SMART, o processo de reformulação das expressões de busca é baseado nas seguintes operações:

- Termos que ocorrem em documentos identificados como relevantes são adicionados à expressão de busca. Os termos que já fazem parte da expressão de busca têm seus pesos aumentados;
- Termos que ocorrem nos documentos identificados como não relevantes são excluídos da expressão de busca original ou seus pesos são apropriadamente reduzidos

A operação de *relevance feedback* pode ser repetida diversas vezes até que o usuário obtenha um resultado satisfatório para suas necessidades.

O sistema SMART continua sendo uma referência no desenvolvimento de sistemas de recuperação de informação, e ainda é utilizado para pesquisas em ambiente acadêmico. Resultados obtidos por uma grande variedade de testes TREC (*Text Retrieval Conference*) indicam que o sistema SMART ainda consegue um desempenho acima da média em relação a outros sistemas, sob determinadas condições (Buckley et al, 1995).

Os programas-fonte do sistema SMART estão disponíveis gratuitamente na Internet através do servidor FTP da Universidade de Cornell (**ftp://ftp.cs.cornell.edu/pub/smart/**).

4.3 Modelo probabilístico

Na matemática, a teoria das probabilidades estuda os experimentos aleatórios que, repetidos em condições idênticas, podem apresentar resultados diferentes e imprevisíveis. Isso ocorre, por exemplo, quando se observa a face superior de um dado após o seu lançamento ou quando se verifica o naipe de uma carta retirada de um baralho. Por apresentarem resultados imprevisíveis, é possível apenas estimar a possibilidade ou a chance de um determinado evento ocorrer.

Para descrever matematicamente um experimento é necessário inicialmente identificar o conjunto dos possíveis resultados do experimento. No lançamento de um dado, por exemplo, o conjunto dos possíveis resultados é {1, 2, 3, 4, 5, 6}. Esse conjunto é denominado *espaço amostral*.

Durante um determinado experimento pode-se estar interessado em algum aspecto particular ou em alguma situação que esperamos que aconteça. No lançamento de um dado, por exemplo, pode-se estar interessado nos números maiores que 3, isto é, no conjunto {4, 5,

6}. Se o interesse reside nos números pares, o conjunto será {2, 4, 6}. Ao conjunto dos valores de interesse em um determinado experimento dá-se o nome de *evento*. Quando este conjunto é composto por um único elemento é chamado de *evento elementar*.

Considerando um experimento aleatório, a cada evento elementar pode-se associar um valor numérico que expressa a chance ou a probabilidade de que esse evento ocorra. A probabilidade de um evento elementar *E* ocorrer em um espaço amostral *S* é a razão entre o número de elementos de *E*, simbolizado por **n(*E*)** e o número de elementos de *S* (**n(*S*)**).

$$\mathrm{p}(E) = \frac{\mathrm{n}(E)}{\mathrm{n}(S)}$$

No lançamento de um dado o espaço amostral é *S*={1, 2, 3, 4, 5, 6} e a probabilidade de sair o número 5 (*E*= { 5 }) é:

$$\mathrm{p}(5) = \frac{\mathrm{n}(E)}{\mathrm{n}(S)} = \frac{1}{6}$$

A probabilidade de ocorrer um determinado evento somado à probabilidade de não ocorrer tal evento será sempre igual a 1. A probabilidade de sair o número 4 no lançamento de um dado, somado à probabilidade de não sair o número 4 será:

$$\mathrm{p}(4) + \mathrm{p}(\overline{4}) = \frac{1}{6} + \frac{5}{6} = 1$$

Um espaço amostral é chamado equiprovável quando seus eventos elementares têm iguais probabilidades de ocorrência. No lançamento de um dado, por exemplo, o espaço amostral é equiprovável já que a possibilidade de ocorrer cada um de seus números é igual a 1/6.

Um determinado experimento pode ser composto por dois eventos. Esses eventos podem ser dependentes ou independentes. Eventos dependentes são aqueles em que a ocorrência de um influencia na probabilidade da ocorrência de outro. Dois eventos são independentes quando um não interfere no outro.

Considerando dois eventos independentes, a probabilidade de ambos ocorrerem é igual à multiplicação da probabilidade de cada um desses eventos isolados. Por exemplo, jogando-se dois dados, a probabilidade de sair o número 1 em um dos dados **e** o número 6 em outro é:

$$\text{p}(\,1\,\text{e}\,6\,) = \text{p}(1) \times \text{p}(6) = \frac{1}{6} \times \frac{1}{6} = \frac{1}{36} = 0.02777$$

A probabilidade de pelo menos um evento ocorrer é calculado através da soma da probabilidade de cada evento isolado. Jogando-se dois dados, qual a probabilidade de sair o número 1 em um dado **ou** o número 6 em outro?

$$\text{p}(\,1\,\text{ou}\,6\,) = \text{p}(1) + \text{p}(6) = \frac{1}{6} + \frac{1}{6} = \frac{2}{6} = 0.33333$$

Quando dois eventos se mostram dependentes, o cálculo da probabilidade envolve as chamadas Probabilidades Condicionais. A probabilidade da ocorrência de um evento *A*, sabendo-se que o evento *B* ocorreu, é calculada como:

$$\text{p}(\,A \mid B) = \frac{\text{p}(A\,\text{e}\,B)}{\text{p}(B)}$$

Por exemplo, uma pesquisa para provar a relação entre o tabagismo e o câncer de pulmão foi realizada com duzentas e trinta pessoas. Os resultados obtidos foram os seguintes:

	com câncer	**sem câncer**	
fumante	70	8	78
não fumante	20	132	152
	90	140	230

De acordo com essa tabela, se uma pessoa é fumante ela necessariamente terá mais chances de ter câncer do que uma pessoa não fumante? Para responder questões como essas se utiliza o conceito de probabilidade condicional. Estamos interessados em duas sub-populações:

A={ pessoas que são fumantes }

B={ pessoas com câncer de pulmão }

A probabilidade que uma pessoa selecionada ao acaso da sub-população *B* (fumante) estar também em *A* (câncer) é calculada como:

$$\text{p}(A \mid B) = \frac{\text{p}(A\,\text{e}\,B)}{\text{p}(B)} = \frac{\frac{70}{230}}{\frac{90}{230}} = \frac{70}{90} = 0.7777 \;\text{ ou }\; 77.77\%$$

Portanto, a probabilidade de uma pessoa ser fumante, sabendo-se que ela tem câncer no pulmão é de 77.77%.

Porém, a questão que um fumante desejaria fazer é: Qual a probabilidade de um fumante ter câncer de pulmão? Isto é, o que lhe interessa é o valor de p($B|A$). Tendo-se o valor de p($A|B$), uma das maneiras de achar a probabilidade p($B|A$) é utilizar o teorema de Bayes, apresentado abaixo.

$$\mathrm{p}(B \mid A) = \frac{\mathrm{p}(A \mid B) \times \mathrm{p}(B)}{\mathrm{p}(A)}$$

No exemplo, o cálculo da probabilidade de p($B|A$) será:

$$\mathrm{p}(B \mid A) = \frac{\mathrm{p}(A \mid B) \times \mathrm{p}(B)}{\mathrm{p}(A)} = \frac{0.7777 \times \frac{90}{230}}{\frac{78}{230}} = \frac{0.3043}{0.3391} = 0.8974 \text{ ou } 89.74\%$$

Portanto, na população utilizada no experimento, a probabilidade de um fumante ter câncer é de 89.74%.

4.3.1 Recuperação probabilística

O modelo probabilístico proposto por Robertson e Jones (1976), posteriormente conhecido como *Binary Indepence Retrieval*, tenta representar o processo de recuperação de informação sob um ponto de vista probabilístico.

Dada uma expressão de busca, pode-se dividir o *corpus* (com ***N*** documentos) em quatro subconjuntos distintos (Figura 11): o conjunto dos documentos relevantes (***Rel***), o conjunto dos documentos recuperados (***Rec***), o conjunto dos documentos relevantes que foram recuperados (***RR***) e o conjunto dos documentos não relevantes e não recuperados. O conjunto dos documentos relevantes e recuperados (*RR*) é resultante da interseção dos conjuntos *Rel* e *Rec*.

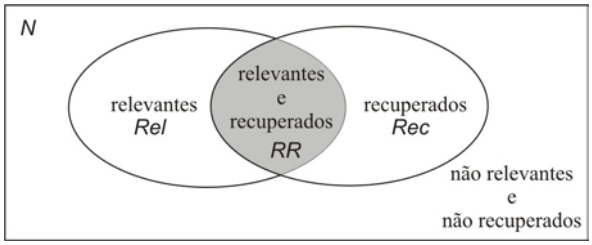


Figura 11 Subconjuntos de documentos após a execução de uma busca

O resultado ideal de uma busca é o conjunto que contenha todos e apenas os documentos relevantes para o usuário, isto é, todo o conjunto ***Rel***. Se o usuário soubesse exatamente o que distingue os documentos desse conjunto dos demais documentos do *corpus* seria fácil recuperá-los. No entanto, como as características dos documentos não são conhecidas, tenta-se adivinhar tais características através da formulação de uma expressão de busca, gerando uma primeira descrição probabilística desse conjunto. Com os resultados obtidos após a execução da primeira busca é possível gradativamente melhorar os resultados através de interações com o usuário.

Seja *Rel* o conjunto de documentos relevantes e $\overline{Rel}$ o complemento de *Rel*, ou seja, o conjunto dos documentos não relevantes. A probabilidade de um documento ***d*** ser relevante em relação à expressão de busca é designada por p(*Rel*|*d*). A probabilidade de um documento ser considerado não relevante é representada por $\mathrm{p}(\overline{Rel} \mid d)$. A similaridade (***sim***) de um documento ***d*** em relação à expressão de busca ***eBUSCA*** é definida como:

$$sim(d, eBUSCA) = \frac{\mathrm{p}(Rel \mid d)}{\mathrm{p}(\overline{Rel} \mid d)}$$

Usando a função de Bayes obtém-se a seguinte expressão:

$$sim(d, eBUSCA) = \frac{\mathrm{p}(d \mid Rel) \times \mathrm{p}(Rel)}{\mathrm{p}(d \mid \overline{Rel}) \times \mathrm{p}(\overline{Rel})}$$

A expressão p(*d*|*Rel*) representa a probabilidade de se selecionar o documento *d* do conjunto de documentos relevantes *Rel* e $\mathrm{p}(d \mid \overline{Rel})$ representa a probabilidade de se selecionar o documento *d* do conjunto dos documentos não relevantes. A expressão p(*Rel*) representa a probabilidade de um documento selecionado aleatoriamente ser relevante, enquanto $\mathrm{p}(\overline{Rel})$ representa a probabilidade de um documento não ser relevante.

Considerando p(*Rel*) e $p(\overline{Rel})$ iguais para todos os documentos do *corpus*, a fórmula da similaridade pode então ser escrita como:

$$sim(d, eBUSCA) \approx \frac{p(d \mid Rel)}{p(d \mid \overline{Rel})}$$

Um documento é representado por um vetor binário cuja presença e a ausência de um determinado termo de indexação (t_i) é designado respectivamente por 1 ou 0.

	t_1	t_2	t_3	…	t_n
DOC	w_1	w_2	w_3	…	w_n

onde w_i pode assumir o valor zero ou um, indicando a ausência ou a presença do termo de indexação $\boldsymbol{t}_i$ no conjunto dos indexadores do documento DOC.

A probabilidade de um termo t_i estar presente em um documento selecionado do conjunto *Rel* é representado por $p(t_i \mid Rel)$ e $p(\overline{t_i} \mid Rel)$ é a probabilidade do termo t_i não estar presente em um documento selecionado de *Rel*. Lembrando que $p(t_i \mid Rel) + p(\overline{t_i} \mid Rel) = 1$, e ignorando fatores que são constantes para todos os documentos no contexto de uma mesma busca, tem-se finalmente:

$$sim(d, eBUSCA) \approx \sum_{i=1}^{t} \left(\log \frac{p(t_i \mid Rel) \times p(\overline{t_i} \mid \overline{Rel})}{p(t_i \mid \overline{Rel}) \times p(\overline{t_i} \mid Rel)} \right) \qquad \textbf{[1]}$$

Esta expressão é fundamental para ordenar os documentos no modelo probabilístico.

Todo cálculo de probabilidade resume-se a um problema de contagem. Portanto, para uma determinada expressão de busca, pode-se representar os documentos do *corpus* da seguinte forma:

	Relevante	**não-Relevante**	
documento contendo t_i	r	n-r	n
documento que não contém t_i	R-r	N-R-n+r	N-n
	R	N-R	N

Considerando um *corpus* com **N** documentos e um determinado termo $\boldsymbol{t}_i$, existe no *corpus* um total de **n** documentos indexados por $\boldsymbol{t}_i$. Desses **n** documentos apenas **r** são relevantes.

A fórmula de similaridade (equação **[1]**) pode ser traduzida com base na tabela acima, considerando as seguintes igualdades:

$$p(t_i \mid Rel) = r \qquad p(\overline{t_i} \mid \overline{Rel}) = N - R - n + r$$
$$p(t_i \mid \overline{Rel}) = n - r \qquad p(\overline{t_i} \mid Rel) = R - r$$

$$sim(d, eBUSCA) \approx \sum_{i=1}^{t} \left(\log \frac{r \times (N - R - n + r)}{(n - r) \times (R - r)} \right)$$

No início do processo de busca não se sabe qual o conjunto de documentos relevantes (R), já que nenhum documento foi ainda recuperado. Portanto, antes da primeira busca é necessário fazer algumas simplificações, tais como: (a) assumir que $p(t_i \mid Rel)$ é constante e igual a 0.5 para todos os temos t_i e (b) assumir que a distribuição dos termos de indexação dos documentos (relevantes ou não) é uniforme. Assim, obtém-se a seguinte fórmula:

$$sim(d, eBUSCA) \approx \sum_{i=1}^{t} \left(\log \frac{N - n}{n} \right) \qquad \textbf{[2]}$$

Através dessa fórmula é ordenado o conjunto de documentos resultantes da primeira busca. Tendo esse conjunto de documentos, o usuário seleciona alguns documentos que considera relevantes para a sua necessidade. O sistema então utiliza esta informação para tentar melhorar os resultados subseqüentes.

Para exemplificar, será considerado um *corpus* contendo 6 documentos e 10 termos de indexação:

	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
DOC$_1$	**1**	0	0	**1**	0	0	0	**1**	**1**	0
DOC$_2$	0	0	0	0	0	0	0	**1**	**1**	**1**
DOC$_3$	0	**1**	0	0	0	**1**	**1**	0	0	0
DOC$_4$	**1**	0	0	**1**	0	0	0	0	0	**1**
DOC$_5$	0	0	0	0	0	0	0	**1**	**1**	0
DOC$_6$	0	0	**1**	0	**1**	0	0	0	0	0

A expressão de busca (eBUSCA) será composta pelos termos t_4 e t_{10} sendo representada pelo seguinte vetor:

	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
eBUSCA	0	0	0	**1**	0	0	0	0	0	**1**

Após a execução da primeira busca os documentos recuperados serão apresentados em ordem do valor resultante da equação **[2]** aplicada a cada documento. Alguns documentos,

como no caso dos documentos 3, 5 e 6, não são recuperados pois apresentaram valor menor ou igual a zero.

	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}	***sim*(DOC$_i$, eBUSCA)**
DOC$_4$	**1**	0	0	**1**	0	0	0	0	0	**1**	0.51
DOC$_1$	**1**	0	0	**1**	0	0	0	**1**	**1**	0	0.26
DOC$_2$	0	0	0	0	0	0	0	**1**	**1**	**1**	0.26

Com esse primeiro resultado o usuário poderá selecionar alguns documentos que são úteis para a sua necessidade. No exemplo apenas três documentos resultaram da primeira busca. Porém, se uma busca resultar uma quantidade muito grande de documentos basta selecionar alguns poucos documentos que considerasse relevante. No exemplo, o documento DOC_1, apesar de ter o mesmo grau de similaridade (***sim***) do documento DOC_2 ele não foi considerado relevante pelo usuário. Após submeter novamente a expressão de busca, juntamente com os documentos selecionados, o sistema calculará para cada documento um valor da similaridade utilizando a equação **[1]**. Esse valor será utilizado para ordenar o conjunto de documentos recuperados:

	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}	***sim*(DOC$_i$, eBUSCA)**
DOC$_4$	**1**	0	0	**1**	0	0	0	0	0	**1**	2.02
DOC$_2$	0	0	0	0	0	0	0	**1**	**1**	**1**	1.65
DOC$_1$	**1**	0	0	**1**	0	0	0	**1**	**1**	0	0.37

Com a repetição desse processo espera-se uma melhora progressiva nos resultados da busca. O usuário poderá repetir esse processo de seleção dos documentos relevantes até que o conjunto de documentos recuperados satisfaça sua necessidade de informação.

A principal virtude do modelo probabilístico está em reconhecer que a atribuição de relevância é uma tarefa do usuário. É o único modelo que incorpora explicitamente o processo de *Relevance Feedback* como base para a sua operacionalização.

É importante observar que o modelo probabilístico pode ser facilmente implementado utilizando a estrutura proposta pelo modelo vetorial, permitindo integrar as vantagens desses dois modelos em um sistema de recuperação de informação.

Embora o modelo probabilístico tenha um forte embasamento teórico, as hipóteses assumidas para realizar simplificações nos cálculos probabilísticos podem deixar dúvidas sobre sua precisão. Uma simplificação bastante questionável está no fato de o modelo

considerar os pesos dos termos de indexação como sendo binários, isto é, no modelo probabilístico não é considerada a freqüência com que os termos ocorrem no texto dos documentos.

Alguns experimentos utilizando poucos documentos demonstram que este modelo produz resultados pouco superiores em relação ao modelo booleano. Pode ser que no contexto heterogêneo e complexo da Web os métodos probabilísticos venham a se destacar. Porém, a sua complexidade desencoraja muitos desenvolvedores de sistema a abandonar os modelos booleano e vetorial (Jones, Walker e Robertson, 2000).

4.4 Modelo *fuzzy*

A lógica aristotélica é uma forte presença na cultura ocidental e está profundamente enraizada em nossa forma de pensar. Uma determinada afirmação é verdadeira ou falsa; uma pessoa ou é amiga ou inimiga. Na ciência a verdade e a precisão estão intimamente ligadas e são partes indispensáveis do método científico. Se algo não é absolutamente correto então não é verdade.

Porém, observa-se um considerável descompasso entre a realidade e a nossa visão bivalente do mundo. O mundo real contém uma infinidade de gradações entre o preto e o branco, entre o certo e o errado, entre o verdadeiro e o falso. O mundo real é multivalente e analógico. Verdade e precisão absolutas existem apenas em casos extremos.

A comunicação humana é vaga e imprecisa, contendo diversas incertezas. Quando se diz que uma determinada "pessoa é alta", o que se está querendo dizer precisamente: 170 cm, 180 cm, 190 cm? Se fosse definido um limite de altura de 180 cm, por exemplo, então uma pessoa com 179 cm não seria considerada alta. Intuitivamente sabemos que não há uma distinção clara entre uma pessoa de 179 cm de altura e uma de 180 cm. Quando os seres humanos pensam em altura eles normalmente não têm um limite fixo em mente, mas uma definição nebulosa, vaga.

O objetivo da lógica *fuzzy* é capturar e operar com a diversidade, a incerteza e as verdades parciais dos fenômenos da natureza de uma forma sistemática e rigorosa (Shaw e Simões, 1999).

4.4.1 Conjuntos *fuzzy*

Zadeh (1965) propôs uma nova teoria de conjuntos em que não há descontinuidades, ou seja, não há uma distinção abrupta entre elementos pertencentes e não pertencentes a um conjunto: os Conjuntos Nebulosos (*Fuzzy Sets*).

Na teoria matemática dos conjuntos, para indicar que um elemento ***x*** pertence a um conjunto ***A***, utiliza-se a expressão $\boldsymbol{x} \in \boldsymbol{A}$. Poderia-se também utilizar a função $\mu_A(x)$, cujo valor indica se o elemento ***x*** pertence ou não ao conjunto ***A*** . Neste caso $\mu_A(x)$ é uma função bivalente que somente resulta 1 (um) ou zero, dependendo se o elemento ***x*** pertence ou não ao conjunto ***A***:

$$\mu_A(x) = \begin{cases} 1 \text{ se } x \in A \\ 0 \text{ se } x \notin A \end{cases}$$

Na Figura 12 observa-se que, se o elemento x_2 for movido em direção ao elemento x_1, no limite do conjunto A ocorrerá subitamente uma alteração de seu estado, passando de não-membro para membro do conjunto.

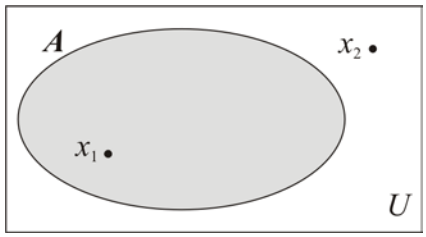


Figura 12 Pertinência de um elemento em relação a um conjunto

Na lógica *fuzzy* um elemento pode ser membro de um conjunto apenas parcialmente. Um valor entre zero e um (1) indicará o quanto o elemento é membro do conjunto.

A teoria dos conjuntos *fuzzy* é baseada no fato de que os conjuntos existentes no mundo real não possuem limites precisos. Um conjunto *fuzzy* é um agrupamento indefinido de elementos no qual a transição de cada elemento de não-membro para membro do conjunto é gradual. Esse grau de imprecisão de um elemento pode ser visto como uma "medida de possibilidade", ou seja, a "possibilidade" de que um elemento seja membro do conjunto.

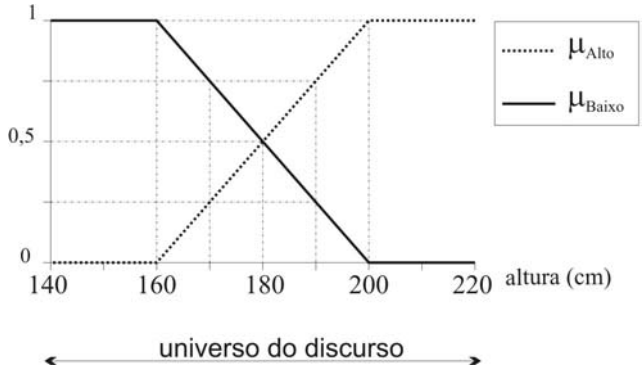


Figura 13 Representação das funções μ_{alto} e μ_{baixo}

No exemplo da Figura 13 o conjunto dos diversos valores das alturas de uma pessoa é denominado *universo do discurso*. Todo conjunto *fuzzy* é na realidade um subconjunto do universo do discurso. Um subconjunto A do universo do discurso U é caracterizado por uma função μ_A que associa a cada elemento x de U um número $\mu_A(x)$ entre 0 e 1. Assim, temos:

$$A = \{x, \mu_A(x)\} \mid x \in U$$

onde $\mu_A(x)$ resulta um valor numérico entre zero e um que representa o quando o elemento x pertence ao conjunto A.

Vejamos um exemplo: supondo que A seja o conjunto de pessoas altas e $\boldsymbol{x}_1$ e $\boldsymbol{x}_2$ representam duas pessoas com 190 cm e 170 cm de altura, respectivamente. O subconjunto A será caracterizado pela função $\mu_A(x)$, que associa a cada elemento x_1 e x_2 do universo do discurso (U) um número, respectivamente $\mu_A(x_1)$ e $\mu_A(x_2)$. No gráfico da Figura 13 teremos $\mu_A(x_1)$ igual a 0,75 ou 75%, e $\mu_A(x_2)$ igual a 0,25 ou 25%. Portanto, no exemplo, uma pessoa com 190cm é 75% alta e uma pessoa com 170cm é apenas 25% alta. Ou seja, em um conjunto *fuzzy* um mesmo objeto pode pertencer a dois ou mais conjuntos com diferentes graus. Uma pessoa com 190 cm pertence 75% ao conjunto das pessoas altas ao mesmo tempo em que pertence 25% ao conjunto das pessoas baixas. Uma pessoa que mede 180 cm é simultaneamente 50% alta e 50% baixa ($\mu_{alta}(180)=\mu_{baixa}(180)=0.5$).

As operações mais utilizadas nos conjuntos *fuzzy* são: complemento, união e interseção e são definidas como segue:

Complemento: $$\mu_{\overline{A}}(u) = 1 - \mu_A(u)$$

União: $$\mu_{A \cup B}(u) = \max(\mu_A(u), \mu_B(u))$$

Inserseção: $$\mu_{A \cap B}(u) = \min(\mu_A(u), \mu_B(u))$$

Utilizando a Figura 13, essas operações são exemplificadas abaixo:

$$\mu_{\overline{baixo}}(170) = 1 - \mu_{baixo}(170) = 1 - 0.75 = \mathbf{0.25}$$

$$\mu_{baixo \cup alto}(170) = \max(\mu_{baixo}(170), \mu_{alto}(170)) = \max(0.75, 0{,}25) = \mathbf{0.75}$$

$$\mu_{baixo \cap alto}(170) = \min(\mu_{baixo}(170), \mu_{alto}(170)) = \min(0.75, 0.25) = \mathbf{0.25}$$

A teoria *fuzzy* possibilita a definição de classes de elementos em situações onde não é possível uma delimitação precisa e natural de suas fronteiras. Este ambiente teórico é capaz de representar de forma mais eficiente a inerente imprecisão das entidades envolvidas em um sistema de recuperação de informação, muito embora seja conflitante com a teoria clássica dos sistemas de classificação, segundo a qual as classes devem ser auto-excludentes.

4.4.2 Conjuntos fuzzy na recuperação de informação

Um documento pode ser visto como um conjunto *fuzzy* de termos, { $\mu(t)$/t }, cujos pesos dependem do documento e do termo em questão, isto é: $\mu(t)=F(d,t)$. Portanto, a representação *fuzzy* de um documento é baseada na definição de uma função $F(d, t)$ que produz um valor numérico que representa o peso do termo t para o documento d.

O peso associado a um termo expressa o quanto esse termo é significativo na descrição do conteúdo do documento. A qualidade da recuperação depende em grande parte da função adotada para calcular os pesos dos termos de indexação (Salton e Buckley, 1988). Geralmente esta função baseia-se no cálculo da freqüência de ocorrência dos termos em todo o texto, e fornece uma representação estática do documento. O cálculo dos pesos não considera que em muitos casos os documentos podem estar estruturados em sub-partes lógicas ou seções, e que as ocorrências de um termo podem assumir significados diferentes dependendo da seção onde ele aparece. Um artigo científico, por exemplo, geralmente está organizado em *título*, *autores*, *palavras-chave*, *resumo*, *referências*, etc. Uma única ocorrência de um termo no título sugere que o artigo discorre sobre o conceito expresso pelo termo. As seções de um documento podem assumir diferentes graus de importância dependendo da necessidade do usuário. Quando, por exemplo, o usuário está procurando artigos escritos por uma determinada pessoa, a parte mais importante a ser analisada é a seção de autores. Quando se procura artigos de um determinado assunto, o título, as palavras-chaves, o resumo e a introdução assumem maior importância.

Bordogna e Pasi (1995) propõem uma representação *fuzzy* para documentos estruturados que pode ser ajustada de acordo com os interesses do usuário. A importância de um termo *t* em um documento *d* é calculada pela avaliação da importância de *t* em cada uma das seções de *d*. Isto é feito através da aplicação de uma função $F_{Si}(d, t)$ que expressa o grau de pertinência do termo *t* na seção S_i do documento *d*, como ilustrado na Figura 14.

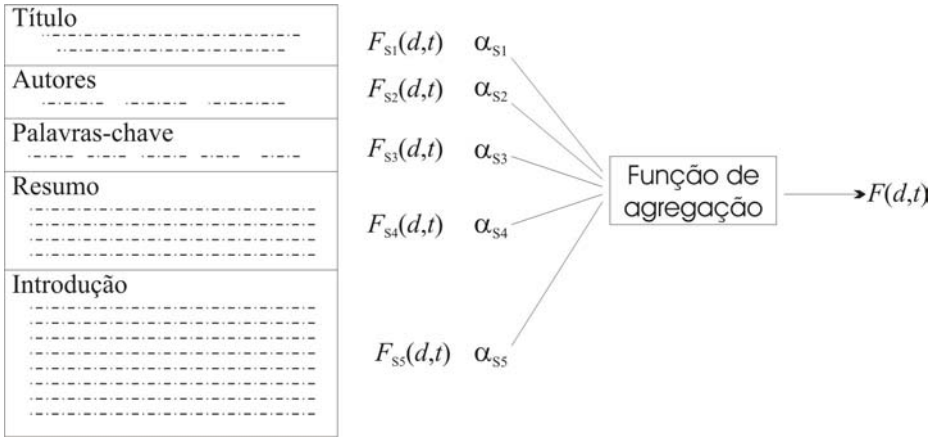


Figura 14 Representação *fuzzy* de um documento estruturado

Para cada seção S_i o usuário pode associar uma importância numérica α_{Si} que será usada para enfatizar a função $F_{Si}(t,d)$. Para se obter um grau de pertinência de um termo em relação a um documento os graus de pertinência do termo em cada uma das seções $F_{S1}(d,t)$, $F_{S2}(d,t)$,...$F_{Sn}(d,t)$ são agregados por meio de uma função, que pode ser selecionada pelo usuário entre um conjunto pré-definido de "quantificadores lingüísticos" tais como *all*, *least one*, *at least about k*, *all* (Yager, 1988). O quantificador lingüístico indica o número de seções em que um termo deve aparecer para que o documento seja considerado relevante. Esta representação *fuzzy* de documentos foi implementada em um sistema denominado DOMINO (Bordogna et al, 1990) e mostrou ser mais eficaz em relação a outros tipos de representação *fuzzy*.

Utilizando idéia semelhante, Molinari e Pasi (1996) propõem um método de indexação de documentos HTML baseado na estrutura sintática dessa linguagem de marcação. Para cada seção de um documento HTML, delimitada pelas marcações (*tags*), é associado um grau de importância. Pode-se supor, por exemplo, que quanto maior o tamanho dos caracteres de um trecho do texto maior a importância atribuída a esse trecho. Da mesma forma, uma palavra em negrito ou itálico geralmente representa um destaque dado pelo autor da página HTML para uma palavra. Assim, para cada *tag* pode ser associado um valor numérico que expressa a sua

importância para o documento. O peso de um termo em relação a um determinado documento é obtido através de uma função de agregação que considera a importância de cada *tag* do documento onde o termo aparece.

O modelo *fuzzy* tem sido discutido principalmente na literatura dedicada à teoria *fuzzy*, não sendo popular entre a comunidade da recuperação de informação. Além disso, a grande maioria dos experimentos realizados com este modelo considera apenas pequenos *corpora*, que não comprovam sua efetiva superioridade em relação a outros modelos de recuperação de informação (Baeza-Yates e Ribeiro-Neto, 1999, p. 38).

4.5 Modelo booleano estendido

No modelo booleano uma expressão de busca composta por termos conectados por operadores OR (t_1 or t_2 or ... or t_n) recuperará documentos indexados por pelo menos um destes termos. Um documento indexado por todos os termos é tratado da mesma forma que um documento indexado por apenas um dos termos. Em uma expressão composta por dez termos conectados por operadores AND (t_1 and t_2 and ... and t_{10}), um documento indexado por nove desses termos é visto da mesma maneira que um documento que não é indexado por nenhum deles. Este julgamento binário, inerente ao modelo booleano, não está de acordo com o senso comum. Intuitivamente sabe-se que após uma busca utilizando uma expressão booleana conjuntiva ($\boldsymbol{t_1}$ **and** $\boldsymbol{t_2}$), os documentos indexados por apenas um dos termos da expressão, que não foram recuperados, possuem um certo grau de importância e poderiam vir a ser considerados relevantes por um usuário. Utilizando uma expressão disjuntiva ($\boldsymbol{t_1}$ **or** $\boldsymbol{t_2}$) um documento indexado por ambos os termos da expressão pode ser considerado mais importante do que os documentos indexados por apenas um dos termos.

O modelo booleano estendido, proposto por Salton, Fox e Wu (1983), tenta unir a potencialidade das expressões booleanas com a precisão do modelo vetorial. Por um lado busca-se flexibilizar o modelo booleano, introduzindo uma gradação no conceito de relevância e, por outro lado, dar maior poder às buscas do modelo vetorial através do uso dos operadores booleanos.

Utilizando-se dois termos (t_1 e t_2) para representar expressões de busca e documentos, define-se um espaço bidimensional onde cada termo é associado a um eixo, como mostrado na Figura 15. Um documento é representado por um vetor com dois elementos contendo o

peso dos respectivos termos. Estes pesos definem o posicionamento do documento nesse espaço.

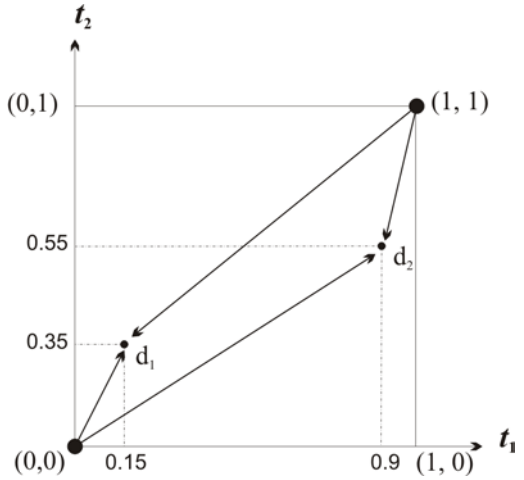


Figura 15 Representação de documentos em um espaço bidimensional

Em expressões disjuntivas o ponto (0, 0) deve ser evitado pois representa a situação na qual nenhum dos termos está presente no documento. Assim, a distância de um documento ao ponto (0,0) é considerado o grau de relevância ou a similaridade do documento em relação à busca. Quanto maior a distância de um documento em relação a este ponto, maior será sua similaridade em relação à expressão de busca.

A similaridade entre um documento DOC=(w_{t1}, w_{t2}) e uma expressão de busca **eBUSCA= t_1 or t_2** é calculada através da seguinte fórmula:

$$sim(\text{DOC}, \text{eBUSCA}_{\text{t1 or t2}}) = \sqrt{\frac{w_{t1}^2 + w_{t2}^2}{2}}$$

onde w_{t1} e w_{t2} representam os pesos de cada um dos termos de indexação do documento DOC.

A similaridade entre uma expressão disjuntiva **eBUSCA= t_1 or t_2** e o documento d_1=(0.15, 0.35), representado na Figura 15, é calculada da seguinte forma:

$$sim(\text{d}_1, \text{eBUSCA}_{\text{t1 or t2}}) = \sqrt{\frac{0.15^2 + 0.35^2}{2}} = 0.2692$$

Para o documento d_2=(0.9, 0.55) o valor da similaridade é:

$$sim(\text{d}_2, \text{eBUSCA}_{\text{t1 or t2}}) = \sqrt{\frac{0.9^2 + 0.55^2}{2}} = 0.7458$$

Para expressões conjuntivas o ponto (1, 1) é o mais desejável, já que representa a situação na qual ambos os termos da expressão estão presentes na representação de um documento. Quanto menor a distância de um documento em relação a este ponto maior sua similaridade em relação à expressão de busca.

A similaridade entre um documento DOC=(w_{t1}, w_{t2}) e uma expressão conjuntiva **eBUSCA=t_1 and t_2** é calculada como:

$$sim(\text{DOC}, \text{eBUSCA}_{\text{t1 and t2}}) = 1 - \sqrt{\frac{(1-w_{\text{t1}})^2 + (1-w_{\text{t2}})^2}{2}}$$

A similaridade entre uma expressão de busca **eBUSCA= t_1 and t_2** e o documento d_1 da Figura 15 é:

$$sim(\text{d}_1, \text{eBUSCA}_{\text{t1 and t2}}) = 1 - \sqrt{\frac{(1-0.15)^2 + (1-0.35)^2}{2}} = 0{,}2434$$

Para o documento d_2 o valor da similaridade é:

$$sim(\text{d}_2, \text{eBUSCA}_{\text{t1 and t2}}) = 1 - \sqrt{\frac{(1-0.9)^2 + (1-0.55)^2}{2}} = 0.6740$$

Para tornar o modelo mais flexível, utiliza-se o conceito matemático de norma L_p, em que a norma de um vetor $\vec{v} = (v_1 + v_2 + ... + v_n)$ é calculada como:

$$\left\| \vec{v} \right\|_p = (v_1^p + v_1^p + ... + v_1^p)^{1/p}$$

A similaridade entre um documento e uma expressão de busca continua sendo uma função da distância entre dois pontos. Porém, ao invés de ser utilizar a distância euclidiana, utiliza-se a norma L_p. Assim, as fórmulas de similaridade entre uma expressão de busca eBUSCA e um documento DOC podem ser representadas da seguinte forma:

$$sim(\text{DOC}, \text{eBUSCA}_{\text{t1 or t2}}) = \left(\frac{w_{\text{t1}}^p + w_{\text{t2}}^p}{2} \right)^{1/p}$$

$$sim(\text{DOC}, \text{eBUSCA}_{\text{t1 and t2}}) = 1 - \left(\frac{(1-w_{\text{t1}})^p + (1-w_{\text{t2}})^p}{2} \right)^{1/p}$$

Pode-se agora generalizar estas fórmulas para considerar não apenas dois termos, mas um número ***n*** de termos. Serão considerados também os pesos dos termos da expressão, como no modelo vetorial. Assim, para uma expressão disjuntiva a fórmula da similaridade será:

$$sim(\text{DOC}, \text{eBUSCA}_{\text{or}(p)}) = \left(\frac{w_{1c}^p w_{1d}^p + w_{2c}^p w_{2d}^p + \ldots + w_{nc}^p w_{nd}^p}{w_{1c}^p + w_{2c}^p + \ldots + w_{nc}^p} \right)^{1/p}$$

onde w_{ic} é o peso atribuído ao ***i***-ésimo termo da expressão eBUSCA e w_{id} é o peso atribuído ao ***i***-ésimo termo de indexação do documento DOC. O parâmetro *p* é definido durante a formulação da expressão de busca.

Para expressões conjuntivas, a similaridade é dada por:

$$sim(\text{DOC}, \text{eBUSCA}_{and(p)}) = 1 - \left(\frac{w_{1c}^p (1 - w_{1d})^p + w_{2c}^p (1 - w_{2d})^p + \ldots + w_{nc}^p (1 - w_{nd})^p}{w_{1c}^p + w_{2c}^p + \ldots w_{nc}^p} \right)^{1/p}$$

O valor do parâmetro *p* determina a interpretação dos operadores booleanos. Os valores de *p* e a sua correspondente interpretação são apresentados a seguir (Salton, 1984):

- Quando *p* é igual a 1 os resultados das expressões disjuntivas e conjuntivas são idênticos, isto é, não há distinção entre **or** ou **and**, e o resultado é semelhante ao obtido no modelo vetorial.
- Quando o valor *p* é bastante alto, ou "tende a infinito" (∞) os resultados são compatíveis com os produzidos pelas expressões booleanas convencionais. De uma forma simplificada, a similaridade de expressões disjuntivas pode ser calculada como:

  $$sim(\text{DOC}, \text{eBUSCA}_{\text{or}(\infty)}) = \max(w_1, w_2, w_3, \ldots).$$

  Isto é, a similaridade de um documento em relação à expressão é igual ao maior peso associado aos termos que representam o documento.

  Para expressões conjuntivas a similaridade pode ser calculada como:

  $$sim(\text{DOC}, \text{eBUSCA}_{\text{and}(\infty)} = \min(w_1. w_2, w_3, \ldots)$$

  Isto é, a similaridade do documento DOC em relação à expressão de busca eBUSCA é igual ao menor peso associado aos termos de indexação do documento.

- Quando p está entre ∞ e 1, os resultados produzidos são intermediários entre uma busca booleana pura e uma busca do modelo vetorial.

Valores de p associados aos operadores booleanos refletem o grau de importância ou o rigor atribuído ao operador correspondente. Quanto menor o valor de p menos estrita será a interpretação do operador. Com um aumento no valor de p aumenta-se a rigidez do operador, aproximando-o do modelo booleano puro. Uma expressão de busca cujos termos possuem pesos e cada operador booleano possui um valor de p pode ser exemplificada como segue:

$$\boldsymbol{t_1}\ (0.3)\ \ \mathbf{and}^2\ \ \boldsymbol{t_2}\ (0.7)\ \ \mathbf{or}^{1.5}\ \ \boldsymbol{t_3}\ (0.4)$$

O cálculo da similaridade para uma expressão de busca composta de operações disjuntivas e conjuntivas é feito através do cálculo da similaridade de partes da expressão.

Para o exemplo apresentado a seguir, será considerado um *corpus* contendo três documentos indexados por três termos com seus respectivos pesos, como apresentado abaixo.

	information	***retrieval***	***document***
DOC_1	0.8	0.2	0.4
DOC_2	0.5	0.4	0.2
DOC3	0.4	0.6	0.0

Será considerada a seguinte expressão de busca:

$$\text{eBUSCA}=(\ \boldsymbol{information}\ (0.6)\ \ \mathbf{OR}^2\ \ \boldsymbol{document}\ (0.3)\ )_{(0.7)}\ \ \mathbf{AND}^{1.5}\ \ \boldsymbol{retrieval}\ (0.5)$$

Para calcular a similaridade desta expressão em relação ao documento DOC_1 será isolada a operação **OR** que aparece entre parênteses. Essa parte da expressão será designada B_1.

$$B_1 = (\ \boldsymbol{information}\ (0.6)\ \ \mathbf{OR}^2\ \ \boldsymbol{document}\ (0.3)\ )$$

$$sim(\mathrm{DOC}_1, \mathrm{B}_1) = \left(\frac{(0.6 \times 0.8)^2 + (0.3 \times 0.4)^2}{0.6^2 + 0.3^2} \right)^{1/2} = 0{,}7376$$

Utilizando o valor da similaridade entre B_1 e DOC_1 ($sim(B_1, DOC_1)$), o enunciado da expressão de busca de busca pode ser representado e calculado da seguinte forma:

$$\text{eBUSCA} = \boldsymbol{sim(DOC_1, B_1)}\ (0.7)\ \ \mathbf{AND}^{1.5}\ \ \boldsymbol{retrieval}\ (0.5)$$

$$sim(\mathrm{DOC}_1, \text{eBUSCA}) = \left(\frac{0.7^{1.5} \times (1 - 0.7376)^{1.5} + 0.5^{1.5} \times (1 - 0.2)^{1.5}}{0.7^{1.5} + 0.5^{1.5}} \right)^{1/1.5} = 0{,}50$$

Utilizando-se o mesmo cálculo para os demais documentos e ordenando os documentos em ordem decrescente do valor da similaridade, o resultado da expressão de busca (eBUSCA) seria a seguinte lista de documentos:

DOC_3	0,5077
DOC_1	0.50
DOC_2	0,4346

Uma das funções de um sistema de recuperação de informação é apresentar os documentos resultantes de forma que os usuários sejam capazes de verificar facilmente sua pertinência. Embora o modelo booleano possua a vantagem de ser de fácil implementação e permitir uma recuperação relativamente eficiente, ele não possibilita o ordenamento dos documentos recuperados. O modelo vetorial, apesar de permitir a ordenação dos documentos resultantes de forma bastante precisa, não possibilita a utilização de buscas booleanas, o que restringe sua capacidade.

O modelo booleano estendido tenta contornar as limitações do modelo vetorial e do modelo booleano clássico através de uma conceituação matemática mais genérica. As expressões booleanas e as buscas do modelo vetorial são casos particulares do modelo booleano estendido. Essa generalização é feita através da introdução de dois novos parâmetros em relação ao modelo booleano tradicional: os pesos associados aos termos da expressão de busca e o parâmetro *p* associado a cada operador booleano. Esse aumento da complexidade na formulação de buscas é a principal desvantagem do modelo booleano estendido.

O modelo booleano estendido nunca foi utilizado extensivamente. Para Baeza-Yates e Ribeiro-Neto (1999, p. 41) este modelo fornece um ambiente “elegante” que poderia ser útil no futuro.

4.6 Conclusão

O processo de recuperação de informação é inerentemente impreciso devido a fatores que talvez nunca serão totalmente equacionados. A modelagem matemática desse processo só é possível através de simplificações teóricas e da adequação de conceitos tipicamente subjetivos como “informação” e “relevância”. Estas simplificações refletem em limitações qualitativas que se relacionam, por um lado, com a representação da complexidade semântica

dos textos, e por outro lado, com a interação do usuário com os sistemas de recuperação de informação.

Na maioria dos modelos apresentados neste capítulo transparece o seu caráter empírico, baseado muitas vezes em suposições e levando a um aumento progressivo da complexidade, sem refletir em avanços significativos dos resultados.

Apesar de seu aparente esgotamento, os modelos "quantitativos" ainda estão presentes na maioria dos sistemas de recuperação de informação e ganharam força com os mecanismos de busca da Web, que introduziram características específicas para tratar a quantidade de informação disponível na Internet (Capítulo 6). Além disso, os modelos quantitativos ainda fornecem seu considerável arsenal teórico para outras disciplinas, servindo de instrumento básico para o desenvolvimento de técnicas de representação do conhecimento ligadas à Inteligência Artificial.

5
Modelos Dinâmicos

No processo de recuperação de informação, os modelos quantitativos impõem uma determinada representação dos documentos. Essa representação é feita geralmente através da associação de termos de indexação e respectivos pesos aos documentos do *corpus*. Além de impositivos e unilaterais, os modelos quantitativos não prevêem qualquer tipo de intervenção do usuário na representação dos documentos.

Os modelos de recuperação de informação apresentados neste capítulo têm como principal característica o reconhecimento da importância do usuário na definição das representações dos documentos. Nesta ótica, os usuários interagem e interferem diretamente na representação dos documentos do *corpus*, permitindo uma evolução ou uma adaptação dos documentos aos interesses dos usuários do sistema, percebidos através de suas buscas e da atribuição de relevância (e não relevância) aos documentos recuperados (*relevance feedback*).

5.1 Sistemas Especialistas

Um sistema especialista é um sistema computacional que procura representar o conhecimento de um especialista humano em um domínio particular, de maneira a auxiliar na tomada de decisões e na resolução de problemas relacionados a esse domínio. A idéia subjacente à construção dos sistemas especialistas é que a inteligência não é apenas raciocínio, mas também memória. É comum considerarmos inteligente uma pessoa que possui

grande quantidade de informação sobre um determinado assunto. Assim, os sistemas especialistas obedecem ao princípio de que memória é condição necessária para a inteligência.

Os sistemas especialistas fazem parte de uma classe de sistemas ditos "baseados em conhecimento", desenvolvidos para servirem como consultores na tomada de decisões em áreas restritas. Estes sistemas são adequados para a solução de problemas de natureza simbólica, que envolvem incertezas resolvíveis somente com regras de "bom senso" e com raciocínio similar ao humano. Permitem representar o conhecimento heurístico na forma de regras obtidas através da experiência e intuição de especialistas de uma área específica.

A construção de sistemas especialistas obedece ao princípio de que a simulação da inteligência pode ser feita a partir do desenvolvimento de ferramentas computacionais para fins específicos. Um sistema especialista é um programa de computador associado a um "banco de memória" que contém conhecimentos sobre uma determinada especialidade (Teixeira, 1998; cap. II).

Um sistema especialista é composto de: uma *base de conhecimento* na qual está representado o conhecimento relevante sobre o problema, e um conjunto de métodos de manipulação desse conhecimento: o *motor de inferência* (Figura 16)

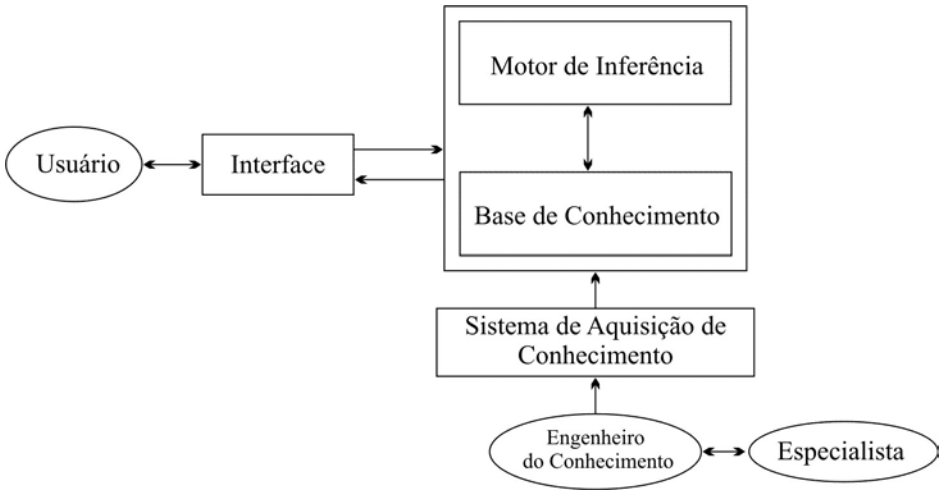


Figura 16 Estrutura de um sistema especialista

Pelo fato de a base de conhecimento estar separada do motor de inferência a modificação da base é facilitada. Assim, uma mudança na base de conhecimento é feita simplesmente através da adição de novas regras ou pela exclusão ou alteração de regras antigas.

A aquisição e a representação do conhecimento é o processo de maior importância na construção de um sistema especialista e levou ao surgimento de uma nova área na Ciência da Computação: a Engenharia do Conhecimento. A tarefa do engenheiro do conhecimento é "extrair" dos especialistas humanos os seus procedimentos, estratégias, raciocínios e codifica-los de forma adequada a fim de gerar a base de conhecimento.

O *sistema de aquisição de conhecimento* é um conjunto de ferramentas que facilita o trabalho do especialista e do engenheiro do conhecimento. Pode ser constituído simplesmente por um editor de texto com verificador da sintaxe exigida na base de conhecimento ou mecanismos de visualização gráfica da estrutura da informação e instrumentos de teste e validação semântica do conhecimento contido na base.

A *base de conhecimento* é o elemento central de um sistema especialista. É o local onde o conhecimento especializado humano está representado e armazenado. Geralmente, o conhecimento armazenado na base de conhecimento é representado por regras do tipo *condição-ação*, na forma SE-ENTÃO.

SE ⟨condição⟩ ENTÃO ⟨ação⟩

Sistemas que utilizam este esquema são chamados de "sistemas baseados em regras". Existem outras formas de representação de conhecimento tais como redes semânticas e *frames*.

O *motor de inferência* é composto por instrumentos para selecionar e aplicar o conhecimento armazenado na base na resolução do problema em questão. Estes instrumentos estão relacionados com a forma de inferência sobre os dados do sistema, com a forma como as regras da base de conhecimento serão testadas e com os métodos de tratamento de incerteza. A máquina de inferência busca as regras necessárias a serem avaliadas, ordena-as de maneira lógica e direciona o processo de inferência baseado nos dados simbólicos contidos na base de conhecimento.

A *interface* é utilizada para estabelecer a comunicação entre o usuário e o sistema, podendo ter a forma de menus, perguntas e representações gráficas. Durante o processamento de um sistema especialista, o usuário poderá ser requisitado pelo sistema a prestar informações adicionais na solução de um determinado problema. A cada pergunta respondida pelo usuário reduz-se a distância entre o problema e sua solução, podendo se desencadear um processo de aprendizagem automática que altere a configuração atual da base de

conhecimento e amplie a capacidade de sistema resolver futuros problemas. Assim, a base de conhecimento pode ser inicialmente constituída de poucas regras, podendo crescer conforme o sistema for sendo utilizado. Esse crescimento é possível graças à estrutura modular da base de conhecimento que permite a inclusão e exclusão de novos elementos.

Um exemplo clássico de sistema especialista é o MYCIN, desenvolvido durante a década de 70 com a finalidade de prescrever medicação para pacientes com infecções bacterianas. A partir de um conjunto de sintomas, ele identifica a moléstia e prescreve um medicamento apropriado utilizando uma base de dados contendo os sintomas e um sistema de raciocínio do tipo SE...ENTÃO. Por exemplo:

SE
o paciente apresenta **febre**,
o paciente apresenta vômitos e **diarréia**,
o paciente está **desidratado**
ENTÃO o paciente sofre de **infecção intestinal**

SE
o paciente sofre de **infecção intestinal**
ENTÃO o paciente deve tomar **ampicilina**

Este seria o caminho preliminar para construir um "diagnosticador" de infecções bacterianas. No entanto, o diagnóstico médico envolve uma grande margem de imprecisão, ou seja, existe um componente probabilístico no acerto de diagnósticos médicos, na medida em que, por exemplo, nem todos os sintomas ocorrem num paciente com determinada doença. Uma maneira de contornar esta dificuldade é através da atribuição de pesos diferentes a sintomas mais relevantes na caracterização de uma doença e, através destes pesos, estipular a probabilidade do paciente estar sofrendo de uma determinada moléstia. É aproximadamente desta maneira que o MYCIN opera: com uma margem de probabilidade que não fica muito distante da exibida pelos especialistas humanos.

Uma outra forma de representar o conhecimento em um sistema especialista é através de uma **rede semântica**. Uma rede semântica é composta por um conjunto de nós conectados por um conjunto de arcos. Os nós em geral representam objetos e os arcos representam as relações existentes entre eles. Dependendo do sistema, os nós podem ser utilizados para representar predicados, classes, palavras de uma linguagem, etc. A representação do conhecimento através de redes semânticas foi proposta por M.R. Quillian. Em artigo intitulado "*Semantic Memory*" Quillian (1968) propõe um modelo computacional da memória humana. Nesse modelo os conceitos são representados por nós, e as relações entre os

conceitos são representadas através dos arcos. Esse modelo tentava explicar diversos resultados experimentais sobre o comportamento da memória humana, como, por exemplo, o fato de que o reconhecimento de objetos que pertencem a classes mais numerosas toma mais tempo do que o reconhecimento dos objetos pertencentes a classes menores.

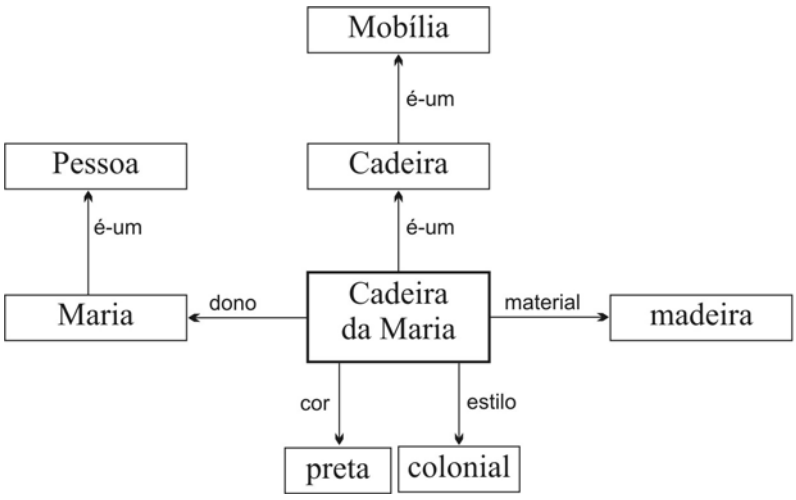


Figura 17 Exemplo de rede semântica na representação do conhecimento

A Figura 17 mostra um exemplo simples da utilização de redes semânticas. Ela representa conceitos sobre "mobília". As relações *é-um* são bastante comuns em sistemas de redes semânticas e determinam uma herança de propriedades. As demais relações (*dono, cor, estilo e material*) são específicas do domínio e representam propriedades dos conceitos (Rich, 1988, p. 253):

Dois artigos publicados em 1975 tiveram grande influência na pesquisa relacionada às redes semânticas: o artigo de Woods (1975), que faz uma análise do significado dos arcos nas redes semânticas e o artigo de Minsky (1975), que apresenta o conceito de *frames*. Em seu artigo, Woods chama a atenção para a necessidade de uma semântica formal que fundamente os sistemas baseados em redes semânticas. Este artigo foi seguido de uma série de outros que descreviam a utilização das redes semânticas apenas como uma notação sintática alternativa para fórmulas lógicas; outros apresentavam as redes semânticas como um método independente de representação de conhecimento, utilizando o formalismo lógico apenas como ferramenta para a definição de uma semântica para os nós e os arcos. O artigo de Minsky introduziu a noção de nós com uma estrutura interna, os ***frames***, criando uma nova forma de representação de conhecimento.

A Figura 18 apresenta uma adaptação da Figura 17 utilizando o conceito de *frames*.

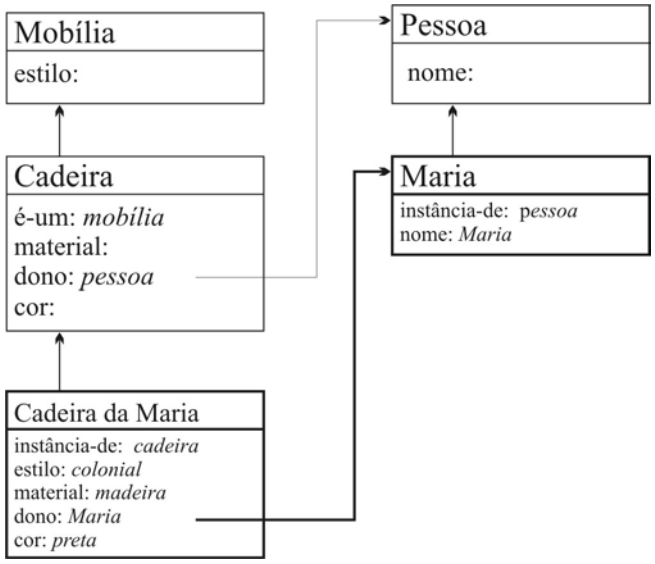


Figura 18 Exemplo da utilização de *frames* na representação do conhecimento

Basicamente um *frame* é uma coleção de atributos ("*slots*"), e valores a eles associados. Cada *frame* representa uma classe ou uma instância (elemento de uma classe). A criação de um sistema de *frames* é possível graças ao fato que o valor de um atributo de um *frame* pode ser um outro *frame*. Um sistema de *frames* pode assim definir uma hierarquia de classes, como na Figura 18. A relação *é-um* define uma relação transitiva de *subclasse*. A relação *instância-de* define a classe a qual um determinado elemento pertence Os *frames* "Mobília", "Pessoa" são exemplos de classes. O *frame* "Cadeira" é também uma classe, mas é ainda uma subclasse de "Mobília", e herda desta a propriedade *estilo*. "Maria" é um elemento (ou instância) da classe "Pessoa". O *frame* "Cadeira da Maria" é uma instância da classe "Cadeira".

John F. Sowa (2000) apresenta um estudo completo e detalhado sobre as diversas formas de representação do conhecimento, e é uma referência obrigatória para quem deseja aprofundar o assunto.

5.1.1 Sistemas Especialistas na recuperação de informação

A recuperação de informação é um processo cuja eficiência depende em grande parte do conhecimento sobre o assunto que se deseja pesquisar e sobre a estrutura de representação dos documentos do *corpus*. Parece então plausível pensar que algum conhecimento necessário ao processo de recuperação de informação poderia ser incorporado a um sistema para que este seja capaz de auxiliar no processo.

Um exemplo da utilização de procedimentos típicos dos sistemas especialistas na recuperação de informação é o sistema IOTA (Chiaramella et al, 1986). O sistema IOTA, desenvolvido no *Laboratoire Génie Informatique de Grenoble*, tem como uma de suas características a sua habilidade de construir automaticamente uma base de conhecimento a partir dos documentos do *corpus* (Chiamarella e Defude, 1987; Bruandet, 1987).

No sistema IOTA o processo de construção automática da base de conhecimento é realizado através da identificação dos principais conceitos contidos nos textos dos documentos do *corpus*. Esses conceitos são identificados utilizando-se cálculos estatísticos de co-ocorrência de pares de palavras. A hipótese que está por trás dessa estratégia é que se duas palavras aparecem próximas em vários documentos do *corpus* então elas possuem um certo relacionamento. O resultado desse processo é um conjunto de conceitos representados por grupos de palavras que caracterizam uma idéia contida nos documentos do *corpus*. Esses conceitos são integrados à rede semântica que compõe a base de conhecimento. Essa rede semântica é utilizada para melhorar a eficiência do sistema e auxiliar o usuário na formulação de suas buscas. Para cada novo documento inserido no *corpus* altera-se a configuração da rede semântica. Ferneda (1997) apresenta detalhadamente as técnicas utilizadas para a construção automática de uma rede semântica a partir de um conjunto de documentos.

Outro sistema que utiliza alguns conceitos dos sistemas especialistas é o sistema RUBRIC (Tong et al, 1985; 1987). O sistema RUBRIC (***Ru**le-**B**ased **R**etrieval of **I**nformation by **C**omputer*) utiliza *frames* e regras para representar conceitos relacionados com a informação que o usuário espera recuperar. No sistema RUBRIC o usuário é capaz de construir sua própria base de conhecimento sobre um determinado assunto através da especificação e organização de conceitos na forma de uma rede de *frames*. Para cada conceito (*frame*) o usuário define um conjunto de regras do tipo **se...então** que caracteriza o conceito. Por exemplo, supondo que o usuário criou o conceito "recuperação de informação" e definiu o seguinte conjunto de regras:

se "recuperação" **e** "informação" **então** "recuperação de informação" (0.5)

se sentence "recuperação" **e "**informação" **então** "recuperação de informação" (0.7)

Se um documento contém ambas as palavras "recuperação" e "informação", então existe 50% de possibilidade (probabilidade) de que o assunto tratado por este documento esteja relacionado à "recuperação de informação". Se as palavras "recuperação" e

"informação" estiverem em uma mesma sentença ("*sentence*"), essa probabilidade aumenta para 70%.

É importante não superestimar o potencial das técnicas de recuperação de informação baseados em conhecimento. Apesar de atualmente as pesquisas em representação do conhecimento apresentarem grandes avanços, dificilmente uma máquina poderá substituir completamente a habilidade humana, mesmo em operações que não envolvam conhecimentos ou habilidades complexas. No entanto, as idéias relacionadas aos sistemas especialistas podem contribuir para a implementação de sistemas que abranjam áreas do conhecimento bastante específicas e em situações nas quais os usuários e os sistemas possam se complementar.

5.2 Redes neurais

Sabe-se que o cérebro é composto de bilhões de *neurônios*. Um neurônio é uma célula formada por três seções com funções específicas e complementares: *corpo*, *dendritos* e *axônio*. Os dendritos recebem informações na forma de impulsos nervosos provenientes de outras células e os conduzem até o corpo celular (*soma*), onde a informação é processada e novos impulsos são eventualmente transmitidos a outras células. A conexão entre o axônio de um neurônio e uma célula vizinha é chamada sinapse. Através das sinapses os neurônios se unem formando as redes neurais. Cada neurônio pode ter entre mil e dez mil sinapses, o que possibilita a formação de redes bastante complexas. As sinapses funcionam também como "válvulas" que controlam a transmissão de impulsos entre os neurônios da rede. A Figura 19 ilustra de forma simplificada as partes de um neurônio.

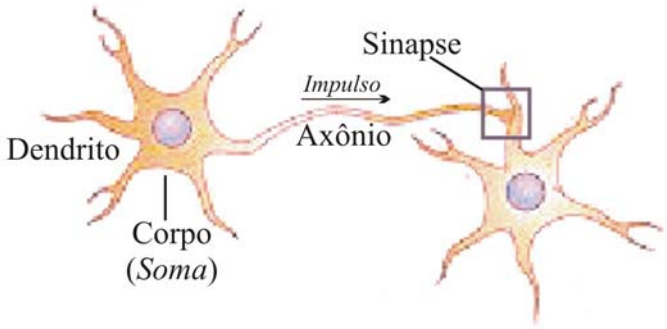


Figura 19 Representação simplificada de um neurônio

Os dendritos captam os estímulos recebidos em um determinado período de tempo e os transmitem ao corpo do neurônio onde são processados. Quando tais estímulos atingirem um determinado limite, o corpo da célula envia um novo impulso que se propaga pelo axônio até as sinapses e daí para as células vizinhas. Este processo pode se repetir através de várias camadas de neurônios. Como resultado, a informação de entrada é processada podendo levar o cérebro a comandar reações físicas.

A habilidade de um ser humano em realizar funções complexas e principalmente a capacidade de aprender advém do processamento paralelo e distribuído da rede de neurônios do cérebro. Os neurônios do córtex, a camada externa do cérebro, são responsáveis pelo processamento cognitivo. Um novo conhecimento ou uma nova experiência pode levar a alterações estruturais no cérebro. Tais alterações são efetivadas através de um rearranjo das redes de neurônios e reforçando ou inibindo algumas sinapses (Haykin, 2001, p. 32-36).

5.2.1 Redes neurais artificiais

A busca por um modelo computacional que simule o funcionamento das células do cérebro data dos anos 40, com o trabalho de McCulloch e Pitts (1943). O entusiasmo pela pesquisa neste campo cresceu durante os anos 50 e 60. Nesse período, Rosenblatt (1958) propôs um método inovador de aprendizagem supervisionada: o *percepton*. Até 1969, muitos trabalhos foram realizados utilizando o *percepton* como modelo. No final dos anos 60, Minsky e Pappert (1969) publicam um livro no qual apresentam importantes limitações do *perceptron*. As dificuldades metodológicas e tecnológicas, juntamente com os ataques extremamente pessimistas de Papert e Minsky, fizeram com que as pesquisas arrefecessem nos anos seguintes. Durante os anos 70 a pesquisa contava apenas com um número ínfimo de cientistas. Mas nos anos 80 o entusiasmo ressurge devido a avanços metodológicos importantes e também graças aos avanços da ciência da computação.

O modelo de neurônio artificial da Figura 20 é uma simplificação do modelo apresentado por Haykin (2001, p. 36):

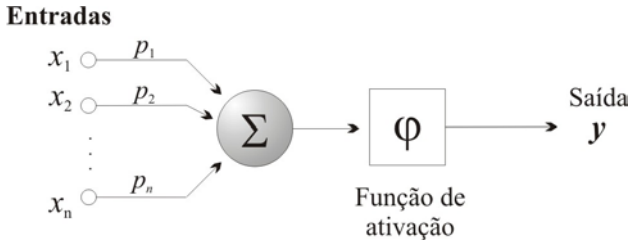


Figura 20 Modelo matemático de um neurônio

Este modelo é composto por três elementos básicos:

- Um conjunto de n conexões de entrada (x1, x2, ..., xn), caracterizadas por pesos (p1, p2, ..., pn);
- Um somador (Σ) para acumular os sinais de entrada;
- Uma função de ativação (φ) que limita o intervalo permissível de amplitude do sinal de saída (y) a um valor fixo.

O comportamento das conexões entre os neurônios é simulado através de seus pesos. Os valores de tais pesos podem ser negativos ou positivos, dependendo das conexões serem inibitórias ou excitatórias. O efeito de um sinal proveniente de um outro neurônio é determinado pela multiplicação do valor (intensidade) do sinal recebido pelo peso da conexão correspondente ($\boldsymbol{x_i} \times \boldsymbol{p_i}$). O *somador* efetua o somatório dos valores $\boldsymbol{x_i} \times \boldsymbol{p_i}$ de todas as conexões e o valor resultante é enviado para a *função de ativação*, que define a saída ($\boldsymbol{y}$) do neurônio.

Combinando diversos neurônios forma-se uma rede neural. As redes neurais artificiais são modelos que buscam simular o processamento de informação do cérebro humano. São compostas por unidades de processamentos simples, os neurônios, que se unem através de conexões.

De uma forma simplificada, uma rede neural artificial pode ser vista como um grafo onde os nós são os neurônios e as ligações fazem a função das sinapses, como exemplificado na Figura 21:

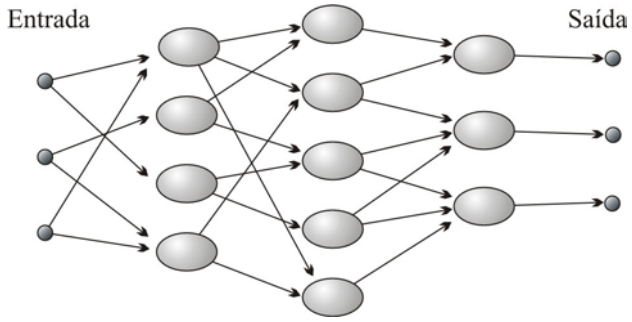


Figura 21 Representação de uma rede neural artificial

As redes neurais se diferenciam pela sua arquitetura e pela forma como os pesos associados às conexões são ajustados durante o processo de aprendizado. A arquitetura de uma rede neural restringe o tipo de problema no qual a rede poderá ser utilizada, e é definida pelo número de camadas (*camada única* ou *múltiplas camadas*); pelo número de nós em cada camada, pelo tipo de conexão entre os nós (*feedforward* ou *feedback*) e por sua topologia (Haykin, 2001, p. 46-49).

5.2.2 Aprendizagem

Uma das propriedades mais importantes de uma rede neural é a capacidade de aprender através de exemplos e fazer inferências sobre o que aprenderam, melhorando gradativamente o seu desempenho. As redes neurais utilizam um *algoritmo de aprendizagem*, cuja tarefa é ajustar os pesos das conexões (Braga, Carvalho e Ludemir, 2000, capítulo 2).

Existem duas formas básicas de aprendizado de redes neurais: *aprendizado supervisionado* e *aprendizado não supervisionado*. Para cada uma dessas formas existem algumas variantes.

No aprendizado supervisionado um agente externo (professor) apresenta à rede neural alguns conjuntos de padrões de entrada e seus correspondentes padrões de saída. Portanto, é necessário ter um conhecimento prévio do comportamento que se deseja ou se espera da rede. Para cada entrada o professor indica explicitamente se a resposta calculada é boa ou ruim. A resposta fornecida pela rede neural é comparada à resposta esperada. O erro verificado é informado à rede para que sejam feitos ajustes a fim de melhorar suas futuras respostas.

Na aprendizagem não supervisionada, ou aprendizado auto-supervisionado, não existe um agente externo para acompanhar o processo de aprendizado. Neste tipo de aprendizagem somente os padrões de entrada estão disponíveis para a rede neural. A rede processa as

entradas e, detectando suas regularidades, tenta progressivamente estabelecer representações internas para codificar características e classificá-las automaticamente. Este tipo de aprendizado só é possível quando existe redundância nos dados de entrada, para que se consiga encontrar padrões em tais dados.

5.2.3 Redes Neurais na recuperação de informação

De uma forma simplificada, a recuperação de informação lida com documentos, termos de indexação e buscas. Uma tarefa comum para um sistema de recuperação de informação é pesquisar documentos relevantes que satisfazem uma determinada expressão de busca através dos termos de indexação. Pode-se dizer que em um sistema de recuperação de informação de um lado estão as expressões de busca, do outro lado estão os documentos e no meio ficam os termos de indexação. Essa estrutura pode ser vista como uma rede neural de três camadas: a camada de busca seria a camada de entrada da rede neural, a camada de documentos seria a saída e a camada de termos de indexação seria uma camada central. A Figura 22 mostra um exemplo genérico da aplicação das redes neurais na recuperação de informação.

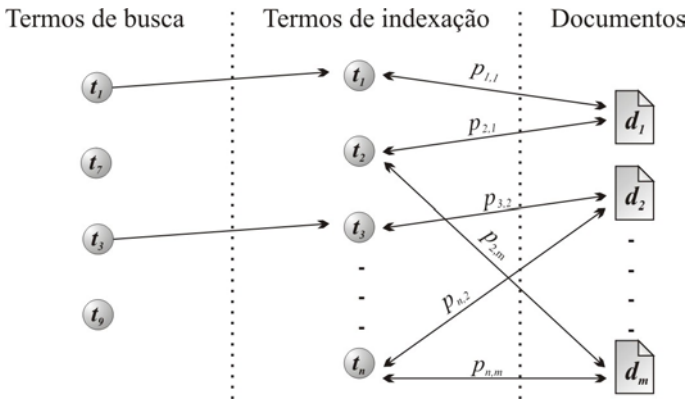


Figura 22 Representação de rede neural aplicada à recuperação de informação

Os termos de busca (t_1, t_7, t_3, t_9) iniciam o processo de inferência através da ativação dos respectivos termos de indexação. Alguns termos da expressão de busca podem não fazer parte do conjunto de termos de indexação, como é o caso do termo t_7 e t_9. Nesse caso, esses termos não ativarão nenhum termo de indexação e, portanto, não serão considerados. Os termos de indexação ativados pelos termos da busca enviam sinais para os documentos que serão multiplicados pelos pesos de cada ligação ($p_{1,1}$, $p_{1,2}$, ..., $p_{n,m}$). Os documentos ativados

enviam sinais que são conduzidos de volta aos termos de indexação. Ao receberem estes estímulos, os termos de indexação enviam novos sinais aos documentos, repetindo o processo. Os sinais tornam-se mais fracos a cada iteração e o processo de propagação eventualmente pára. O resultado final de uma busca será o conjunto dos documentos que foram ativados, cada qual com um nível ativação, que pode ser interpretado como o grau de relevância do documento em relação à busca. Entre os documentos resultantes podem aparecer documentos que não estão diretamente relacionados aos termos utilizados na expressão de busca, mas que foram inferidos durante a pesquisa e possuem um certo grau de relacionamento com a necessidade de informação do usuário. A ativação do termo de indexação t_1, por exemplo, ativou a conexão com o documento d_2. O documento d_2 por sua vez também ativou o termo t_2, que não fazia parte do conjunto de termos de busca. O termo t_2 poderá ativar o documento d_n que, dependendo do seu grau de ativação, pode vir a fazer parte do conjunto de documentos recuperados.

Mozer (1984) foi o pioneiro na utilização de técnicas de redes neurais na recuperação de informação. Ele utilizou uma arquitetura bastante simples que não empregava uma das principais características das redes neurais que é a capacidade de aprender. A Figura 23 mostra um exemplo apresentado por Ford (1991, p. 108), que utiliza a arquitetura de rede neural idealizada por Mozer:

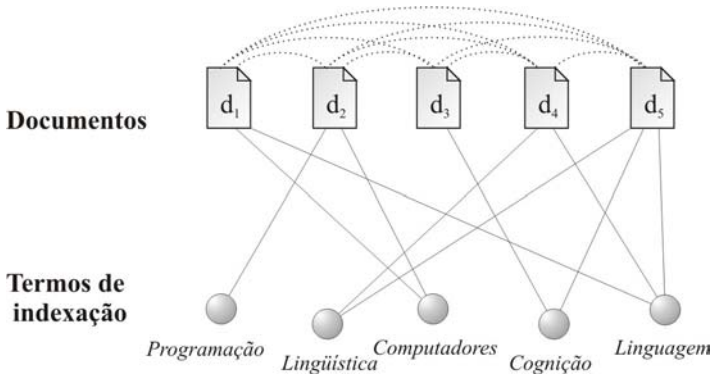
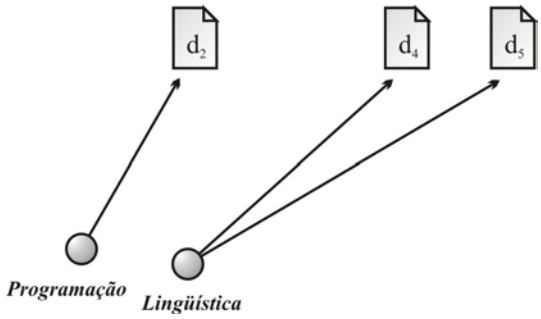


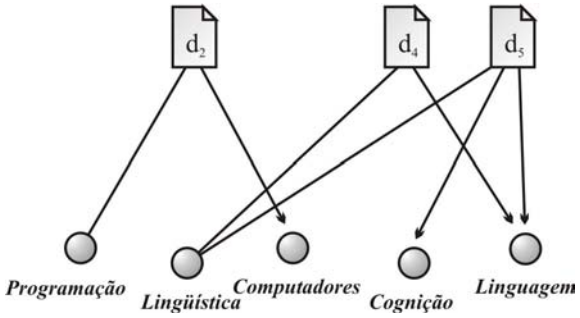
Figura 23 Exemplo de uma rede neural

A linhas contínuas representam ligações *excitatórias* entre os termos de indexação e os documentos. As linhas pontilhadas, que ligam pares de documentos, representam ligações *inibitórias*, isto é, ligações que reduzem a força de associação entre os nós. Os termos de indexação ativam os documentos que são indexados por eles e vice-versa. Um documento, ao ser ativado, reduz o nível de ativação dos demais documentos.

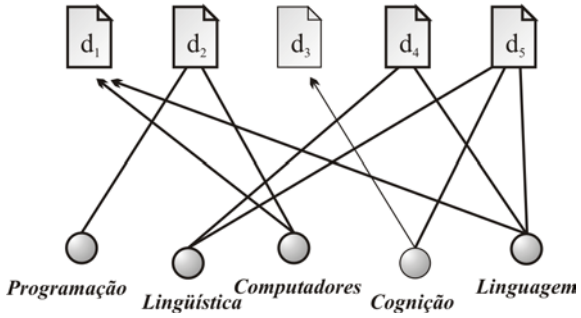
Utilizando uma expressão de busca que contém os termos "programação" e "lingüística", por exemplo, a rede neural da Figura 23 apresentará a seguinte seqüência de ativação:



1. Inicialmente serão ativados os nós correspondentes aos termos de busca ("*programação*" e "*lingüística*"). O termo "*programação*" irá ativar o documento **d_2**. O termo "*lingüística*" ativará os documentos **d_4** e **d_5**:



2. O documento **d_2** ativará todos os termos de indexação usados para indexá-lo: "*programação*" e "*computadores*". Assim, o termo "*programação*" é reforçado e o termo "*computadores*" é ativado pela primeira vez. Os documentos **d_4** e **d_5** ativarão o termo "*linguagem*" e reforçar a ativação do termo "*lingüística*". O documento **d_5** ainda ativará também o termo "*cognição*":



3. O termo "*computadores*" ativará os documentos indexados por ele. Assim o documento **d_2** é reforçado, e o documento **d_1** é ativado. O termo "*linguagem*" reforçará a ativação dos documentos **d_4** e **d_5** e ativará também o documento **d_1**. O termo "*cognição*" ativará o documento **d_3**.

Este processo se propaga até uma estabilização da rede neural, quando cessam as ativações entre seus nós.

O nível de ativação de cada documento representa a sua relevância em relação à busca. Os documentos **d_2**, **d_4** e **d_5**, que foram ativados diretamente pelos termos de busca, terão um nível de ativação maior do que o documento d_3, que é indexado por um termo que foi indiretamente ativado durante a busca ("*cognição*").

Para que sejam apresentados resultados satisfatórios, os parâmetros da rede neural (pesos das conexões, funções de ativação, etc.) devem ser configurados de forma precisa. Porém, o sistema pode compensar algumas inconsistências na indexação e até possíveis imprecisões nas expressões de busca dos usuários. Mozer enfatiza que a grande vantagem deste modelo é a habilidade em produzir resultados não esperados, recuperando documentos que não possuem nenhum termo em comum com a expressão de busca, mas mesmo assim são relevantes para o usuário. No exemplo apresentado, em resposta à expressão de busca contendo os termos "*programação*" e "*lingüística*", o documento d_1, que é indexado pelos termos "*computadores*" e "*linguagem*", obteve um certo nível de ativação (Ford, 1991, p. 109).

As ligações entre os documentos são inibitórias, isto é, um documento, quando ativado, reduz o nível de ativação dos demais. Isso causa uma competição entre os documentos, fazendo com que apenas os documentos mais ativados durante o processo de busca sejam efetivamente recuperados, reduzindo assim o número de documentos resultantes.

Ao final do processo de pesquisa, o grau de ativação de cada documento pode ser utilizado como critério de ordenamento dos itens resultantes. Os documentos com maior nível de ativação são geralmente aqueles que possuem todos os termos utilizados na expressão de busca, seguidos dos documentos que possuem somente alguns dos termos de busca e dos que foram apenas inferidos durante o processo de pesquisa.

Bein e Smolensky (1988) implementaram e testaram esse modelo de rede neural proposta por Mozer, utilizando 12.990 documentos e 6.832 termos de indexação. Eles avaliaram os resultados apresentados como satisfatórios, e sugerem novos testes utilizando bases de dados maiores e com características diversas. Eles ressaltam também a necessidade de um melhor conhecimento do funcionamento interno da rede neural para que seja possível identificar os parâmetros que afetam o seu desempenho.

Como foi observado anteriormente, Mozer não utilizou uma das características mais fortes das redes neurais, que é a habilidade de aprender através da alteração dos pesos associados às ligações entre os nós. Um sistema mais recente, que explora tal habilidade das redes neurais, é o sistema AIR.

Desenvolvido por Belew (1989), o sistema AIR (*Adaptative Information Retrieval*) utiliza uma arquitetura de rede neural composta de três camadas que representam os termos de indexação, os documentos e os seus autores. As ligações são feitas entre os documentos e seus autores e entre documentos e seus termos de indexação, como apresentado na Figura 24.

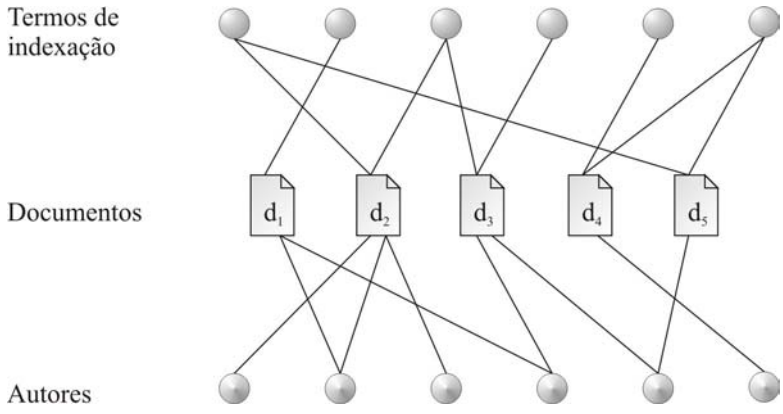


Figura 24 Arquitetura de rede neural do sistema AIR

Uma busca pode ser feita não apenas através da ativação dos termos de indexação, mas por qualquer tipo de nó (autor, documento ou termo de indexação), ou por alguma combinação deles. Durante a pesquisa é feita a ativação dos nós da rede e, quando o sistema

se estabiliza, os nós e as ligações que foram inferidos são apresentados ao usuário. Em uma interface apropriada o usuário poderá atribuir um grau de relevância para cada um dos itens recuperados utilizando uma escala fixa com quatro níveis, variando do "*muito relevante*" ao "*totalmente irrelevante*". Este *feedback* é utilizado na aprendizagem da rede neural, que modifica os pesos associados às conexões entre seus nós.

Através da aprendizagem, o sistema busca gradualmente adequar os pesos das conexões, a fim de melhor representar a relevância percebida através da interação do usuário. Segundo Ford (1991, p. 161-172), o sistema AIR implementa a noção de "relevância consensual", que pode ser útil para usuários não familiarizados com o domínio do *corpus*.

Não existem evidências conclusivas da superioridade das redes neurais em relação aos modelos tradicionais de recuperação de informação. Porém, as redes neurais oferecem muitas características atrativas no processo de recuperação de informação, principalmente a habilidade inata de se adaptarem às modificações nas condições do "ambiente", representado pelas buscas dos usuários (Doszkocs, Reggia e Lin, 1990).

5.3 Algoritmos genéticos

Em 27 de dezembro de 1831, Charles Darwin zarpou a bordo do HMS Beagle para uma viagem de pesquisa cujo roteiro incluía o litoral da América do Sul, várias ilhas do Pacífico, a Austrália e uma circunavegação no globo. Durante a viagem, Darwin observou que à medida que passava de uma região para outra, uma mesma espécie animal apresentava características diferentes. Notou ainda que entre as espécies extintas e as atuais existiam traços comuns, embora bastante diferenciados. Tais fatos levaram-no a supor que os seres vivos não eram imutáveis como se pensava, mas que se transformam. Com base nestas observações, Darwin começou a esboçar a teoria da evolução das espécies.

Na base da teoria evolucionista proposta por Darwin está a luta pela vida, segundo a qual em cada espécie animal existe uma permanente concorrência entre os indivíduos. Os mais adaptados ao ambiente terão maior probabilidade de sobreviver e procriar, e a própria natureza se incumbe de proceder a esta seleção (Strathern, 2001).

As idéias gerais da teoria da evolução das espécies sofreram, aos poucos, alterações e aperfeiçoamentos, mas as bases do evolucionismo subsistem até hoje e estão ligadas ao nome

de Darwin. No entanto, a teoria de Darwin não explicava como era feita a transmissão das características dos pais para os filhos, a hereditariedade.

No ano de 1900 Hugo Vries deparou-se com alguns artigos publicados pelo monge austríaco Gregor Mendel. Embora seu trabalho tivesse sido ignorado durante sua vida, Mendel, trabalhando com ervilhas, descobrira as leis da hereditariedade que revolucionaram a biologia e traçariam as bases da genética.

Sabe-se hoje que todos os organismos vivos são constituídos de células que possuem o mesmo conjunto de cromossomos. Os cromossomos são cadeias de DNA (ácido desoxirribonucléico) que servem como "molde" para "fabricar" seres vivos. Um cromossomo é formado por genes, blocos de DNA, que ditam os aspectos da hereditariedade dos indivíduos. Pode-se dizer que cada gene é responsável por uma característica do ser vivo, como a cor dos olhos, a cor dos cabelos, etc. Durante a reprodução, cada um dos pais passa metade de seus cromossomos aos filhos, em um processo denominado *crossover*.

O material genético pode sofrer mutações decorrentes de operações de *crossover* imperfeitas ou de estímulos externos. Embora a ocorrência de mutações seja rara, ela tem como conseqüência uma grande diversificação nas características de um indivíduo ou até de uma população.

Sobre a casualidade da mutação age a seleção natural que seleciona características que melhoram a adaptação dos organismos ao seu meio ambiente. Os indivíduos mais adaptados ao ambiente possuem mais chances de sobreviverem e se reproduzirem, transmitindo seu material genético para gerações futuras.

5.3.1 Evolução computacional

Como se pode supor, os algoritmos genéticos foram criados tendo como referência a teoria de Darwin sobre a evolução dos seres vivos. Dessa forma, pode-se dizer que soluções obtidas através de algoritmos genéticos são ditas evolutivas.

Um algoritmo genético é um processo repetitivo que mantém uma população de "indivíduos", que representam as possíveis soluções para um determinado problema. A cada "geração" os indivíduos da população passam por uma avaliação de sua capacidade em oferecer uma solução satisfatória para o problema. Essa avaliação é feita por uma função de adaptação ou função de *fitness*. De acordo com esta avaliação alguns indivíduos, selecionados

de acordo com uma regra probabilística, passam por um processo de reprodução, gerando uma nova população de possíveis soluções. Pressupõe-se que a população vá gradativamente ficando mais apta para solucionar o problema. A estrutura funcional de um algoritmo genético está representada na Figura 25.

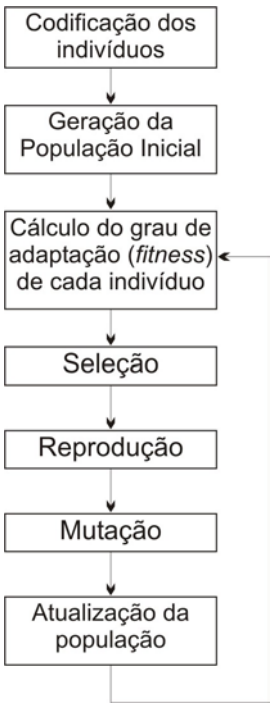


Figura 25 Seqüência de execução de um algoritmo genético

Embora um algoritmo genético nem sempre possa encontrar uma solução ótima para um determinado problema, na maioria das vezes é capaz de encontrar soluções aceitáveis para problemas relativamente complexos.

A partir dos anos 80 os algoritmos genéticos receberam um grande impulso em diversas áreas científicas devido principalmente à versatilidade e aos excelentes resultados apresentados. A popularização dos computadores e o aparecimento de sistemas cada vez mais rápidos e potentes também ajudaram muito o seu desenvolvimento.

O ponto de partida para a utilização de um algoritmo genético na solução de um problema consiste em definir uma representação adequada dos indivíduos (soluções) envolvidos no problema de maneira que o algoritmo possa operá-los. No algoritmo proposto por Holland (1998), cada cromossomo é representado por uma cadeia binária de tamanho fixo, onde cada gene pode assumir o valor um ou o valor zero. Por exemplo:

Cromossomo 1	01001
Cromossomo 2	01110
Cromossomo 3	10000
Cromossomo 4	10110

Apesar da representação binária ser a mais utilizada, dependendo do tipo de aplicação podem existir formas mais eficientes de representar os cromossomos, como a utilização de símbolos ou números reais (Mitchell, 2002, p.156-158).

Feita a escolha de como os indivíduos serão representados, o próximo passo é definir quantos e quais indivíduos farão parte da população inicial. A população inicial pode ser obtida através da geração aleatória de indivíduos, obedecendo a certas condições estabelecidas pelo usuário, ou cada indivíduo pode ser criado individualmente com objetivo de gerar uma população dentro de certo intervalo onde se acredita estar a resposta para o problema.

O tamanho da população (número de indivíduos) pode afetar o desempenho global e a eficiência dos algoritmos genéticos. Populações muito pequenas têm grandes chances de perder a diversidade necessária para convergir para uma boa solução do problema que se deseja resolver. Por outro lado, se a população tiver muitos indivíduos o algoritmo poderá perder grande parte de sua eficiência pela demora no cálculo da *função de adaptação* de todos os indivíduos a cada iteração.

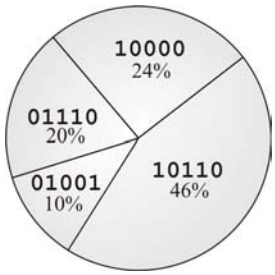
Para a população inicial e a cada nova geração será necessário calcular o grau de adaptação de cada indivíduo. Esse cálculo é feito através de uma função de adaptação que deve ser definida tendo em vista o tipo de problema a ser resolvido. A função de adaptação (também chamada de *função de fitness*) deve refletir a qualidade de cada indivíduo em solucionar o problema. Uma função de *fitness* bastante utilizada é o Coeficiente de Similaridade de Jaccard (van Rijsbergen, 1979). Esta função calcula o valor da similaridade entre duas seqüências binárias e é definida como o número de posições com valor 1 em ambas as seqüências, dividido pelo número de posições com valor 1 em pelo menos uma das seqüências.

$$\frac{\text{Quantidade de posições com 1 em ambas as seqüências}}{\text{Quantidade de posições com 1 em pelo menos uma das seqüências}}$$

De acordo com a teoria de Darwin, os indivíduos mais adaptados (com maior *fitness*) ao meio ambiente têm maior chance de se reproduzirem. Para simular a casualidade da seleção natural, um algoritmo genético pode utilizar alguns métodos para selecionar aleatoriamente os indivíduos que deverão se reproduzir. Um dos métodos mais utilizados é chamado de Roleta (*Roulette Wheel*).

No método da Roleta, para cada indivíduo da população é atribuída uma probabilidade de reprodução proporcional ao seu *fitness*. Assim, quanto maior o *fitness* de um indivíduo, maior a possibilidade dele se reproduzir. Por exemplo:

Nº	Cromossomo	*fitness*	percentual
1	01001	0.05	10%
2	01110	0.10	20%
3	10000	0.12	24%
4	10110	0.23	46%
	total	0.50	100%



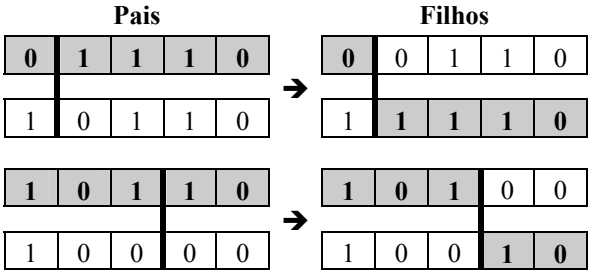
No exemplo acima, o cromossomo número 4 (`10110`) é o mais adaptado e sua chance de ser selecionado para reprodução é de 46%. O cromossomo `01001` é o menos adaptado e sua chance de ser selecionado é de apenas 10%.

O processo computacional da seleção assemelha-se a um sorteio feito através de uma roleta onde estão representados todos os indivíduos da população. O tamanho de cada "fatia" da roleta é proporcional ao grau de adaptação (*fitness*) de cada indivíduo.

A roleta é "girada" 4 vezes, sorteando quatro indivíduos que se reproduzirão. Supondo que os indivíduos selecionados foram: `01110` (2), `10110` (4), `10110` (4) e `10000` (3), observa-se que o cromossomo 4 foi selecionado duas vezes, o que é coerente já que o valor de seu *fitness* é bastante superior aos demais. O cromossomo 1 não foi selecionado pois possui baixo valor de *fitness*.

Com a utilização da roleta, existe a probabilidade de o indivíduo com o maior *fitness* não se reproduzir. Uma estratégia alternativa à roleta é simplesmente manter sempre o indivíduo com maior *fitness* da geração atual na geração seguinte, estratégia conhecida como seleção elitista. Outros métodos de seleção são apresentados por Mitchell (2002, p. 166-171).

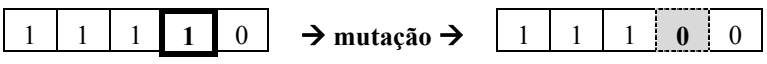
Definido o grupo de indivíduos reprodutores, o próximo passo é realizar a reprodução propriamente dita, ou seja, o *crossover*. Em termos biológicos, *crossover* significa efetuar uma recombinação do material genético dos "pais", gerando novos indivíduos "filhos". Do grupo de cromossomos reprodutores, dois pares são selecionados aleatoriamente. Para cada par é escolhido (também aleatoriamente) um ponto de divisão. Supondo-se que para o par de cromossomos `01110` e `10110` foi escolhido para realizar *crossover* a partir do ponto de divisão 1 e para o par `10110` e `10000` o ponto de *crossover* será o ponto 3.



Os cromossomos resultantes da reprodução do primeiro par são `00111` e `11110`. A reprodução do segundo par resultou nos cromossomos `10100` e `10010`.

Nesse exemplo foi utilizado o chamado *crossover* simples, com apenas um único ponto de divisão. Dependendo do tipo de problema é possível utilizar dois ou mais pontos de divisão.

Durante o processo de reprodução, há uma *probabilidade de mutação,* que determina a freqüência de ocorrência da mutação. Para cada gene dentro de um cromossomo é decidido se uma mutação deverá ou não ocorrer. Se a mutação for realizada, o valor do gene que está sendo verificado será alterado. Em cadeias binárias, um gene com valor 1 terá seu valor alterado para 0, um gene com valor 0 será alterado para 1. Por exemplo, o algoritmo decide alterar o valor do bit (gene) da posição 4 do cromossomo `11110`:



Após a mutação obtém-se um novo conjunto de indivíduos (cromossomos), uma nova população. O cálculo do grau de adaptação de cada indivíduo é calculado e o processo se repete.

Grande parte da capacidade dos algoritmos genéticos provém do fato de existir um conjunto de cromossomos muito diverso. As mutações ajudam a prevenir a estagnação das populações, ajudando a preservar esta diversidade através das gerações.

5.3.2 Algoritmos Genéticos na recuperação de informação

A aplicação dos algoritmos genéticos na recuperação de informação representa um novo modelo para todo o processo de recuperação. As representações dos documentos podem ser vistas como um tipo de "código genético". Nesse código genético um cromossomo é representado por um vetor binário onde cada elemento armazena o valor 0 ou o valor 1, correspondendo respectivamente à presença ou ausência de um determinado termo na representação do documento.

Gordon (1988) e Blair (1990) apresentam um modelo no qual cada documento é representado por um conjunto de cromossomos. Segundo Gordon, a inerente indeterminação da representação de um documento pode ser interpretada como um tipo de variabilidade genética que permite aos documentos se adaptarem aos diferentes tipos de "meio ambiente". Entenda-se por "meio ambiente" o conjunto das buscas realizadas pelos usuários. No código genético de um documento alguns cromossomos identificarão melhor a relevância do documento e outros descreverão melhor a sua não-relevância. Após execução da busca, o usuário seleciona os documentos que considera relevantes para sua necessidade de informação. Durante esse processo, conhecido como *relevance feedback*, para um documento considerado relevante as descrições que foram responsáveis pela sua recuperação recebem um crédito pelo seu sucesso e as descrições que não participaram de sua recuperação são rebaixadas. Para um documento recuperado que não foi considerado relevante, as descrições que foram responsáveis pela sua recuperação são rebaixadas e as demais descrições recebem um crédito.

A seguir será apresentado um exemplo do processo de recuperação de informação utilizando algoritmo genético. Os documentos do *corpus* serão representados por um conjunto de cromossomos, como utilizado por Gordon (1988). Porém serão feitas algumas simplificações no processo "evolutivo" para não sobrecarregar o exemplo com uma quantidade excessiva de detalhes.

Na Figura 26 é representado um *corpus* contendo seis documentos, sendo que cada documento é descrito de quatro diferentes maneiras através de quatro cromossomos compostos por cinco genes. Um gene representa a presença (1) ou a ausência (0) de um determinado termo de indexação (t_i) na descrição do documento:

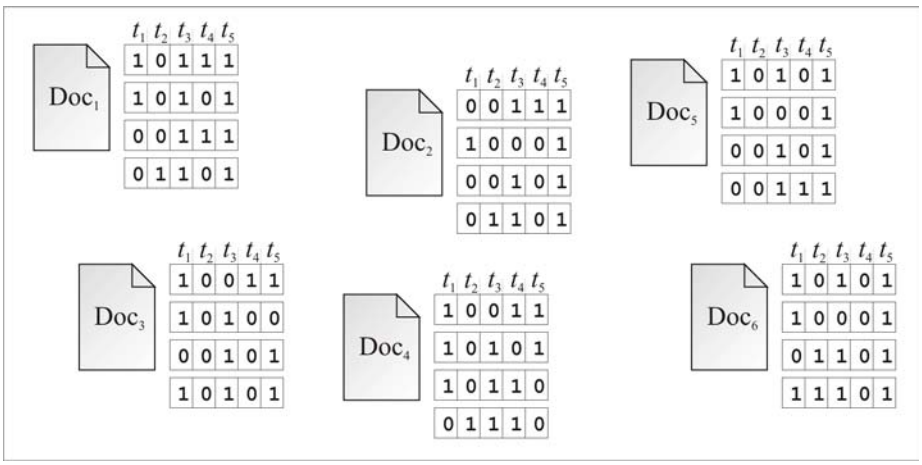


Figura 26 *Corpus* com documentos representados por quatro "cromossomos"

A cada busca do usuário será calculado o coeficiente de Jaccard para cada um dos cromossomos de cada um dos documentos. O grau de adaptação (*fitness*) de cada cromossomo é dado pela média dos coeficientes obtidos a cada busca. É calculado também o ***fitness* do documento** através da média do *fitness* de cada cromossomo.

Após uma busca expressa através de uma seqüência binária, por exemplo, `01010`, e supondo que o documento Doc_1 tenha sido considerado relevante pelo usuário, este documento apresentará os seguintes valores:

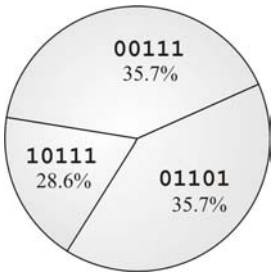
expressão de busca: `01010`

							fitness
Doc_1	**1**	1	0	1	1	1	0.2	0.2
	2	1	0	1	0	1	0.0	0.0
	3	0	0	1	1	1	0.25	0.25
	4	0	1	1	0	1	0.25	0.25
							***fitness* do documento**	**0.175**

Estes cálculos são feitos para todos os documentos considerados relevantes pelo usuário. O valor do "*fitness* do documento" pode ser utilizado no ordenamento do conjunto de documentos resultante da busca. Os valores do *fitness* são utilizados para construir uma "roleta" que fornecerá a base para o processo de seleção: para cada cromossomo é calculado o

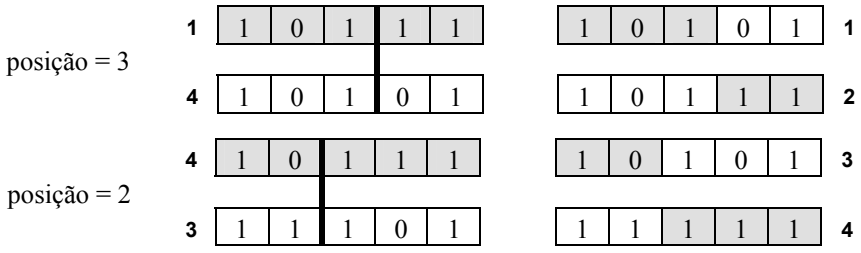
percentual do *fitness* em relação ao total. Portanto, cada cromossomo terá chance de reprodução proporcional ao seu *fitness*:

		Cromossomo	*fitness*	percentual
Doc_1	1	10111	0.2	28.6%
	2	10101	0.0	-
	3	00111	0.25	35.7%
	4	01101	0.25	35.7%
		total	0.70	100%



O cromossomo 2, que possui *fitness* igual a zero, não terá representação na roleta e não ser reproduzirá. Os documentos com maior *fitness* terão mais chances de se reproduzir e transmitir seus genes para as próximas gerações.

A roleta será "girada" quatro vezes a fim de selecionar dois casais de cromossomos para reprodução. Para cada casal o *crossover* é executado utilizando uma posição escolhida aleatoriamente. Supondo-se que para o documento Doc_1 foram escolhidos os casais 1-4 e 4-3, e as posições 3 e 2, respectivamente, o *crossover* será executado da seguinte forma:



Após a reprodução, o documento Doc_1 será representado por quatro novos cromossomos, apresentados abaixo.

Doc_1					
1	1	0	1	0	1
2	1	0	1	1	1
3	1	0	1	0	1
4	1	1	1	1	1

Como observado anteriormente, a capacidade dos algoritmos genéticos provém da diversidade. As mutações ajudam a prevenir a estagnação das populações, ajudando a preservar esta diversidade através das gerações.

Após a reprodução será selecionado aleatoriamente um conjunto de cromossomos que deverá sofrer mutação. Para cada cromossomo será escolhida, também aleatoriamente, a

posição (o gene) onde esta mutação será efetuada. Utilizando ainda o documento Doc_1 como exemplo, e supondo terem sido escolhidos os cromossomos 4 e 1 e os respectivos genes 3 e 4, a mutação será processada da seguinte forma:

posição = 3	**4**	1	1	1	1	1	→	1	1	**0**	1	1
posição = 4	**1**	1	0	1	0	1	→	1	0	1	**1**	1

O processo de mutação deve obedecer a certos critérios. Um índice de mutação muito alto destruirá os indivíduos mais adaptados, impedindo uma rápida evolução da população. Após a operação e mutação, o documento Doc_1 será descrito por um novo conjunto de cromossomos, apresentado abaixo:

Doc_1						
	1	1	0	1	1	1
	2	1	0	1	1	1
	3	1	0	1	0	1
	4	1	1	0	1	1

Fecha-se assim um ciclo da evolução do *corpus*, exemplificado através do documento Doc_1. Assim como o Doc_1, todos os documentos do *corpus* terão o seu "código genético" modificado em função da expressão de busca do usuário.

Posteriormente, em uma nova busca expressa pela seqüência `10011`, por exemplo, o documento Doc_1 terá os seguintes valores:

expressão de busca: `01010` `10011`

								fitness
Doc_1	**1**	1	0	1	1	1	0.2	0.75	(0.2+0.75)/2=**0.475**
	2	1	0	1	1	1	0.0	0.75	(0.0+0.75)/2=**0.375**
	3	1	0	1	0	1	0.25	0.5	(0.25+0.5)/2=**0.375**
	4	1	1	0	1	1	0.25	0.75	(0.25+0.75)/2=**0.5**
								***fitness* do documento**	**0.43125**

O novo valor do *fitness* de cada cromossomo é calculado através da média aritmética do *fitness* das diversas buscas realizadas. Para o documento Doc_1, o grau de adaptação do cromossomo **1** após a primeira busca foi 0.2 e para essa segunda busca é de 0.75. Portanto, o novo valor do *fitness* desse cromossomo será calculado pela média entre 0.2 e 0.75, o que resulta 0.475. Esse cálculo é feito para todos os cromossomos de todos os documentos do *corpus*. O ***fitness* do documento** é calculado através da média dos *fitness* dos cromossomos que representam o documento.

A aplicação dos algoritmos genéticos na recuperação de informação se apresenta apenas como uma possibilidade, uma proposição para futuras implementações de sistemas com características evolutivas. Os trabalhos práticos disponíveis na literatura apresentam apenas testes utilizando pequenos protótipos de sistemas, não determinando sua aplicabilidade em sistemas reais (Gordon, 1988; Vrajitoru, 2000). Apesar da característica evolutiva representar uma forma inovadora de abordar o problema da recuperação de informação, introduz diversos questionamentos relacionados aos efeitos de sua inerente imprevisibilidade quando utilizado em situações reais.

5.4 Conclusão

Os modelos aqui denominados "dinâmicos" representam um enfoque diferenciado em relação aos modelos quantitativos, dando ao conjunto de usuários uma participação ativa na representação dos documentos. Se por um lado essa característica se mostra atrativa, por outro lado restringe sua utilização a pequenos grupos de usuários com interesses comuns ou ao desenvolvimento de "filtros" de informação personalizados (Morgan e Kilgour, 1996). A utilização em grandes comunidades de usuários, com interesses variados, resultaria em uma dispersão das representações dos documentos, eliminando a principal vantagem desses modelos.

A complexidade de implementação dos modelos dinâmicos deixa dúvidas sobre sua aplicabilidade em grandes *corpora*. A maioria dos experimentos apresentados em livros ou artigos utiliza um ambiente controlado, com um conjunto reduzido de documentos. Tais experimentos dão ênfase à observação da evolução das representações dos documentos após um determinado número de interações dos usuários. Portanto, o desempenho computacional desses modelos em situações reais pode ser considerado ainda uma incógnita.

6
Processamento da Linguagem Natural

O Processamento da Linguagem Natural (PLN) surge como uma possível solução aos problemas relacionados à recuperação de informação pela simples observação de que os documentos e as expressões de busca são objetos lingüísticos. O PLN é um conjunto de técnicas computacionais para a análise de textos em um ou mais níveis lingüísticos, com o propósito de simular o processamento humano da língua.

O desenvolvimento de sistemas de recuperação de informação que podem "entender" os documentos exige técnicas computacionais de grande complexidade. Por esta razão, na maioria das vezes as técnicas de PLN são utilizadas apenas na melhoria do desempenho de algumas tarefas da recuperação de informação tradicional, como a indexação automática (Faloutsos e Oard, 1995).

Liddy (1998) classifica as técnicas de PLN de acordo com o nível da unidade lingüística processada: fonológico, morfológico, lexical, sintático, semântico, discurso e pragmático.

O nível *fonológico* é o nível da interpretação dos sons da fala, os fonemas. Ele é de maior interesse na implementação de sistemas de reconhecimento da fala onde é possível o usuário exprimir verbalmente sua busca ou receber alguma forma de resposta audível (Jones et al, 1996; Hauptmann et al, 1998).

O nível *morfológico* está relacionado com a análise de formas variantes de uma determinada palavra através de seus componentes como prefixos, radicais e sufixos. Exemplos de processamento morfológico na recuperação de informação são as técnicas tradicionais de extração de radicais (*stemming*), que visam substituir a variante de uma palavra a uma forma normalizada.

O nível *léxical* trata da análise da estrutura e significado da palavra. Um exemplo de processamento lexical nos sistemas de recuperação tradicionais é a construção de listas de palavras de pouco valor semântico como artigos e preposições. O nível lexical está relacionado com a geração e uso de vocabulários controlados na indexação de documentos e para a formulação e expansão de expressões de busca.

No nível *sintático* busca-se determinar a estrutura sintática das frases de um texto. Por causa da enorme diversidade de estruturas frasais, a determinação precisa da estrutura de uma frase requer conhecimento de alto nível a um custo computacional relativamente alto. Por este motivo o processamento sintático é pouco utilizado na recuperação de informação tradicional.

O nível *semântico* busca interpretar o significado não só de palavras individuais, mas também de expressões ou frases. A resolução de ambigüidades de palavras é uma tarefa do nível semântico (e não do sintático) porque tais ambigüidades muitas vezes só podem ser solucionadas no contexto de uma unidade textual maior como a frase ou o parágrafo onde a palavra está posicionada. Algumas vezes a ambigüidade só pode ser solucionada através de um conhecimento do mundo real, seja ele genérico ou específico do domínio.

Para os objetivos da recuperação de informação, o nível *discursivo* examina a estrutura e os princípios organizacionais de um documento "*para entender qual é função específica de uma informação em um documento, por exemplo – é uma conclusão, é uma opinião, uma previsão ou um fato?*" (Liddy, 1998, p.16).

O nível *pragmático* utiliza conhecimentos externos aos documentos e às buscas do sistema. Este conhecimento pode ser um conhecimento geral do mundo, conhecimento específico para um determinado domínio ou ainda conhecimento sobre as necessidades dos usuários, preferências e objetivos na formulação de uma determinada expressão de busca.

Nas subseções seguintes será discutida a utilização do PLN em alguns problemas clássicos da recuperação de informação. Deve ser ressaltado que, quase sem exceção, os

métodos de PLN discutidos a seguir são utilizados em conjunto com os modelos quantitativos (ou clássicos) (Lewis e Jones, 1996).

6.1 Normalização de variações lingüísticas

O reconhecimento de variações lingüísticas encontradas em um texto permite, por exemplo, o controle de vocabulário (Jacquemin, Klavans e Tzoukermann,1997). A normalização lingüística pode ser subdividida em três casos distintos: *morfológica*, *sintática* e *léxico-semântica*.

A **normalização morfológica** ocorre quando há redução dos itens lexicais através de *conflação* a uma forma que procura representar classes de conceitos. *Conflação* ("*conflation*") é a operação que combina a representação de dois ou mais termos em um único, reduzindo variantes de uma palavra a uma única forma.

Os procedimentos mais conhecidos para *conflação* são:

- *stemming*, reduz uma palavra ao seu radical (*stem*) através da eliminação de afixos oriundos de derivação ou de flexão (Orengo e Huyck, 2001);
- redução à forma canônica, processo também conhecido como *lematização* ("*lemmatization*"), que geralmente reduz os verbos ao infinitivo e os adjetivos e substantivos à forma masculina singular (Arampatzis, 2000).

No caso da forma canônica a categoria morfológica original da palavra é preservada. Já o processo de *stemming* pode resultar palavras de categorias diferentes. Por exemplo, "construção" e "construiremos" seriam reduzidas a "constru", no processo de *stemming*. Utilizando a forma canônica teríamos, respectivamente, "construção" e "construir".

A **normalização sintática** ocorre quando há a normalização de frases semanticamente equivalentes em uma forma única e representativa das mesmas, como "trabalho eficiente e rápido" e "trabalho rápido e eficiente".

A **normalização léxico-semântica** ocorre quando são utilizados relacionamentos semânticos (como a sinonímia, hiponímia) entre os itens lexicais para criar um agrupamento de similaridades semânticas, identificado por um item lexical que representa um conceito único.

Podem-se encontrar duas formas de normalização lexical. De um lado está a normalização morfológica através do processo de *stemming*, que explora similaridades morfológicas. Em outro extremo está a normalização léxico-semântica, por exemplo, através de busca de sinônimos em tesauros, considerando informações terminológicas.

6.2 Identificação de termos compostos

Em um sistema de recuperação de informação geralmente termos compostos são identificados para que possam também ser usados como termos de indexação, não se limitando à utilização de palavras isoladas. Será adotado a expressão "termo composto" para identificar indistintamente "sintagmas", "termos complexos" ou "unidades lexicais complexas".

Os termos compostos geralmente apresentam menor ambigüidade e maior especificidade do que os itens lexicais simples, permitindo uma maior aproximação com o seu significado expresso no texto onde ocorrem. Nos sistemas de recuperação de informação os termos compostos são geralmente identificados através de cálculos de co-ocorrência de pares de palavras. A utilização da análise sintática permite identificar termos compostos mesmo quando as palavras que compõem o termo não são adjacentes ou não co-ocorrem com grande freqüência. Segundo Croft, Turtle e Lewis (1991), a extração de termos compostos por meios puramente sintáticos freqüentemente não apresenta bons resultados. Uma combinação de técnicas de PLN com métodos estatísticos é mais eficaz (Lewis, 1992).

Lewis e Jones (1996) notam que o grau de sofisticação do PLN poderia ser consideravelmente maior para as expressões de busca dos usuários do que para os documentos. Um motivo para isso seria a grande dificuldade no processamento dos documentos de um *corpus* em relação a uma expressão de busca. Outro motivo seria a importância em entender quais são as necessidades do usuário; visto que geralmente as suas expressões de busca são muito mais curtas (com poucos termos). Eventuais erros no processamento dos documentos podem ser corrigidos (ou pelo menos compensados) levando em conta outros termos extraídos do mesmo documento, o que não é possível para uma expressão de busca.

Algumas técnicas comuns na recuperação de informação, como a utilização de listas de *stopwords* e a normalização das variações lingüísticas dos termos (como visto

anteriormente), podem dificultar o julgamento de relevância de um documento ou descontextualizar um determinado termo (Riloff, 1995). Por exemplo, a presença do termo "morto" em um documento não garante que o documento descreva um assassinato. Porém, a frase "morto a tiros" possui uma conotação de crime. A presença do termo "assassinato" (singular) em um documento é um indicador de que o documento descreve um assassinato específico. Já a presença do termo "assassinatos" (plural) pressupõe que o texto descreva diferentes assassinatos ou fale sobre assassinatos de uma forma geral. Preposições, formas verbais, afirmações positivas ou negativas, podem ser significantes para determinar o sentido de uma frase. Um exemplo apresentado por Riloff mostra que o termo "*venture*" (aventura, risco, iniciativa, aventurar-se) tomado isoladamente não é um bom termo de indexação para um documento que descreve um empreendimento conjunto entre empresas ("*joint venture*"). No entanto o termo composto "*venture with*" ou "*venture between*" seriam bons termos de indexação, já que as preposições *with* e *between* indicam uma noção de parceria.

6.3 Resolução de ambigüidade

A ambigüidade é a propriedade que faz com que um objeto lingüístico, seja uma palavra, um termo composto ou todo um texto, possa ser interpretado de modos diferentes. Quanto ao nível de processamento existem dois tipos de ambigüidade: sintática e semântica. A ambigüidade sintática ocorre quando um item lexical pode pertencer a mais de uma classe gramatical, como "casa" que pode ser substantivo ou verbo. Outras causas da ambigüidade sintática são: mais de uma ligação possível do sintagma preposicional, como em "comprei um cofre com dinheiro"; a possibilidade de mais de uma coordenação ou conjunção, como em "tenho amigos e parentes muito queridos"; ou a possibilidade de múltiplas combinações para substantivos compostos, como em "lareira da casa de pedras" (Smeaton, 1997).

Um exemplo de ambigüidade semântica é a que ocorre com o verbo "passar", que pode apresentar mais de um significado, como em "passar a ferro", "passar no exame" e "passar em casa".

As causas da ambigüidade podem ser dos seguintes tipos (Beardon, Lumsden e Holmes, 1991):

- **lexical**, que ocorre quando uma palavra pode possuir múltiplos significados;

- **estrutural**, quando é possível mais de uma estrutura sintática para a sentença, podendo ser: **local**, quando a ambigüidade pode ser resolvida dispensando o conhecimento do contexto onde ela ocorre; ou **global**, quando exige análise do contexto para sua resolução.

Por exemplo, na frase "ele olhou o computador com esperança" existe uma ambigüidade estrutural local. Neste caso o sentido expresso pela frase "computador com esperança" pode, em princípio, ser descartada. Em "ele olhou o colega com esperança" há ambigüidade estrutural global, sendo possível construir duas associações diferentes: "olhou com esperança" e "colega com esperança".

Em relação à ambigüidade lexical deverão ser ressaltados dois fenômenos lingüísticos: a *homonímia* e a *polissemia* (Krovetz, 1997; Krovetz e Croft, 1992).

A *homonímia* ocorre entre itens lexicais com significados diferentes que possuem o mesmo som e a mesma grafia (homônimos perfeitos: como substantivo "alvo" e adjetivo "alvo"), ou apenas o mesmo som (homônimos homófonos: como "acento" e "assento"), ou apenas a mesma grafia (homônimos homógrafos: como o verbo "seco" e o adjetivo "seco") (Sacconi, 1999). Os homônimos homógrafos podem existir por possuírem origem comum (o adjetivo "triangular" e o verbo "triangular"), por coincidência ("vogal", a letra, e "vogal", um membro de júri) ou por derivação (substantivo "procura", derivado do verbo procurar) (Santos, 1996).

No caso da *polissemia* uma mesma palavra pode adquirir diferentes significados, como no caso da palavra "banco".

Ainda quanto à ambigüidade lexical, pode-se notar que alguns sentidos de algumas palavras são mais específicos do que outros. Esta propriedade recebe o nome de "*vagueness*" (imprecisão, incerteza) (Allen, 1995). Assim, diferentes significados produzem diferentes graus de incerteza. Por exemplo, dependendo do contexto, quando é usada a palavra "cavalos" pode-se ter incerteza quanto à raça desses animais; já a palavra "banco" pode produzir incerteza maior, podendo se tratar de uma instituição ou um móvel, entre outros significados. Conforme o grau de incerteza, a ambigüidade pode até ser insignificante, dependendo do contexto da sentença.

Quanto à ambigüidade estrutural, as suas principais causas são problemas de localização dos sintagmas preposicionais, adverbiais ou das orações relativas na estrutura de uma sentença.

A resolução da ambigüidade lexical pode ter uma abordagem cognitiva ou lingüística. A primeira procura investigar como fatores semânticos, sintáticos e neuropsicológicos podem contribuir na resolução deste tipo de ambigüidade. A abordagem lingüística considera estratégias em nível sintático e semântico. Em nível sintático, são levadas em consideração as vizinhanças da palavra ambígua. Já a abordagem semântica considera metodologias para representação do conhecimento sobre os itens lexicais, necessitando especificar contextos ou domínios restritos.

A resolução da ambigüidade sintática requer decidir sobre diversas estruturas prováveis que representam sintaticamente a sentença analisada. Em alguns casos, somente restrições semânticas podem auxiliar a resolução da ambigüidade sintática.

No contexto da Recuperação de Informação, Krovetz (1997) defende três hipóteses relacionadas à ambigüidade lexical:

- **Hipótese 1**. A resolução da ambigüidade lexical beneficia o desempenho da recuperação de informação;
- **Hipótese 2**. Os significados das palavras determinam uma separação entre os documentos relevantes e não relevantes;
- **Hipótese 3**. Mesmo em um *corpus* pequeno e de domínio específico, há uma proporção significante de ambigüidade lexical.

A resolução automática de ambigüidade constitui um problema complexo. As abordagens para a resolução de ambigüidade na Recuperação de Informação podem ser divididas em duas categorias principais:

- baseadas em regras de co-ocorrência ou de padrões sintáticos;
- baseadas em informações oriundas do *corpus*, de dicionários ou de tesauros.

Gauch e Futrelle (1994) usam uma combinação de informações para estabelecer similaridades entre itens lexicais e definir classes de palavras. Estas classes são utilizadas para resolver ambigüidades de palavras da língua inglesa terminadas em *ed*, indicando se são verbos no particípio passado ou adjetivos.

Krovetz (1997) considera informações provenientes de dicionários como morfologia, categoria gramatical e composição de termos como fontes de evidência para a resolução de ambigüidades. Krovetz parte do princípio segundo o qual as palavras podem diferir em morfologia (exemplo: "autorizo" e "autorizei"), em categoria gramatical (exemplo: "diabético", como substantivo ou adjetivo) ou quanto à capacidade de ocorrer em termos compostos (exemplo: "base de dados"), representando diferentes conceitos. Tais diferenças são consideradas associadas às diferenças em significados e, em virtude disto, deve-se estabelecer associações entre tais variações. Para atacar o problema, é explorada a presença de variantes de um termo na definição deste termo no dicionário, além de serem utilizadas sobreposições de palavras em definições supostamente variantes.

Kaji et al (2000) procuram resolver a ambigüidade de sintagmas nominais aliando estatística ao PLN. A ambigüidade tratada ocorre quando um sintagma nominal pode ser interpretado como $P_1(P_2P_3)$ ou como $(P_1P_2)P_3$, como, por exemplo, "casa de bairro grande", em que podemos ter o adjetivo "grande" modificando "casa" ou "bairro". Utilizando uma regra simples, a estrutura é determinada através da freqüência: se o componente P_2P_3 ocorre mais freqüentemente, então a estrutura $P_1(P_2P_3)$ será a preferida; caso contrário $(P_1P_2)P_3$ será a escolhida.

6.4 Conclusão

O Processamento da Linguagem Natural (PLN) não se caracteriza como um *modelo* de recuperação de informação, na medida em que não propõe uma estrutura para a *representação* dos documentos e não formaliza explicitamente uma *função de busca*, como apresentado no Capítulo 3. Porém, é através do PLN que a Recuperação de Informação se aproxima do arsenal metodológico da Inteligência Artificial e viabiliza soluções para alguns de seus problemas.

Obviamente, espera-se que as técnicas de PLN se mostrem mais efetivas nas etapas do processo de recuperação de informação em que a qualidade dos resultados depende de uma interpretação adequada das entidades textuais, que são, por um lado, os documentos do *corpus* e, por outro lado, a expressão de busca do usuário, assumindo que esta seja enunciada em linguagem natural.

O PLN aplicado às expressões de busca de um sistema de recuperação de informação assume uma importância considerável na medida em que tenta interpretar a necessidade de informação dos usuários. Porém, essa tarefa é dificultada pelo tamanho (número de palavras) reduzido das expressões de busca que geralmente são utilizadas pelos usuários, não permitindo uma interpretação adequada das expressões.

A utilização mais importante do PLN está, portanto, na interpretação do conteúdo dos documentos, a fim de gerar uma representação adequada destes. No entanto, o PLN não elimina a necessidade da utilização de métodos estatísticos e deve ser visto como uma ferramenta complementar aos mesmos.

Os procedimentos envolvidos no PLN estão geralmente restritos a uma determinada língua como o inglês, o alemão ou, em menor proporção, o português. Essa limitação, aliada ao custo relativamente alto do PLN, é um fator que diminui sua atratividade, considerando que os métodos estatísticos (quantitativos) envolvem menor custo e geralmente são adaptáveis a diversas línguas.

7
Recuperação de Informação na WEB

A historia da Internet é de certa forma uma versão acelerada da história da imprensa, desde o invento de Gutenberg até o *offset*. Essa história pode ser contada a partir da Guerra Fria, período histórico que teve seu início no pós-guerra. Em 1957, em resposta ao sucesso do programa espacial soviético representado pelo lançamento do *Sputnik*, os Estados Unidos criaram o Departamento de Defesa (DoD) e a ARPA (*Advanced Research Projects Agency*). Em 1969, o DoD promoveu a criação de um sistema de comunicações que permitisse interligar computadores dos principais centros da ARPA. Surgiu assim a ARPAnet, uma rede de computadores que deveria continuar funcionando mesmo se algum dos computadores sofresse um ataque nuclear.

A ARPAnet inicialmente interligava quatro centros de computação: a Universidade da Califórnia, em Los Angeles e em Santa Bárbara, o Instituto de Pesquisa de Stanford e a Universidade de Utah, em Salt Lake City. Em 1973 as primeiras conexões internacionais foram montadas, conectando a ARPAnet à *University College* em Londres e ao *Royal Radar Establishment*, na Noruega. A partir de 1975 outras redes foram criadas por instituições de pesquisa e empresas privadas. Essas redes acabaram por criar uma comunidade, que trocava entre si informações através de uma versão primitiva do atual correio eletrônico, embora não houvesse ainda a possibilidade de comunicação entre as diversas redes. No início dos anos 80 a ARPA adotou o TCP/IP (*Transfer Control Protocol / Internet Protocol*), um protocolo que

facilitava a comunicação entre redes de computadores. Com a utilização do TCP/IP por diversas instituições de pesquisa, uma "rede de redes" estava se formando, permitindo que milhares de usuários compartilhassem suas informações: a Internet. Os interesses militares da ARPAnet foram transferidos para uma nova rede, a MILnet, extinguindo-se então a ARPAnet.

Em 1992 a Internet já conectava um milhão de computadores e alcançou áreas comerciais, fora da esfera acadêmica. Foram então criados o ARCHIE (um sistema de busca em arquivos) e o GOPHER (um sistema de busca de informação que utiliza menus e diretórios).

Desde 1989 Tim Berners-Lee começara a desenvolver uma tecnologia para compartilhamento de informação usando documentos textuais que se referenciavam através de ligações. O objetivo inicial era construir uma ferramenta de comunicação baseada na Internet para compartilhar informação com diferentes universidades em todo o mundo. Berners-Lee criou uma linguagem de marcação baseada na já bem sucedida SGML (*Standard Generalized Markup Language*) e batizou-a de HTML (*HyperText Markup Language*). Ele também desenvolveu protocolos de comunicações para formar a espinha dorsal do seu novo sistema de informações em hipertexto, o qual denominou *World Wide Web*, ou simplesmente Web. Em 1994, Berners-Lee fundou o W3C (*World Wide Web Consortium*), uma organização destinada a padronizar e desenvolver tecnologias de domínio público para a Web.

A Web é a face hipertextual da Internet e é hoje considerada como a maior fonte de informação nas principais áreas do conhecimento. O seu uso intensivo aliado ao seu crescimento exponencial vem mudando diversos aspectos da sociedade contemporânea.

7.1 Características da Web

A Web é formada por um conjunto de unidades de informação chamadas “páginas”. Uma página é um arquivo de computador cujo tamanho (quantidade de caracteres) pode variar desde o tamanho de uma página de um livro até o tamanho de um livro inteiro. Essas páginas possuem as seguintes características comuns:

- Esquema de endereçamento chamado *Universal Resource Locator* (URL);
- Protocolo, o *Hypertext Transfer Protocol* (http), que permite que um programa no computador do usuário requisite uma página (através de sua URL) ao computador onde a página está localizada (servidor ou *host*). O servidor responde à requisição enviando uma cópia da página ao computador do usuário;
- Padrão para a especificação da estrutura da página, *Hypertext Markup Language* (HTML), uma linguagem de marcação que permite definir diferentes componentes em uma página Web.

Uma URL é o endereço de um arquivo acessível através da Internet. Como exemplificado na Figura 27, uma URL é uma cadeia de caracteres formada por componentes padronizados, em uma ordem específica.

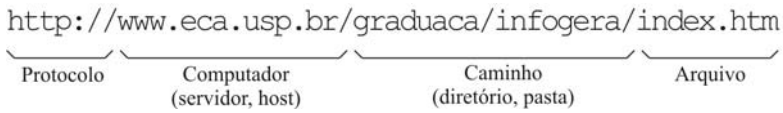


Figura 27 Partes de uma URL

A URL **http://www.eca.usp.br/graduaca/infogera/index.htm** identifica um arquivo que deve ser acessado utilizando o protocolo da Web (**http://**) e que está armazenado no computador chamado **www.eca.usp.br**, e cujo domínio é ("**.br"**), indicando que este computador está localizado no Brasil. No diretório (ou pasta) **/graduaca/infogera/** deste computador está localizado o arquivo com o nome **index.htm**. A extensão "**.htm"** indica que se trata de um arquivo no formato HTML.

Atualmente a maioria das páginas Web está escrita na linguagem HTML. Esta linguagem de marcação possui um conjunto pré-definido de códigos chamados *tags* usados para definir componentes relacionados com a aparência e com a funcionalidade das páginas como título, autor, resumo, figuras, etc. Uma página HTML pode conter *tags* que especifiquem URLs de outras páginas. Essas ligações (*links*) entre páginas formam uma estrutura de complexidade arbitrária, o que explica o uso do termo "*Web*" (teia). A Figura 28 mostra o conteúdo de um arquivo HTML e o resultado de sua apresentação em um programa de navegação na Web, conhecido como *Browser*.

```
<html>
 <header>
   <title>O Mundo é Grande</title>
 </header>
 <body>
  <b>O Mundo é Grande</b><p>
  O mundo é grande e cabe<p>
  nesta janela sobre o mar.<p>
  O mar é grande e cabe<p>
  na cama e no colchão de amar.<p>
  O amor é grande e cabe<p>
  no breve espaço de beijar.<p>
  <b>Carlos Drummond<b>
  <a href=http://www.carlosdrummond.com.br>Home Page
 </body>
</html>
```

O Mundo é Grande

O mundo é grande e cabe
nesta janela sobre o mar.
O mar é grande e cabe
na cama e no colchão de amar.
O amor é grande e cabe
no breve espaço de beijar.

Carlos Drummond Home Page

Figura 28 Exemplo de um arquivo HTML e sua visualização

A linguagem HTML possui um conjunto fixo de *tags* que permitem a definição da aparência da página. Um documento HTML é um arquivo textual puro, que pode ser criado a partir de qualquer editor de texto. Apesar de sua simplicidade, a linguagem HTML permite a utilização de um grande número de recursos, como a criação de páginas com várias janelas (*frames*), a utilização de imagens e tabelas e a definição de ligações entre páginas Web.

O arquivo HTML apresentado na Figura 28 possui uma ligação (hiperlink ou *link*) para a página de URL **http://www.carlosdrummond.com.br**. O fato de uma página Web poder apontar para outra página pressupõe algum tipo de semelhança entre essas páginas. Porém, não existe uma regra geral que assegure tal semelhança.

Embora a Web, tipicamente baseada em páginas HTML, não imponha qualquer estruturação semântica, é possível um agrupamento conceitual de páginas a partir de determinados pontos de vista. Uma página P_1 pode referenciar uma outra página P_2 por esta tratar do mesmo assunto de P_1. Em P_1 pode existir também uma referência à página P_3 por esta tratar de um aspecto especial do assunto tratado em P_1. Desta forma, as URLs podem criar uma elaborada rede de citações por assunto, autor, instituição, etc.

Os *links* são unidirecionais, consistem de pares virtuais (não estão fisicamente armazenados) de URLs de origem e destino e são inseridos no corpo das próprias páginas. Conseqüentemente, não é possível, por exemplo, determinar quais são as páginas que referenciam uma página específica. Segundo pesquisadores da área de hipertexto, uma solução para esse problema seria a especificação (cadastramento) dos *links* num contexto exterior e independente das páginas, o que, por um lado, implicaria na criação de servidores e

bases de dados de ligações, mas por outro lado acrescentaria uma nova dimensão aos recursos de busca da Web (Kappe, 1991; Andrews, Kappe e Maurer, 1995). Nesta perspectiva, a Web, além de disponibilizar informações, permitiria também a gestão das relações entre essas informações. O deslocamento dos nós da rede para as ligações entre os nós aponta para uma visão extremamente contemporânea dos sistemas de acesso à informação na medida em que incorpora o mutável (as ligações entre informações) ao fixo (acervo de informações disponíveis).

Uma URL pode apontar para um arquivo que não está no formato HTML. Neste caso, será necessário algum programa adicional para apresentar o conteúdo desse arquivo: um editor de texto, um programa gráfico, etc. Uma URL pode referenciar também um arquivo que não pode ser acessado através do protocolo HTTP pelo fato de o computador onde o arquivo está armazenado não ser um servidor Web. Neste caso algum outro tipo de servidor deve ser usado para recuperar o arquivo. O servidor não-Web mais comum é o FTP.

Um grande número de arquivos textuais ou binários (imagens, sons, vídeos, etc.) estão disponíveis para transferência (*download*) através de um servidor FTP (*File Transfer Protocol*). Os arquivos em um *site* FTP geralmente estão organizados em uma estrutura hierárquica de diretórios (ou pastas) e arquivos. Esta estrutura pode ser visualizada por um usuário da Internet através de um *browser*. Qualquer arquivo que o usuário achar interessante (talvez pelo nome desse arquivo), pode ser recuperado. Esses arquivos não são páginas Web, e, portanto, não contêm *links* para outras páginas ou arquivos. A única forma de busca que pode ser feita em um *site* FTP é a navegação em sua estrutura hierárquica. Na Figura 29 é apresentado o diretório inicial do servidor FTP do Instituto de Física da USP (**ftp://ftp.if.usp.br**)

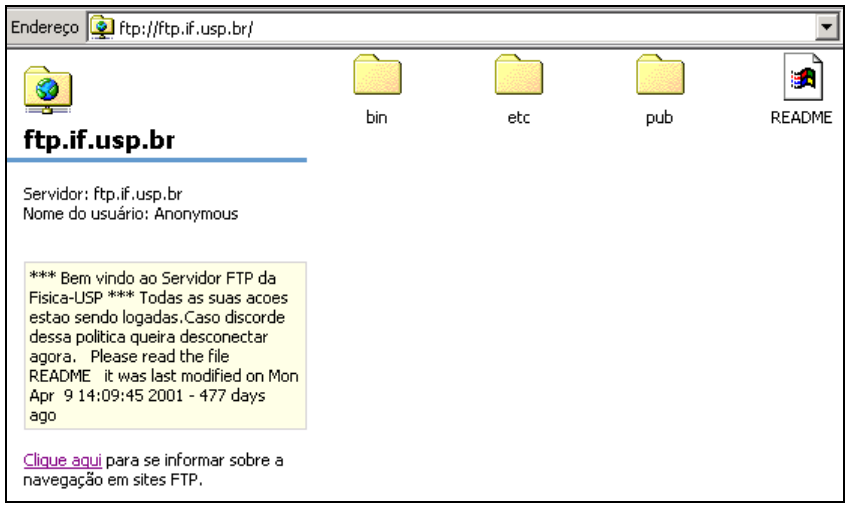


Figura 29 Diretório de um servidor FTP apresentado em um *Browser*

7.2 Mecanismos de busca

Grande parte dos mecanismos de busca encontrados na Web é de uso geral. Esses mecanismos, chamados de *search engines*, *sites de busca* ou *portais*, permitem ao usuário submeter sua expressão de busca e recuperar uma lista (geralmente ordenada) de endereços de páginas (URLs) que presumivelmente são relevantes para a sua necessidade de informação.

Em um acervo extremamente grande como é a Web é essencial uma indexação antecipada de seus documentos (páginas). A maioria dos mecanismos de busca da Web gera índices. Pelo caráter dinâmico da Web esses índices devem permanecer em constante processo de atualização. Existem duas alternativas básicas para a criação de índices:

- O índice pode ser construído manualmente por indexadores profissionais. A vantagem óbvia está na utilização da insubstituível capacidade humana em julgar relevância e categorizar documentos, refletindo diretamente na qualidade do índice gerado e, conseqüentemente, na precisão da recuperação, desde que exista algum tipo de controle de vocabulário.
- O índice pode ser gerado automaticamente, permitindo uma cobertura mais ampla e rápida das páginas Web.

7.2.1 Indexação Manual

Alguns mecanismos de busca empregam indexadores profissionais que especificam uma hierarquia de assuntos, similar às classificações encontradas em uma biblioteca tradicional, como a Classificação Decimal de Dewey (CDD), e indexam as páginas Web utilizando tais categorias.

Um exemplo de mecanismo de busca que utiliza indexação manual é o Yahoo! (www.yahoo.com.br). A eficiência do Yahoo! depende em grande parte de voluntários para obter URLs para seu banco de dados. O autor de uma página Web pode cadastrar a URL de sua página associando a ela uma ou mais categorias que descrevem o assunto tratado na página. No cadastramento da página, o usuário fornece um título, um texto curto descrevendo a página e a URL da página que será cadastrada.

Algumas características do Yahoo! são:

- Cada categoria de assunto é também uma página Web. A página de uma determinada categoria é formada por um conjunto de *links* para as páginas relacionadas àquela categoria e um conjunto de *links* para sub-categorias. A Figura 30 apresenta a página da sub-categoria "Biblioteconomia e Ciência da Informação". A primeira lista de *links* aponta para páginas das sub-categorias. Em seguida é apresentada uma lista de *links* para páginas ou *sites* que estão diretamente ligadas à categoria *Biblioteconomia e Ciência da Informação*;

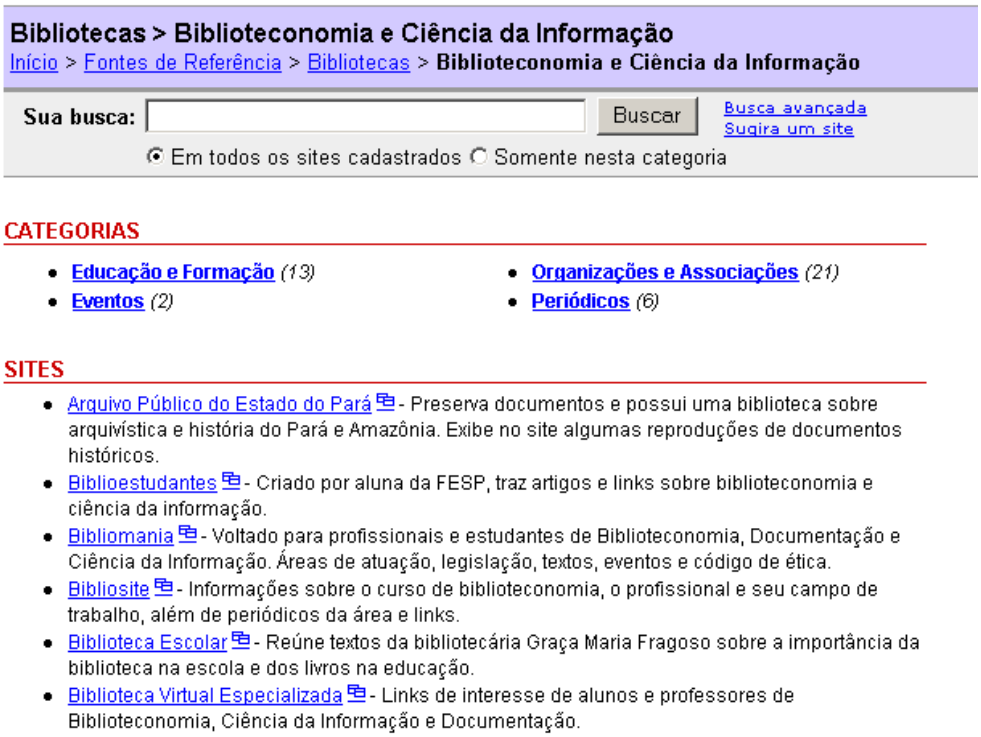


Figura 30 Página Yahoo! referente à categoria *Biblioteconomia e Ciência da Informação*

- Uma URL submetida ao Yahoo! pode ser associada a uma categoria de qualquer nível. Por exemplo, ela pode ser ligada a uma categoria principal, "Ciência", ou à subcategoria, "Ciências Humanas", ou à sub-subcategoria, "Biblioteconomia e Ciência da Informação".
- Os funcionários do Yahoo! avaliam os dados de cada URL cadastrada, podendo alterar os dados fornecidos pelo usuário.
- Caso um usuário não consiga encontrar uma categoria apropriada para descrever sua página, ele pode sugerir uma nova categoria. Os profissionais do Yahoo! podem aceitar, rejeitar ou modificar as sugestões dos usuários.

O método utilizado pelo Yahoo! possui inevitáveis desvantagens mas também muitas vantagens. Já que o Yahoo! depende do cadastramento voluntário de páginas, sua cobertura da Web é inevitavelmente incompleta e irregular. Se o usuário deseja fazer uma busca de um determinado assunto que não se enquadra em qualquer categoria existente, ou é uma combinação de categorias, o resultado obtido não terá a precisão esperada. Por outro lado, se a busca do usuário está relacionada diretamente a uma das categorias existentes, é de se esperar uma alta precisão no resultado. Além disso, uma página indexada pelo Yahoo! geralmente

possui *links* para outras páginas relevantes sobre um determinado assunto, sendo, portanto, um bom recurso para começar uma pesquisa na Web.

7.2.2 Indexação Automática

Outros mecanismos de busca, tais como o AltaVista (www.altavista.com) e o Excite (www.excite.com), indexam automaticamente as páginas da Web. A indexação automática é realizada através de duas etapas:

1. Seleção de endereços (URLs) de páginas;
2. Indexação das páginas, gerando para cada uma um conjunto de termos de indexação.

As páginas Web estão distribuídas em um imenso e dinâmico conjunto de *sites*. Além do texto, cada uma dessas páginas contém um conjunto de *links* que apontam URLs de outras páginas. Existem programas que "viajam" através da Web a fim de selecionar URLs de páginas de potencial interesse para que sejam indexadas. Utilizando a metáfora da Web, esses programas são chamados de *spiders* (aranhas) ou ainda *robôs*, *crawlers* ou *worms*. Partindo de uma lista inicial de URLs, esses robôs rastreiam a estrutura hipertextual da Web colhendo informação sobre as páginas que encontram.

A estrutura da Web é complexa. Diferentes *sites* ou regiões da Web podem estar estruturadas de acordo com princípios organizacionais diferentes. Alguns *sites* podem ter uma estrutura profunda, isto é, com vários níveis de *links*. Outros *sites* podem apresentar uma estrutura mais ampla, com grande número de *links* para páginas de diferentes *sites*. Em *sites* com estrutura profunda um robô, que tente rastrear todas as suas páginas, pode gastar muito tempo para percorrê-los, reduzindo o tempo para percorrer outros *sites*.

Duas estratégias podem ser adotadas pelos robôs para rastrear as páginas da Web: uma chamada *breadth-first* e outra chamada *deep-first*. A primeira visa maximizar a amplitude da pesquisa descendo apenas poucos níveis de cada *site*. A segunda estratégia visa maximizar a profundidade buscando um maior detalhamento do assunto tratado pelo *site*.

Quando uma nova página é recuperada, o robô extrai todas as URLs dessa página e os adiciona na sua base de dados. Para aumentar a velocidade de cobertura da Web podem ser usados vários robôs trabalhando em paralelo, cada um cobrindo uma região ou um domínio diferente da Web e enviando suas URLs para a base de dados.

Um robô salva todas as URLs que descobre. Ele pode usar algumas características da URL ou da própria página para determinar se a página merece ou não ser indexada. Os critérios usados para essa seleção geralmente não são documentados ou tornados públicos pelas empresas. Algumas URLs poderão ser descartadas ou porque apontam para páginas que não existem mais ou porque apontam para páginas protegidas por senha.

Com freqüência um robô poderá descobrir URLs que já fazem parte de seu banco de dados. Portanto, uma importante característica da construção de um banco de dados de URLs é a remoção de URLs duplicadas. Um problema adicional é que uma mesma página pode ser replicada em diversos *sites* ou um mesmo *site* pode ser referenciado por várias URLs diferentes (apelidos). Assim, não é suficiente eliminar URLs duplicadas; é importante reconhecer se duas páginas acessadas por diferentes URLs são idênticas. Existem algoritmos que permitem detectar semelhanças não só entre páginas Web, no formato HTML, mas também entre arquivos de formatos diferentes.

Depois de formado o banco de dados de URLs o robô poderá acessar cada página e indexá-la usando métodos de indexação automática. Esses métodos de indexação também não são revelados pelas empresas, o que compromete a avaliação do processo de recuperação dos *sites* de busca.

Um recurso adicional na indexação das páginas Web é a utilização das *tags* para restringir a indexação das páginas a determinados componentes, ou ainda para atribuir pesos diferentes a termos localizados em diferentes componentes da página. Por exemplo, poderia ser dado um peso maior a uma palavra localizada entre as marcas de título (**`<title> </title>`**).

Os diferentes mecanismos de busca baseados em robôs podem variar no tipo de páginas que indexam. Como foi dito anteriormente, nem todas as URLs apontam páginas Web, formatadas em HTML. Alguns mecanismos indexam também páginas Gopher, FTP ou páginas de texto simples, não formatado.

Apesar de sua pretensa modernidade, sabe-se que grande parte dos mecanismos de busca utiliza técnicas de indexação desenvolvidas nos anos 60. Alguns utilizam *stop lists* para eliminar palavras comuns, de pouco valor semântico como preposições, artigos, conjunções, etc. Outros utilizam técnicas estatísticas ou processamento de linguagem natural para atribuir

pesos às palavras. Existem também mecanismos que utilizam técnicas de extração de radicais (*stemming*) para normalizar os termos de indexação.

A maioria dos mecanismos de busca constrói e armazena um resumo de cada página em suas bases de dados. Em muitos casos este resumo é formado por uma quantidade fixa de palavras ou caracteres a partir do início do texto.

O AltaVista indexa os termos de uma página pela posição relativa no componente HTML onde o termo aparece. Isto permite elaborar expressões de busca utilizando operadores de proximidade e buscas restritas a um determinado componente ou área da página.

Apesar da variedade de critérios usados pelos mecanismos de busca para construir seus índices, os termos de indexação são na maioria das vezes palavras ou frases contidas nas páginas. O Excite difere da maioria dos mecanismos por utilizar um método de indexação chamado *Latent Semantic Indexing* (LSI) que cria um índice de conceitos, estatisticamente derivados por co-ocorrência de suas palavras (Deerwester et al, 1990).

7.2.3 Especificação de busca

A maioria dos *sites* de busca dispõe de dois níveis de especificação da expressão de busca: básico e avançado. O nível básico permite geralmente a utilização de palavras combinadas logicamente por operadores booleanos. A maioria dos sites permite também a definição de frases através da delimitação de uma seqüência de palavras utilizando aspas.

Além das buscas booleanas, o nível avançado oferece recursos mais sofisticados. O WebCrawler (www.webcrawler.com), por exemplo, oferece os operadores NEAR e ADJ. Uma expressão do tipo "***a*** NEAR/*n* ***b***" especifica que o termo ***a*** e o termo ***b*** deve ter *n* palavras entre elas. A expressão "***a*** ADJ ***b***" especifica que a palavra ***a*** deve aparecer seguida da palavra ***b***, nesta ordem. No AltaVista o operador NEAR não permite que o usuário especifique a proximidade. A expressão de busca "***a*** NEAR ***b***" retornará URLs de páginas onde aparecem as palavras ***a*** e ***b*** com no máximo 10 palavras entre ***a*** e ***b***.

Alguns mecanismos de busca utilizam listas de palavras de pouco valor semântico como artigos e preposições, embora geralmente essas listas de palavras não sejam disponibilizadas. Outros mecanismos geram essas listas estatisticamente com palavras que são encontradas com muita freqüência nas páginas Web. Existem mecanismos que permitem a utilização de "máscaras". No Altavista, por exemplo, é possível utilizar o asterisco ("*") no

início e final de uma palavra, ou no meio, desde que precedido de pelo menos três caracteres. Assim, a expressão "livr*" pode encontrar URLs de páginas onde apareçam as palavras "livro", "livraria", "livreiro". A expressão "inter*ção" recuperará páginas que possuem a palavra "intervenção", "internacionalização", "interação", "interseção", ou qualquer outra palavra que comece com "inter" e termine com "ção". Em alguns casos pode-se querer especificar que a busca deve ser feita utilizando a palavra exatamente da forma como foi informada, sem admitir derivações. No Lycos, por exemplo, ao elaborar a expressão de busca é possível utilizar o caractere ponto (".") no final de uma palavra para indicar que a busca deve se limitar à palavra, sem derivações. Assim, uma expressão de busca com a palavra "escolar" seguida de um ponto, encontrará referências que exatamente a palavra "escolar" e não "escolaridade", por exemplo.

A estrutura da Web permite a implementação de alguns recursos que consideram sua organização. O Hotbot (www.hotbot.com), por exemplo, permite restringir a busca a um determinado domínio, como por exemplo ".br" para especificar páginas localizadas no Brasil ou ".edu" para restringir a busca a páginas de entidades educacionais. O Hotbot também permite restringir a busca às páginas que contenham arquivos de um determinado tipo de mídia. Por exemplo, uma busca utilizando a expressão "biblioteca AND usp" e *page content* ".jpg" recuperará páginas nas quais aparecem as palavras "biblioteca" e "usp" e que contenham alguma imagem do tipo JPEG.

Como resultado de uma busca, o *site* apresenta uma lista ordenada de endereços de páginas (URLs) que atendem à expressão de busca. Esse ordenamento é feito através da utilização de algum método de cálculo efetuado entre a expressão de busca e o conteúdo da página, como nos modelos discutidos no Capítulo 3. Os primeiros itens que aparecem na lista são os que presumivelmente possuem maior relevância para a necessidade de informação do usuário. Dada a grande quantidade de páginas que podem ser recuperadas, é quase imprescindível a utilização de alguma forma de ordenamento. As empresas não divulgam os métodos (algoritmos) utilizados para esse ordenamento; sabe-se, porém, que alguns *sites* dão peso maior para os termos menos comuns na Web. Alguns *sites* atribuem maior importância (peso) aos termos da expressão de busca que aparecem em determinadas posições da página. O Lycos e o InfoSeek dão peso maior aos termos que aparecem no título de uma página. O AltaVista dá peso maior quanto mais próximo do início da página um termo estiver

localizado. O InfoSeek, o AltaVista e o HotBot atribuem pesos aos termos baseados na freqüência com que eles ocorrem na página.

Através do processo denominado *relevance feedback* (realimentação por relevância) o usuário identifica, no conjunto de documentos inicialmente recuperados, algum subconjunto de documentos que são relevantes. O sistema então extrai os termos comuns a esse subconjunto de documentos e os acrescenta na expressão de busca, refinando-a. Esse processo, também conhecido como busca por documentos similares, pode ser repetido várias vezes até que o usuário consiga um conjunto de documentos que o satisfaça. O problema central desse processo está na seleção de características comuns dos documentos relevantes e o cálculo de pesos para tais características no contexto da nova busca. Os mecanismos de busca da Web fornecem uma forma limitada de *relevance feedback*, permitindo ao usuário escolher uma página que atenda às suas necessidades e comande a busca de páginas semelhantes à mesma. O Google (www.google.com.br), após apresentação dos itens resultantes de uma busca, permite especificar uma nova expressão e efetuar a busca apenas nesses itens recuperados.

Como mencionado anteriormente, o Excite indexa suas páginas utilizando um método chamado *Latent Semantic Indexing*. Esse método de indexação acrescenta ao Excite alguns recursos de busca diferenciados. Uma busca utilizando a expressão "financiamento especial a pequenas empresas", por exemplo, ao invés de recuperar apenas documentos que contenham cada uma destas palavras ou a frase inteira, recuperará também documentos que contenham os termos "pequenas empresas" e "trabalhadores autônomos". Isso porque durante o processo de indexação estabeleceu-se uma relação entre os termos "pequenas empresas" e "trabalhadores autônomos". A cada novo documento que é indexado, o sistema melhora progressivamente seu "conhecimento" sobre os termos de indexação e suas relações.

7.2.4 Meta buscas

Um único mecanismo de busca não consegue cobrir todo o espaço informacional da Web. Diferentes mecanismos possuem diferentes algoritmos de coleta de URLs e variam no número de robôs que utilizam e a freqüência com que rastreiam a Web. Por esse motivo ocorre uma grande diferença no conjunto de URLs que cada mecanismo coleta e na maneira como extrai os termos que irão compor seus índices. Eles podem diferir também na forma como são processadas as buscas dos usuários e como são ordenados e apresentados os

resultados. Por esse motivo, para se realizar uma busca exaustiva de uma determinada informação é necessário a utilização de vários mecanismos para se garantir a cobertura de uma boa parte da Web. Este seria um processo extremamente trabalhoso.

Para resolver este problema, alguns mecanismos fazem suas buscas utilizando diversos outros mecanismos de busca. Nesses meta-buscadores, ou meta-mecanismos, o usuário define sua expressão de busca como em qualquer *site* de busca. Essa expressão de busca é traduzida e enviada para cada um dos mecanismos que o meta-buscador gerencia. As buscas são então executadas e cada mecanismo retornará uma lista ordenada de URLs. O meta-buscador agrega estas listas em uma única lista de URLs, eliminando possíveis duplicações e a exibe ao usuário.

Um exemplo de meta-buscador é o MetaCrawler (www.metacrawler.com). O MetaCrawler unifica em uma única interface diversos mecanismos de busca. O MetaCrawler possui sua própria interface e oferece aos usuários alguns recursos para elaborarem suas buscas. Se algum recurso disponível no MetaCrawler não está presente em algum dos mecanismos de busca que gerencia, o MetaCrawler pode alterar a busca para adequá-la aos recursos oferecidos pelo mecanismo. Caso isso não seja possível, simplesmente aquele mecanismo não será acionado para realizar aquela busca.

Os meta-mecanismos são programas menores que os mecanismos de busca, pois não precisam utilizar robôs e não mantêm um banco de dados de URLs. Todo o "trabalho pesado" fica a cargo dos mecanismos de busca. Uma tarefa específica dos meta-mecanismos é a eliminação de itens (URLs) repetidos e a reordenação dos resultados fornecidos por seus mecanismos. Como os meta-mecanismos são programas relativamente simples, algumas empresas agregam a eles alguns recursos adicionais para melhorar seu desempenho, como, por exemplo, a possibilidade de definição de filtros personalizados que eliminam automaticamente determinados itens não desejados ou URLs que endereçam páginas que não mais existem.

Ao utilizarmos um *site* de busca percebe-se que, mesmo com os diversos recursos oferecidos, na maioria das vezes a precisão dos resultados fica longe do ideal. Apesar do grande número de itens encontrados, a maior parte dos mesmos não se enquadra perfeitamente à necessidade de informação. Alguns nem mesmo dizem respeito ao assunto procurado. Com sorte são encontradas referências que se aproximam do que realmente se procura, após uma verificação de cada item recuperado. Um dos motivos dessa baixa precisão

está no fato de a maioria dos mecanismos de busca ignorar as marcações das páginas HTML, considerando apenas o seu texto. Alguns mecanismos de busca consideram tais marcações, possibilitando uma busca restrita, por exemplo, ao título ou autor da página. Porém, as *tags* da linguagem HTML estão relacionadas apenas com o aspecto visual da página e não à atribuição de significado à informação nela contida. Essa limitação da linguagem HTML reflete diretamente na qualidade da informação recuperada, e motivou a criação da linguagem XML, que vem se tornando o novo padrão de páginas da Web.

7.3 A linguagem XML

A grande aceitação da linguagem HTML fez com que ela se tornasse o padrão para a construção de páginas da Web. Porém, com o passar do tempo e apesar de constantes atualizações, surgiram novas exigências de mercado não atendidas pelas características da linguagem HTML. Visando resolver as limitações da HTML, em 1996 especialistas se uniram para a definição de um novo padrão de linguagem de marcação. A principal característica dessa nova linguagem deveria ser a possibilidade de se definir um número ilimitado de *tags*. Um desenvolvedor de páginas Web poderia definir suas próprias *tags* quando necessário, em vez de ficar restrito ao esquema de marcação da HTML. Essa nova linguagem é conhecida com a sigla XML (e**X***tensible* **M***arkup* **L***anguage*).

HTML

```
<html>
  <body>
  Micromputador Pentium 4, 1.5 GHz,
256MB de RAM, Monitor 17 polegadas,
mouse, teclado, estabilizador.
  </body>
</html>
```

XML

```
<microcomputador>
   <modelo>Pentium 4</modelo>
   <velocidade>1.5 GHz</velocidade>
   <ram>256Mb de memória</ram>
   <monitor>17 polegadas</monitor>
   <teclado>Sim</teclaco>
   <mouse>Sim</mouse>
   <estabilizador>Sim</estabilizador>
   <impressora>Não</impressora>
</microcomputador>
```

Figura 31 Comparação entre as linguagens HTML e XML

A Figura 31 apresenta uma definição de uma página HTML e uma página XML. Apesar da finalidade das duas páginas (HTML e XML) ser a de apresentar as características de um microcomputador, a linguagem XML possibilita discriminar cada uma das características e apresentar o dado relacionado à característica. Se, por exemplo, a página

XML fosse de um *site* de uma loja de computadores permitiria a seus consumidores obterem uma busca mais refinada do microcomputador que desejasse adquirir.

Em uma fase anterior à criação de um documento XML, geralmente define-se a estrutura ou uma sintaxe desse documento através de um *esquema*. A especificação de um esquema, embora opcional, é importante para manter a consistência do documento XML, permitindo verificar sua validade frente ao esquema previamente definido. Existem dois principais tipos de esquemas: DTD e XML *Schema*.

A DTD (*Document Type Definition*) é um arquivo do tipo texto onde estão definidas as *tags*, a ordem em que elas devem aparecer no documento XML e sua obrigatoriedade. Essas definições são feitas com a utilização de uma meta-linguagem cuja sintaxe difere significativamente da sintaxe XML, como pode ser visto na Figura 32. Na maioria das vezes dois documentos, XML e DTD, trabalham em conjunto em uma página da Web. Com a ajuda da DTD, o *browser* consegue verificar todos os detalhes do documento XML e informar alguma inconsistência.

DTD (arquivo: "livro.dtd")

```
<!ELEMENT livro (titulo,genero?,autor+,editora)>
<!ELEMENT titulo (#PCDATA)>
<!ELEMENT genero (#PCDATA)>
<!ELEMENT autor (nome, dtnasc)>
<!ELEMENT nome (#PCDATA)>
<!ELEMENT dtnasc (#PCDATA)>
<!ELEMENT editora (#PCDATA)>
```

XML

```
<!DOCTYPE livro SYSTEM "livro.dtd">
<livro>
  <titulo>A Rosa do Povo</titulo>
  <genero>poesia</genero>
  <autor>
    <nome>Carlos Drummond de Andrade</nome>
    <dtnasc>1902-10-31</dtnasc>
  </autor>
  <editora>José Olympio</editora>
</livro>
```

Figura 32 Exemplo de utilização de uma DTD em um documento XML

Na DTD da Figura 32, armazenada em arquivo de nome "livro.dtd", é definido um elemento principal "livro". A especificação de um "livro" é feita através de seu título, gênero, autores e editora. A interrogação (?) após a palavra "genero" indica que a especificação do

gênero do livro será opcional. O sinal de mais (+) após a palavra “autor” indica que um livro pode ter um ou mais autores.

No documento XML é feito inicialmente o vínculo com o arquivo “livro.dtd” através da declaração **!DOCTYPE.** No arquivo “livro.dtd” está a definição da estrutura do documento XML com o qual este documento será validado.

Uma outra linguagem para a especificação de esquemas é a ***XML Schema***. A linguagem *XML Schema*, apesar de ter a mesma função da DTD, possui muitas características que a torna mais poderosa (e mais complexa) do que a DTD. Com a *XML Schema* é possível não apenas especificar a sintaxe de um documento XML, mas também especificar os tipos de dados de cada elemento desse documento. É possível também reutilizar a definição de elementos de outros esquemas, criar tipos de dados personalizados, especificar o número mínimo e máximo de vezes que um elemento pode ocorrer, criar listas e grupo de atributos (Furgeri, 2001). De fato, as definições feitas em *XML Schema* são elas próprias documentos XML. Desta forma, aplicações desenvolvidas para XML podem também ser aplicadas às definições de esquemas da linguagem *XML Schema*.

DTD

```
<!ELEMENT livro (titulo,genero?,autor+,editora)>
<!ELEMENT titulo (#PCDATA)>
<!ELEMENT genero (#PCDATA)>
<!ELEMENT autor (nome, dtnasc)>
<!ELEMENT editora (#PCDATA)>
<!ELEMENT nome (#PCDATA)>
<!ELEMENT dtnasc (#PCDATA)>
```

XML Schema

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="livro">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="titulo"  type="xs:string"/>
        <xs:element name="genero"  type="xs:string"/>
        <xs:element name="autor"   type="TAutor" minOccurs="1"/>
        <xs:element name="editora" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:complexType name="TAutor">
    <xs:sequence>
      <xs:element name="nome"    type="xs:string"/>
      <xs:element name="dtnasc"  type="xs:date"/>
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

Figura 33 Comparação entre DTD e XML *Schema*

Na Figura 33 é apresentada uma comparação entre a DTD e a *XML Schema*. A *XML Schema* possui a mesma sintaxe da linguagem XML, apresenta explicitamente a hierarquia dos elementos do documento e permite definir o tipo desses elementos. Existem alguns tipos pré-definidos como *string*, *date*, *float*, etc., mas pode-se definir tipos complexos através do agrupamento de vários elementos. Na Figura 33 foi definido o tipo "TAutor" que é composto pelos elementos "nome" (do tipo *string*) e "dtnasc" (do tipo *date*). Um livro, como especificado na *XML Schema*, deve ter pelo menos um (1) "autor". Esta restrição é definida pela declaração "**minOccurs**".

Na primeira linha de um documento *XML Schema* é definido um endereço Web onde estão definidos os elementos da sintaxe da própria linguagem *XML Schema*: *schema*, *element*, *sequence*, *complexType*, *string*, etc. Este endereço é conhecido como ***namespace***, e pode ser identificado pela expressão **xmlns**. O uso de *namespaces* aumenta a flexibilidade da linguagem *XML Schema* permitindo a reutilização de definições feitas em outros esquemas.

XML Schema (http://sites.uol.com.br/ferneda/livro.xsd)

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="livro">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="titulo"  type="xs:string"/>
        <xs:element name="genero"  type="xs:string"/>
        <xs:element name="autor"   type="TAutor" minOccurs="1"/>
        <xs:element name="editora" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:complexType name="TAutor">
    <xs:sequence>
      <xs:element name="nome"    type="xs:string"/>
      <xs:element name="dtnasc"  type="xs:date"/>
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

XML

```
<livro xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://sites.uol.com.br/ferneda/livro.xsd">
  <titulo>A Rosa do Povo</titulo>
  <genero>poesia</genero>
  <autor>
    <nome>Carlos Drummond de Andrade</nome>
    <dtnasc>1902-10-31</dtnasc>
  </autor>
  <editora>Jose Olympio</editora>
</livro>
```

Figura 34 Exemplo de utilização de um *XML Schema* em um documento XML

A primeira linha do documento XML especifica o *namespace* e o esquema (*XML Schema*) que será utilizada para validar o documento. No exemplo da Figura 34, o documento XML referencia o arquivo com endereço **http://sites.uol.com.br/ferneda/livro.xsd**.

A linguagem XML está se tornando padrão na criação de páginas Web e, graças à sua flexibilidade, ela é a base para a criação de diversas outras linguagens. Ao final do ano 2000 existiam cerca de 500 linguagens de uso especial baseadas na XML e, como será visto a seguir, esta linguagem desempenha um papel fundamental na construção da Web Semântica (Daum e Merten, 2002).

É possível traçar um paralelo entre a linguagem XML e a norma ISO 2709. A ISO 2709 (*Document Format for bibliographic interchange on magnetic tape*), criada em 1973, estabelece o conceito de *registro*, *campos*, características associadas aos campos (campo

repetitivo, campo numérico, etc.), ordem dos campos e *tags* para identificação dos campos, de forma semelhante à linguagem XML.

A ISO 2709 é um formato de transmissão de dados projetado para ser utilizado por aplicações de um determinado domínio. Esta norma foi criada em um contexto particular, no qual os softwares de bibliotecas precisavam trocar dados através de arquivos seqüenciais, geralmente fitas magnéticas. Da mesma forma, a linguagem XML está sendo ajustada para o mesmo objetivo, em um ambiente extremamente complexo que caracteriza a sociedade contemporânea. Particularmente, o formato MARC (junção da ISO 2709 com um conjunto de elementos de metadados) vem sendo muito utilizado como padrão em especificações XML.

7.4 Web Semântica

Com o objetivo de melhorar a recuperação de informação em grandes repositórios como a Web, pesquisas atualmente em curso estão buscando encontrar formas de possibilitar a agregação de um maior nível semântico às páginas Web. Procura-se aumentar a eficiência dos mecanismos de busca e de outros tipos de ferramentas de processamento automático de documentos através da utilização de linguagens que permitam definir dados e regras para o raciocínio sobre esses dados. Este grande desafio é a proposta da Web Semântica (Daconta, Obrst, e Smith, 2003).

Para a realização da Web Semântica são necessárias linguagens que permitam não apenas a definição de dados através de marcações, mas que possibilitem também descrever formalmente estruturas conceituais que possam ser utilizadas pelos agentes (robôs) de indexação dos mecanismos de busca.

O *World Wide Web Consortium* (W3C), através de Tim Berners-Lee, definiu uma estrutura em camadas que reflete os passos que devem ser dados para que o projeto da Web Semântica seja realizado de uma forma incremental (Figura .35).

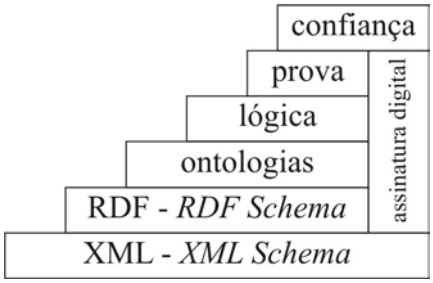


Figura 35 Arquitetura da Web Semântica

A primeira camada refere-se ao conjunto de páginas Web que utilizam a linguagem XML e suas respectivas definições estruturais feitas através da linguagem *XML Schema*. Como visto anteriormente, a linguagem XML permite definir documentos Web com marcações personalizadas, garantindo um maior nível semântico em relação às páginas HTML. A linguagem *XML Schema* permite formalizar a estrutura de páginas XML e valida-las, garantindo que estejam corretamente definidas. Estas duas linguagens (XML e *XML Schema*) já estão consolidadas e o número de documentos cresce rapidamente.

Apesar de a camada **XML – *XML Schema*** constituir um sólido alicerce, ela não faz parte da Web Semântica. A definição da Web Semântica inicia-se de fato com a camada **RDF-*RDF Schema*** e a cada nova camada aumenta-se o nível de abstração de seus componentes.

Os próximos tópicos serão abordados seguindo (de baixo para cima) cada camada da estrutura da Web Semântica apresentada na Figura 35, iniciando-se pela camada **RDF-*RDF Schema***.

7.4.1 A camada RDF-*RDF Schema*

A semântica da linguagem XML é um subproduto da definição da estrutura de um documento. Portanto, a estrutura e a semântica se confundem no interior de um documento XML. A linguagem denominada *Resource Description Framework* (RDF) fornece um meio de agregar semântica a um documento sem se referir à sua estrutura. A RDF visa oferecer uma forma eficiente de descrever metadados na Web, possibilitando a interoperabilidade entre aplicações que compartilham metadados.

A RDF está baseada em três tipos de objetos: *recurso* ("*resource*"), *propriedade* ("*property*") e *declaração* ("*statement*"). Um **recurso** é qualquer objeto da Web que possui

um endereço, como, por exemplo, uma página HTML ou XML identificada por uma URL. Uma **propriedade** é uma característica, um atributo ou uma relação usada para descrever um recurso. Um recurso, juntamente com uma propriedade e seu valor é denominado **declaração**. Essas três partes de uma declaração são chamadas respectivamente de sujeito ("*subject*"), predicado ("*predicate*") e objeto ("*object*").

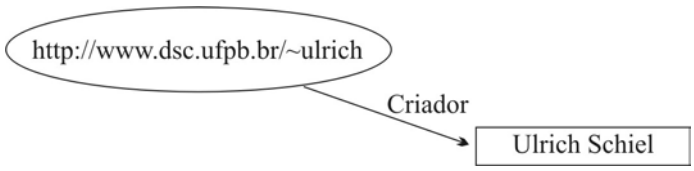
Para exemplificar, vamos considerar a seguinte sentença:

"*Ulrich Schiel é o criador do recurso http://www.dsc.ufpb.br/~ulrich*"

Conforme a definição da linguagem RDF, essa sentença (declaração) é dividida nas seguintes partes:

Sujeito (recurso)	http://www.dsc.ufpb.br/~ulrich
Predicado (propriedade)	Criador
Objeto	Ulrich Schiel

A sentença utilizada no exemplo poderia ser representada na forma de um grafo:



Utilizando a linguagem RDF a sentença seria representada como:

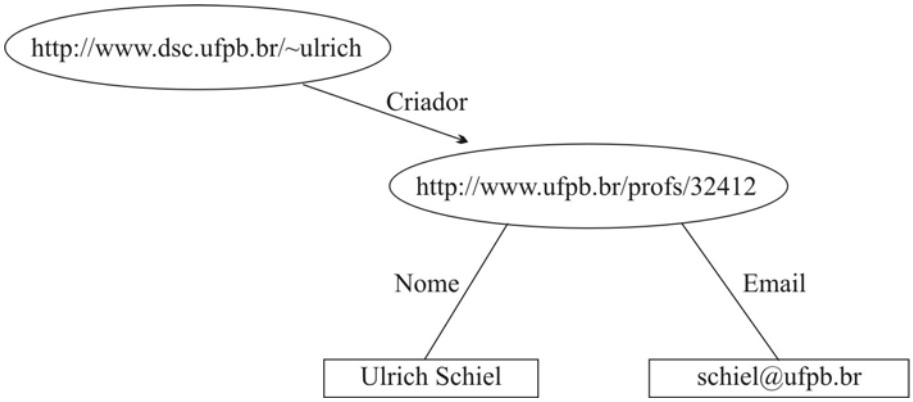
```
<rdf:RDF xmlns="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:s="http://description.org/schema/">
  <rdf:Description about="http://www.dsc.ufpb.br/~ulrich">
    <s:Creator>Ulrich Schiel</s:Creator>
  </rdf:Description>
</rdf:RDF>
```

As primeiras linhas de um documento RDF especificam endereços (*namespaces*) onde são encontradas a descrição da sintaxe da linguagem RDF e a descrição do esquema utilizado no documento.

Supondo que se deseja apresentar algumas características do criador de um recurso, exemplificado pela seguinte sentença:

"*O recurso http://www.dsc.ufpb.br/~ulrich foi criado pelo professor de número 32412 chamado Ulrich Schiel e cujo endereço de e-mail é schiel@ufpb.br*"

O modelo RDF para esta sentença pode ser representado pelo seguinte grafo:



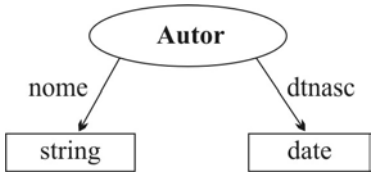
O código RDF para esta sentença é:

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:s="http://description.org/schema/">
  <rdf:Description about="http://www.dsc.ufpb.br/~ulrich">
    <s:Creator>
       <rdf:Description about="http://www.ufpb.br/profs/32412">
         <v:Name>Ulrich Schiel</v:Creator>
         <v:Email>schiel@ufpb.br</v:Email>
       </rdf:Description>
    </s:Creator>
  </rdf:Description>
</rdf:RDF>
```

A linguagem RDF define um modelo para descrever relacionamentos entre recursos através de suas propriedades e valores. Porém, a RDF não fornece mecanismos para declarar essas entidades nem para definir tais relacionamentos. Para esse objetivo foi desenvolvida a linguagem *RDF Schema*.

A *RDF Schema* é uma linguagem capaz de definir um sistema de classes extensível e genérico que pode ser utilizado como base para a descrição conceitual de um domínio específico.

Para exemplificar, serão definidas a seguir as classes **Autor**, **Publicação** e **Livro**. A classe **Livro** pode ser definida como uma subclasse de **Publicacao**. Esse relacionamento entre classes é especificado através da propriedade *subClassOf*. A classe **Autor** possui duas propriedades: *nome* e *dtnasc*. A propriedade *nome* é definida como sendo do tipo *string*, podendo receber qualquer cadeia de caracteres. A propriedades *dtnasc* é do tipo *date* e deve conter apenas datas válidas. A Figura 36 apresenta uma representação gráfica da classe **Autor**, seguida de sua definição na linguagem *RDF Schema*.




```
<rdfs:Class rdf:ID="Autor">
    rdfs:subClassOf rdf:resource="http://www.w3.org/2000/01/rdf-schema#"/>
</rdfs:Class>

<rdf:Property rdf:ID="nome">
  <rdfs:domain rdf:resource="#Autor"/>
  <rdfs:range rdf:resource="http://www.w3.org/TR/xmlschema-2/#string"/>
</rdf:Property>

<rdf:Property rdf:ID="dtnasc">
  <rdfs:domain rdf:resource="#Autor"/>
  <rdfs:range rdf:resource="http://www.w3.org/TR/xmlschema-2/#date"/>
</rdf:Property>
```

Figura 36 Definição *RDF Schema* da classe Autor

Toda classe deve ser necessariamente derivada de uma classe hierarquicamente superior. A classe **Autor** é derivada da classe de mais alto nível definida pelo recurso **`http://www.w3.org/2000/01/rdf-schema#`**.

A Figura 37 apresenta a definição da classe **Publicação**, que possui duas propriedades: *título* e *gênero*, ambas do tipo *string*.


```
<rdfs:Class rdf:ID="Publicacao">
  <rdfs:subClassOf resource="http://www.w3.org/2000/01/rdf-schema#"/>
</rdfs:Class>

<rdf:Property rdf:ID="titulo">
  <rdfs:domain rdf:resource="#Publicacao"/>
  <rdfs:range rdf:resource="http://www.w3.org/TR/xmlschema-2/#string"/>
</rdf:Property>

<rdf:Property rdf:ID="genero">
  <rdfs:domain rdf:resource="#Publicacao"/>
  <rdfs:range rdf:resource="http://www.w3.org/TR/xmlschema-2/#string"/>
</rdf:Property>
```

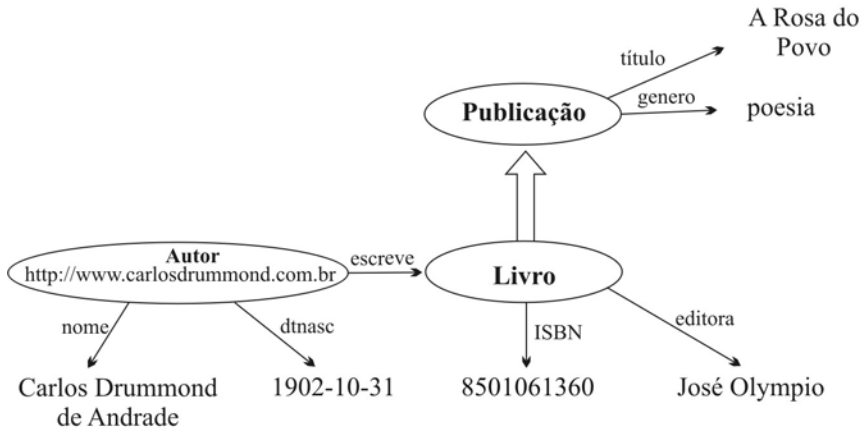
Figura 37 Definição *RDF Schema* da classe Publicação

A classe **Livro** é uma subclasse da classe **Publicação** e, além das propriedades herdadas desta, possui duas propriedades específicas: *ISBN* e *editora*. Existe também uma relação entre **Livro** e **Autor** representada pela propriedade *escreve*. Na Figura 38 é apresentado um diagrama da classe **Livro** e a sua codificação em *RDF Schema*.


```
<rdfs:Class rdf:ID="Livro">
  <rdfs:subClassOf rdf:resource="#Publicacao"/>
</rdfs:Class>

<rdf:Property rdf:ID="ISBN">
  <rdfs:domain rdf:resource="#Livro"/>
  <rdfs:range rdf:resource="http://www.w3.org/TR/xmlschema-2/#integer"/>
</rdf:Property>

<rdf:Property rdf:ID="editora">
  <rdfs:domain rdf:resource="#Livro"/>
  <rdfs:range rdf:resource="http://www.w3.org/TR/xmlschema-2/#string"/>
</rdf:Property>

<rdf:Property rdf:ID="escreve">
  <rdfs:domain rdf:resource="#Autor"/>
  <rdfs:domain rdf:resource="#Livro"/>
</rdf:Property>
```

Figura 38 Definição *RDF Schema* da classe Livro

Definida a estrutura de classes, podem-se associar a ela recursos (*resources*) na forma de instâncias de uma ou mais classes. A Figura 39 apresenta um exemplo simplificado de um documento RDF no qual é definida uma instância da classe Autor.




```
<Autor rdf:about="http://www.carlosdrummond.com.br">
  <nome>Carlos Drummond de Andrade</nome>
  <dtnasc>1902-10-31</dtnasc>
  <escreve>
    <Livro>
      <titulo>A Rosa do Povo</titulo>
      <genero>Poesia</genero>
      <editora>Jose Olympio</editora>
      <ISBN>8501061360</ISBN>
    </Livro>
  </escreve>
</Autor>
```

Figura 39 Documento RDF definido a partir de um *RDF Schema*

Apesar de haver muitos esforços concentrados na evolução da linguagem RDF, há ainda muito por se fazer para que ela esteja consolidada. A linguagem RDF ainda é muito pouco conhecida, até porque é muito nova, mas espera-se que, assim com a linguagem XML, ela se fortaleça para que o projeto da Web Semântica se realize.

7.4.2 A camada de Ontologias

A camada de ontologias aproveita a extensibilidade da linguagem *RDF Schema* para definir estruturas que se assemelham aos *frames*, como visto no Capítulo 5.

Na maioria das vezes uma ontologia toma a forma de uma árvore hierárquica de classes, de maneira que cada classe herda as características de uma ou mais classes superiores. Cada classe representa um conceito do domínio que está sendo modelado, e seu significado é expresso pelas suas propriedades, similaridades e diferenças em relação aos outros conceitos. No contexto da Ciência da Informação este recurso é utilizado em larga medida, denominado "plano de classificação" ou "tesauro".

Os relacionamentos entre conceitos devem ser definidos de maneira clara e sem ambigüidade para um correto processamento por sistemas computacionais. Além disso, é importante que os usuários possam visualizar e entender uma ontologia. Por isso algumas abordagens suportam a modelagem de ontologias em várias camadas, onde a camada superior corresponde ao que um ser humano consegue entender facilmente. Desta maneira o usuário poderá percorrer a ontologia a fim de modificá-la ou consultá-la. Já a camada inferior deve ser definida mais formalmente para que possa ser compreendida pelo computador. As camadas intermediárias se constituem de mapeamentos entre as camadas superiores, menos formais, e as camadas inferiores, mais formais.

Além do significado dos conceitos e suas relações, uma ontologia pode conter também *axiomas* que definem regras sobre os relacionamentos entre os conceitos. Por exemplo, um axioma pode definir se um relacionamento entre dois conceitos é simétrico ou não.

Algumas abordagens não só fornecem meios para a modelagem e armazenamento de ontologias, mas também tentam automatizar pelo menos parcialmente este processo através da utilização de ferramentas de *aprendizado automatizado de conceitos*. Geralmente essas ferramentas analisam páginas Web de *sites* relacionados ao domínio da aplicação a fim de extrair uma terminologia do domínio. Posteriormente, as informações obtidas são filtradas e os relacionamentos são apreendidos (Maedche e Staab, 2000).

Existem algumas linguagens específicas para a modelagem de ontologias. Uma delas é a linguagem OIL. OIL (*Ontology Inference Layer*) é uma linguagem criada para representar a semântica de determinados domínios através da definição de uma estrutura acessível por computadores. Desenvolvida para ser compatível com as linguagens XML e RDF, OIL explora a estrutura de modelagem da *RDF Schema*. Desta maneira, aplicações que utilizam apenas RDF podem entender pelo menos parcialmente um documento OIL.

Uma ontologia definida na linguagem OIL consiste de uma lista de definições de classes (*class-def*) e atributos (*slot*), como exemplificado na Figura 40.

```
slot-def come
   inverse é-comido-por

slot-def tem-parte
   inverse é-parte-de
   properties transitive

class-def animal

class-def planta
   subclass-of NOT animal

class-def árvore
   subclass-of planta

class-def galho
   slot-constraint é-parte-de
            has-value árvore

class-def folha
   slot-constraint é-parte-de
            has-value galho

class-def defined carnívoro
   subclass-of animal
   slot-constraint come value-type animal

class-def defined herbívoro
   subclass-of animal
   slot-constraint come
            value-type planta OR
            (slot-constraint é-parte-de has-value planta)

class-def herbívoro
  subclass-of NOT carnívoro

class-def girafa
   subclass-of animal
   slot-constraint come value-type folha

class-def leão
   subclass-of animal
   slot-constraint come value-type herbívoro
```

Figura 40 Exemplo de ontologia utilizando a linguagem OIL

As pesquisas atuais na Web Semântica têm como principal enfoque as ontologias. Essa tendência é evidenciada pelo desenvolvimento de uma variedade de sistemas e arquiteturas visando prover a integração de ontologias, a criação de linguagens, bibliotecas e editores de ontologias.

As ontologias, ao ordenarem os termos, incorporam à Web a preocupação com a organização da informação e, conseqüentemente, de atribuição de significado aos mesmos. A inclusão de termos numa estrutura, qualquer que seja esta estrutura, veicula uma informação a

mais sobre os mesmos, informação esta fornecida pela localização relativa do termo na estrutura.

Os tesauros utilizados para representar a informação em Ciência da Informação têm o mesmo propósito que as ontologias, mas incorporaram ao longo do tempo a noção de *ponto de vista*. Em oposição aos sistemas de classificação universal, os tesauros organizam conceitos de áreas delimitadas do saber de acordo com objetivos pré-determinados. Estes objetivos nortearão o procedimento de categorização dos termos, pois este procedimento não é absoluto, objetivo ou universal.

As primeiras camadas da Web Semântica dispõem, desta maneira, padrões de registro dos documentos e especificam seu preenchimento com termos inseridos numa estrutura significante. A Web Semântica retoma procedimentos adotados pela Ciência da Informação desde a década de 70, qual seja: a definição de formatos de intercâmbio de registros bibliográficos e o desenvolvimento de tesauros.

Os três mais altos níveis da estrutura da Web Semântica apresentada na Figura 35 (lógica, prova e confiança) ainda não estão bem desenvolvidos. Assim, existem apenas conceitos gerais que norteiam o futuro desenvolvimento dessas três camadas.

7.4.3 As camadas Lógica, Prova e Confiança

A camada *lógica* é composta por um conjunto de regras de inferência que os agentes (computacionais ou humanos) poderão utilizar para relacionar e processar informação. As regras de inferência fornecem aos agentes computacionais o poder de raciocinar sobre as estruturas de dados definidas nas camadas mais baixas (XML e RDF), utilizando as relações entre esses objetos definidas na camada de ontologia.

Por exemplo, imaginando que uma revendedora de veículos define que quem vender mais do que 20 produtos em um ano será categorizado como *Super Vendedor*. Um programa pode seguir essa regra e fazer uma simples dedução: "José vendeu 25 veículos, portanto José é um *Super Vendedor*".

Uma vez que se constrói um sistema que segue a lógica definida, podem-se seguir as ligações semânticas para construir a *prova*. Pessoas podem escrever diversas definições lógicas. Por exemplo, os registros da empresa mostram que Maria vendeu 15 automóveis e 8 caminhões. O sistema define que automóveis e caminhões são produtos da empresa. As regras

matemáticas dizem que 15 + 8 = 23, que é maior que 20. Existe uma regra que diz que quem vende mais de 20 produtos é classificado como *Super Vendedor*. O computador junta as regras para provar que Maria é uma S*uper Vendedora*.

Na Web qualquer um pode dizer qualquer coisa sobre qualquer coisa. A *assinatura digital* é imprescindível para garantir a confiabilidade das informações. A autenticidade e confiabilidade das fontes adquirem um novo significado quando consideramos que agentes raciocinando sobre os dados podem chegar a conclusões que afetem a ação humana. As *assinaturas digitais* serão a forma de cada agente verificar a autenticidade das suas fontes. De acordo com a informação que a assinatura digital lhe fornecer, o agente poderá alterar o grau de certeza associado ao resultado do seu raciocínio ou mesmo ignorar a informação.

Ironicamente, a Web Semântica resgata os fundamentos da Diplomática, disciplina "*ligada à questão da falsificação e das dúvidas sobre a autenticidade de documentos medievais*" (Bellotto, 2002, p.15). Segundo a autora, a Diplomática nasceu quando jesuítas franceses, em 1643, resolveram publicar uma história dos santos, movidos pela intenção de separar a realidade das lendas. Na introdução à obra um dos jesuítas declarou ser falso um diploma assinado pelo rei Dagoberto I, o que invalidava vários diplomas medievais e que tinham sido preservados e tratados como completamente autênticos pelos beneditinos da Abadia de Saint Denis. Os beneditinos iniciam então uma guerra diplomática para responder à desconfiança provocada pelos jesuítas. Em 1681 o beneditino Jean de Mabillon publica uma obra em 6 volumes intitulada "De re diplomatica libri Sex" na qual estabelecia os procedimentos para garantir autenticidade, análise e compreensão dos atos escritos.

Vislumbra-se, neste aspecto, a necessidade de desenvolvimento de uma "diplomática da Web", cuja discussão já foi iniciada a partir dos debates sobre a assinatura eletrônica e o valor do documento digital em transações financeiras e procedimentos jurídico.

7.5 Conclusão

Pensada inicialmente para ser um hipertexto de dimensões mundiais, a estrutura da Web está fundamentada na apresentação de textos. Imagens e sons, quando presentes, estão na maior parte das vezes apoiadas sobre um suporte textual.

Segundo Barros (1999, p.7) um texto pode ser definido de duas formas complementares. Uma primeira concepção de texto toma-o como objeto de comunicação, que

se estabelece entre um destinador e um destinatário, e uma segunda definição faz dele um objeto de significação. Na história da Web observa-se inicialmente uma ênfase no caráter comunicativo de seu conteúdo. Progressivamente o foco da atenção recai sobre a significação. Essa mudança é notada claramente pelo surgimento sucessivo das linguagens de marcação. Da HTML à Web Semântica, novos recursos estão sendo implementados, sempre visando um maior nível semântico para os documentos da Web.

A atual predominância da linguagem HTML como estrutura informacional da Web é uma característica que afeta diretamente o processo de recuperação de informação. De fato, verifica-se que os mecanismos de recuperação na Web, apesar de se diferenciarem em muitos aspectos, não se distinguem muito quanto à qualidade de seus resultados. Para a solução de alguns desses problemas a linguagem XML desponta como um novo padrão para a criação das páginas Web.

A linguagem XML é sem dúvida um avanço em relação ao HTML no que se refere à descrição dos documentos Web. Porém, é difícil crer que a rigidez imposta por esta linguagem possa se adequar a toda a variedade de documentos existentes na rede. A linguagem XML é a base para criação de outras linguagens e forma a estrutura de suporte para a Web Semântica.

A Web Semântica ainda está dando os seus primeiros passos, sendo difícil prever seu futuro. A sua complexidade é ainda um grande empecilho, mas isso poderá ser contornado com a sua consolidação e a criação de ferramentas que facilitem sua utilização.

A Web é um enorme campo de prova para diversas teorias relacionadas ao tratamento e recuperação da informação. Desde o seu nascimento poucas mudanças ocorreram em sua estrutura básica. Talvez a Web Semântica seja a mudança necessária para que a Web se torne realmente uma fonte de informação confiável.

8
Conclusão

A hipótese que norteou este trabalho versa sobre a incapacidade de as técnicas computacionais fornecerem soluções absolutas e completas, mesmo em aspectos da ciência da informação em que o computador se apresenta de forma mais acentuada.

A partir dessa conjectura, e centrando-se na recuperação de informação, foram analisados os recursos oriundos da Ciência da Computação utilizados no processo de recuperação de informação.

Freqüentemente o computador é referenciado como o mais recente artefato utilizado para a mecanização do cálculo matemático. De fato, por volta de 1950 a utilização dos computadores estava quase que totalmente restrita à solução de cálculos matemáticos complexos. Com a "explosão da informação" e a urgência no tratamento da crescente produção de informação, o computador foi (e ainda parece ser) a solução mais direta para a época. Porém, deve-se sempre considerar que a utilização de recursos computacionais no tratamento da informação parte de reduções ou simplificações do conceito de informação que na maioria das vezes mostram-se insuficientes para os objetivos da Ciência da Informação, mesmo quando restrito ao processo de recuperação de informação.

A natural vocação dos computadores pelo processamento matemático justifica a predominância dos modelos quantitativos de recuperação de informação. Muitas teorias matemáticas foram trazidas para o interior da Ciência da Informação, formando um conjunto bastante diversificado de soluções para o tratamento da informação. Porém, os modelos

quantitativos impõem uma lógica na qual a informação deve ser numericamente definida no interior de um sistema fechado, desconsiderando alguns importantes fatores envolvidos no processo de recuperação de informação.

O ato de interpretar uma informação, de forma individual ou coletiva, é dependente da existência de um sujeito. Os modelos quantitativos desconsideram a presença de tal sujeito, não permitindo sua participação efetiva na adequação da representação dos documentos do sistema. Os modelos dinâmicos rompem a rigidez imposta pelos modelos quantitativos através da participação ativa do conjunto de usuários de um sistema de informação na representação dos documentos.

No âmbito da Ciência da Informação, as idéias inerentes aos modelos dinâmicos oferecem uma visão diferenciada do processo de recuperação de informação e abrem um campo de discussão sobre sua aplicabilidade em circunstâncias reais.

Os elementos envolvidos no processo de recuperação de informação são tipicamente lingüísticos; geralmente objetos textuais. Uma interpretação correta desses elementos refletirá positivamente na qualidade dos resultados de um sistema de recuperação de informação.

Aplicado aos sistemas de recuperação de informação, o Processamento da Linguagem Natural (PLN) visa resolver alguns fenômenos lingüísticos que dificultam uma interpretação correta das informações contidas nos documentos, como visto no Capítulo 6. Através do PLN a Ciência da Informação se aproxima da Inteligência Artificial e herda desta uma imensa bagagem teórica e prática.

A história da Ciência da Computação é caracterizada por uma sucessão de inventos que, de forma imprevisível, podem se perpetuar ou desaparecer. O futuro de um novo dispositivo ou uma nova tecnologia está condicionado não apenas à sua qualidade, mas também a fatores sociais de difícil mensuração ou análise. A evolução dos recursos computacionais não pode ser vista como um caminhar pé ante pé em uma estrada de mão única. Muito se tateia, se experimenta e por vezes se retoma idéias esquecidas, se reinventa. A Internet, como a conhecemos hoje, é em grande parte fruto dessa imprevisibilidade e do empirismo que caracteriza principalmente as ciências duras.

A Internet, particularmente a Web, evidencia a dificuldade inata dos computadores no tratamento adequado da *informação*, na acepção dada ao termo pela Ciência da Informação. Os desenvolvimentos recentes da Web reconhecem essa inabilidade na medida em que

buscam a criação de novas linguagens que objetivam uma maior valoração semântica aos documentos da Web. É interessante observar que no projeto da Web Semântica estão inseridos conceitos e idéias que há muito tempo são utilizados pela Ciência da Informação no tratamento documental.

Os primeiros computadores eletrônicos pesavam várias toneladas e ocupavam toda uma sala. A programação era feita através da conexão direta de seus circuitos por meio de cabos. Nos anos 50 a programação era feita através da transmissão de instruções em código binário por meio de cartões e fitas perfuradas. Com o surgimento das linguagens de programação, o código binário ficou limitado ao núcleo do computador e a comunicação com o mundo externo era feita por uma nova camada de programa.

Atualmente os computadores são constituídos por um conjunto de dispositivos e camadas de programas que se comunicam umas com as outras, permitindo um enorme distanciamento do seu núcleo no qual os dados e o processamento algoritmo desses dados são representados por meio de zeros e uns. Porém, o núcleo binário de um computador perpassa todas as suas camadas de programas e limita sua capacidade de efetuar tarefas que os seres humanos fazem com relativa facilidade como, por exemplo, a tradução, a indexação, a elaboração de resumos e diversos outros processos relacionados ao tratamento da informação.

A aplicação de métodos oriundos da Ciência da Computação contribui com a Ciência da Informação na medida em que viabiliza a operação de grandes quantidades de dados de uma forma rápida e ágil. No entanto, estas características não necessariamente resultam em processos consistentes ou satisfatórios de recuperação da informação.

A informação, tomada no contexto da Ciência da Informação, está diretamente relacionada ao seu significado, o que implica procedimentos menos formais ou operacionais, baseados na capacidade e na habilidade de abstração, apreensão e representação da significação, contextualizando-a. Estes processos não prescindem de uma efetiva análise dos conceitos para posterior representação. Esta operação intelectual não pode ser realizada de forma absoluta por modelos computacionais, pois estes trabalham apenas com formas significantes.

Recuperar informação implica operar seletivamente um estoque de informação, o que envolve processos cognitivos que dificilmente podem ser formalizados através de um algoritmo. Mesmo que um modelo computacional de recuperação da informação tenha como

base algum tipo de vocabulário e organização lógica, a equiparação dos significados supostamente implícitos pelos significantes depende de uma análise intelectual.

Seria desejável que os avanços teóricos e metodológicos já realizados pelos processos documentários no âmbito da Ciência da Informação fossem avaliados conjuntamente com os avanços realizados pela Ciência da Computação e vice-versa, quando da realização de pesquisas ou desenvolvimento de projetos voltados à recuperação de informação.

A capacidade do computador em operar com modelos formais poderia ser associada aos procedimentos intelectuais humanos, trabalhando-se com o melhor de cada um para a obtenção de resultados mais satisfatórios e adequados. A utilização de modelos puramente computacionais poderia ser uma escolha consciente baseada na relação custo-benefício.

Pode-se concluir que os métodos e técnicas desenvolvidos pela Ciência da Computação devem ser continuamente avaliados e até absorvidos pela Ciência da Informação. Porém a Ciência da Informação não poderá ser desenvolvida no vazio cultural de um sistema de raciocínio algorítmico. Além disso, considerando as tarefas intelectuais do profissional da informação e tudo que se espera deles, é improvável que suas habilidades possam ser substituídas por qualquer tipo de tecnologia.

8.1 Sugestões para pesquisas futuras

Ao iniciar este trabalho, há quatro anos, me perguntava como a Ciência da Computação poderia contribuir para o avanço da Ciência da Informação, já que, para mim, muitos recursos computacionais estavam sendo ignorados. Hoje me questiono como a Ciência da Informação pode contribuir para o avanço da Ciência da Computação.

Durante a elaboração deste trabalho foram consultadas diversas dissertações e teses em Ciência da Computação que versam sobre o tratamento da informação textual. Muitas delas mostram desconhecer até mesmo a existência da Ciência da Informação, e apresentam como novos, métodos e técnicas há muito tempo utilizados por esta ciência. Por outro lado, quando se trata da utilização de métodos computacionais no tratamento da informação, observa-se na literatura da Ciência da Informação reações que vão desde o ceticismo até o otimismo exagerado, mostrando também desconhecimento sobre a Ciência da Computação.

Portanto, é desejável que futuras pesquisas venham a ser desenvolvidas de forma mais integrada, buscando trazer para a Ciência da Informação conhecimentos e idéias da Ciência da

Computação. Da mesma forma, as pesquisas em Ciência da Computação devem considerar a existência de uma ciência que há muito tempo vem abordando de forma sistemática os problemas relacionados ao tratamento e recuperação da informação.

O surgimento acelerado de novas tecnologias requer dos profissionais da informação uma pesquisa contínua, lançando sobre tais tecnologias um olhar crítico a fim de avaliar a sua adequação, especificamente no tratamento da informação ou à Ciência da Informação como um todo.

No contexto deste trabalho é possível destacar alguns assuntos que merecem aprofundamento em futuras pesquisas. É o caso dos modelos dinâmicos, que apresentam idéias que devem ser avaliadas de forma sistemática, pois rompem certos paradigmas da Ciência da Informação ao permitirem que a representação da informação no interior de um sistema seja alterada de acordo com sua demanda.

Desde o seu nascimento a Internet e a Web são estudados nas mais variadas áreas do conhecimento. Ao que tudo indica, a Web Semântica propiciará um campo fértil de pesquisa, principalmente para a Ciência da Informação, pois, como visto no Capítulo 7, a mesma incorpora conceitos criados no interior desta ciência e que estão sendo aplicados a um *corpus* de dimensões nunca imaginadas.

Bibliografia

ALLEN, J. (1995) **Natural language understanding**. Redwood City: The Benjamin/Cummings.

ANDREWS, K., KAPPE, F. e MAURER, H. (1995) Serving information to the Web with Hyper-G. **Computer Network and ISDN Systems**, v. 27, n. 6, p.919-926.

ARAMPATZIS, et al. (2000) Linguistically-motivated Information Retrieval. **Encyclopedia of Library and Information Science**, v.69, p.201-222.

BAEZA-YATES, R. e RIBEIRO-NETO, B. (1999) **Modern Information Retrieval**. Addison-Wesley.

BARRETO, A. (1994) A questão da informação. **São Paulo em Perspectiva**, v.8, n.4, p.3-8.

BARROS, D.L.P. (1999) **Teoria semiótica do texto**. São Paulo: Ática. (Série Fundamentos, n.72).

BEARDON, C., LUMSDEN, D. e HOLMES, G. (1991) **Natural language and computational linguistics**. Melksham-Wiltshire, England: Ellis Horwoood.

BEIN, J. e SMOLENSKY, P. (1988) **Application of the interactive activation model to document retrieval**. Technical Report CU-CS-405-88. University of Colorado at Boulder. Department of Computer Science.

BELEW, R. K. (1989) Adaptive information retrieval. **Proceedings of the 12th annual international ACM SIGIR conference on research and development in information retrieval**, p.11-20.

BELLEI, S.L.P. (2002) **O livro, a literatura e o computador**. São Paulo: EDUC.

BELLOTTO, H.L. (2002) **Como fazer análise diplomática e análise tipográfica de documento de arquivo**. São Paulo: Arquivo do Estado, Imprensa Oficial do Estado. (Projeto Como Fazer, n.8).

BLAIR, D.C. (1990) **Language and representation in information retrieval**. Amsterdam: Elsevier.

BORDOGNA, G. et al. (1990) A system architecture for multimedia information retrieval. **Journal of Information Science**. v. 16, n. 2, p.229-238.

BORDOGNA, G. e PASI, G. (1995) Controlling Information Retrieval through a user adaptive representation of documents. **International Journal of Approximate Reasoning**, 12, p.317-339.

BORGMAN, C.L. (2000) **From Gutenberg to the global information infrastructure**: access to information in the networked world. Cambridge: MIT Press.

BORKO, H. (1968) Information Science: What is it? **American Documentation**, v. 19, n. 1, p.3-5..

BOUGNOUX, D. (1994) **Introdução às ciências da informação e da comunicação**. Petrópolis: Vozes.

BRAGA, A.P., CARVALHO, A.C.P.L.F. e LUDEMIR, T.B. (2000) **Redes neurais artificiais**: teoria e aplicações. Rio de Janeiro: LTC.

BRAGA, G. M. (1995) Informação, ciência da informação: breves reflexões em três tempos. **Ciência da Informação**, v. 24, n. 1, p.84-88.

BRITO, A.N., VALE, O.A. (orgs) (1998) **Filosofia, lingüística, informática**: aspectos da linguagem. Goiânia: Universidade Federal de Goiás.

BRUANDET, M-F. (1987) Outline of a knowledge-base model for an intelligent information retrieval system. **Information Processing and Management**, v. 25, n. 1, p.89-115.

BUCKLAND, M.K. (1991a) **Information and Information Systems**. New York: Greenwood.

BUCKLAND, M.K. (1991b) Information as thing. **Journal of the American Society of Information Science**, v.42, n.5, p.351-360.

BUCKLAND, M.K. (1997) What is a "document"? **Journal of the American Society of Information Science**, v.48, n.9, p.804-809**.**

BUCKLEY, C. et al. (1995) Automatic query expansion using SMART: TREC 3. In: Harmon, D.K. (ed.) **Overview of the Third Text REtrieval Conference (TREC-3)**. NIST Special Publication 500-225, p.69-80.

BURKE, M.A. (1999) **Organization of multimedia resources**: principle and practice of information retrieval. Aldershot: Gower.

BUSH, V. (1945) As we may think. **The Atlantic Monthly**, v. 176, n. 1; pp 101-108. Disponível em <http://www.theatlantic.com/unbound/flashbks/computer/bushf.htm>. Acessado em 06.02.2003.

CASTELLS, M. (1999) **A sociedade em rede**. 2ª edição. São Paulo: Paz e Terra.

CHARTIER, Roger. (1999) **A aventura do livro**: do leitor ao navegador; conversações com Jean Lebrun. São Paulo: Fundação Editora da UNESP.

CHARTIER, Roger. (2002) **Os desafios da escrita**. São Paulo: Fundação Editora da UNESP.

CHIARAMELLA, Y. et al. (1986) IOTA: A Full Text Information Retrieval System. **Proceedings of the 9th annual international ACM SIGIR conference on research and development in information retrieval**, p.207-213

CHIARAMELLA, Y. e DEFUDE, B. (1987) A prototype of an intelligent system for information retrieval: IOTA. **Information Processing and Management**, v. 23, n. 4, p.285-303.

CINTRA, A.M.M. et al. (1994) **Para entender as linguagens documentárias**. São Paulo: Polis: APB. (Coleção Palavra Chave, 4)

CROFT, W.B., TURTLE, H.R., LEWIS, D.D. (1991) The use of phrases and structured queries in information retrieval. **Proceedings of the 14th annual international ACM SIGIR conference on research and development in information retrieval**, p.32-45.

DACONTA, M.C, OBRST, L.J. e SMITH, K.T. (2003) **The Semantic Web**: a guide to the future of XML, Web services, and knowledge management. Indianapolis: Wiley.

DAUM, B e MERTEN U. (2002) **Arquitetura de sistemas com XML**. Rio de Janeiro: Campus.

DEERWESTER, S.C. et al. (1990) Indexing by latent semantic analysis. **Journal of the American Society for Information Science**, v. 41, n. 6, p.391-407.

DENNING, P.J. et al. (1989) Computing as a discipline. **Communication of the ACM**, v. 32, N. 1, p.9-23.

DERTOUZOS, M. (1997) **O que será**: como o novo mundo da informação transformará nossas vidas. São Paulo: Companhia das Letras.

DEVLIN, K. (1991) **Logic and Information**. Cambridge: University Press.

DOSZKOCS T., REGGIA, J. e LIN, X. (1990) Connectionist models and information retrieval. **Annual Review of Information Science & Technology**, v. 25, p.209-260.

DREYFUS, H.L. (1999) **What computers still can't do**: a critique of artificial reason. Cambridge: MIT Press.

ELLIS, D. (1996) **Progress and Problems in Information Retrieval**. London: Library Association Publishing.

FALOUTSOS, C. e OARD, D. (1995) **A survey of information retrieval and filtering methods**. Techinical Report CS-TR-3514. Department of Computer Science, University of Maryland.

FERNEDA, E. (1997) **Construção automática de um thesaurus retangular**. Campina Grande. Dissertação (Mestrado em Informática), Universidade Federal da Paraíba.

FERREIRA, S.M.S.P. (1995) Novos paradigmas e novos usuários de informação. **Ciência da Informação**. v.25, n.2. Versão eletrônica.

FIGUEIREDO, N.M. (1999) **Paradigmas modernos da Ciência da Informação**. São Paulo: Polis. (Coleção Palavra-Chave, 10).

FONSECA FILHO, C. (1999) **História da computação – teoria e tecnologia**. São Paulo: LTr.

FORD, N. (1991) **Expert systems and artificial intelligence**: an information manager´s guide. London: Library Association Publishing.

FURGERI, S. (2001) **Ensino didático da linguagem XML**. São Paulo: Érica.

GAUCH, S. e FUTRELLE, R.P. (1994) Experiments in automatic word class and word sense identification for information retrieval. **Proceedings of 3rd Annual symposium on document analysis and information retrieval**, p.425-434.

GORDON, M. (1988) Probabilistic and genetic algorithms for document retrieval. **Communications of the ACM**, v. 31, n. 10, p.1208-1218.

HAUPTMANN, A.G. et al. (1998) Experiments in Information Retrieval from Spoken Documents. **Proceedings of the DARPA Workshop on Broadcast News Understanding Systems - BNTUW-98**, p.175-181**.**

HAYES, R.M. (1986) Information Science Education. In: **ALA World Encyclopedia of Library and Information Science.** Chicago: American Library Association.

HAYKIN, S. (2001) **Redes Neurais**: Princípios e prática. Porto Alegre: Bookman.

HOLLAND, J.H. (1998) **Adaptation in natural and artificial systems**: an introductory analysis with applications to biology, control, and artificial intelligence. Cambridge: MIT Press.

INGWERSEN, P. (1992) **Information Retrieval Interaction**. London: Taylor Graham. Disponível em < http://www.db.dk/pi/iri/files/Ingwersen_IRI.pdf> Acessado em 23.10.2003.

JACQUEMIN, C., KLAVANS, J.L. e TZOUKERMANN, E. (1997) Expansion of Multi-Word Terms for Indexing and Retrieval Using Morphology and Syntax. **35th Annual Meeting of the Association for Computational Linguistic (ACL) and 8th Conference of the European Chapter of the ACL**, Madri, p.24-31.

JEAN, G. (2002) **A escrita – memória dos homens**. Rio de Janeiro: Objetiva. (Coleção Descobertas).

JOHNSON, S. (2001) **Cultura da Interface**: como o computador transforma nossa maneira de criar e comunicar. Rio de Janeiro: Jorge Zahar.

JONES, K.S. (1991) The role of artificial intelligence in information retrieval. **Journal of the American Society for Information Science**, v.42, n.8, p.558-565.

JONES, K.S. et al. (1996) Experiments in spoken document retrieval. **Information Processing and Management**, v.32, n.4, p.399-417.

JONES, K.S., WALKER, S. e ROBERTSON, S.E. (2000) A probabilistic model of information retrieval: development and comparative experiments – Part 2. **Information Processing and Management**, v. 36, n. 6, p.809-840.

JONES, K. S. e WILLETT, P. (eds) (1997) . **Readings in Information Retrieval**. San Francisco: Morgan Kaufmann Publishers.

KAJI, H. et al. (2000) Corpus-dependent association thesauri for information retrieval. **18th International conference of computational linguistics** – Coling, Nancy, p.1-7.

KAPPE, F. (1991) **Aspects of a modern multi-media information system**. PhD Thesis, Graz University of Technology, Austria.

KORFHAGE, R.R. (1997) **Information Storage and Retrieval**. New York: John Wiley & Sons.

KOWALSKI, G. (1997) **Information Retrieval Systems**: theory and implementation. Kluwer Academic Publishers

KROVETZ, R. (1997) Homonymy and Polysemy in Information Retrieval. **Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics**, p.72-79.

KROVETZ, R. e CROFT, B.W. (1992) Lexical ambiguity and Information Retrieval. **ACM transactions on Information System**, v. 10, n. 2., p.115-141.

LANCASTER, F.W. (1993) **Indexação e Resumos:** teoria e prática. Brasília: Briquet de Lemos.

LANCASTER, F.W. (1996) **Avaliação de serviços de bibliotecas**. Brasília: Briquet de Lemos.

LANCASTER, F.W. e SANDORE, B. (1997) **Technology and Management in Library and Information Services**. University of Illinois Graduate School of Library and Information Science Science.

LE COADIC, Y-F. (1996) **A ciência da informação**. Brasília: Briquet de Lemos.

LESK, M. (1995) **The seven ages of information retrieval**. Presented on: Conference for the 50th anniversary of "As We May Think", MIT, Cambridge, Massachussets. Disponível em <http://www.ifla.org/VI/5/op/udtop5/udtop5.htm>. Acessado em 13.10.2003.

LEVY, P. (1993) **As tecnologia da inteligência**: o futuro do pensamento na era da informática. Rio de Janeiro: Editora 34.

LEWIS, D.D. (1992) An evaluation of phrasal and clustered representation on a text categorization task. **Proceedings of the 15th annual international ACM SIGIR conference on research and development in information retrieval**, p.37-50.

LEWIS, D.D. e JONES, K.S. (1996) Natural Language Processing for Information Retrieval. **Communications of the ACM**, v. 39, n. 1, p.92-101.

LIDDY, E.D. (1998) Enhanced text retrieval using Natural Language Processing. **Bulletin of the American Society for Information Science**, v. 24, n. 4.

MACHADO, A.M.N. (2003) **Informação e controle bibliográfico**: um olhar sobre a cibernética. São Paulo: Editora UNESP.

MAEDCHE, A. e STAAB, S. (2000) Semi-automatic engineering of ontologies from text. In: **Proceedings of SEKE'00: 12th International Conference on Software Engineering and Knowledge Engineering**. Disponivel em <http://citeseer.nj.nec.com/maedche00semiautomatic.html> Acessado em 14.10.2003.

MARON, M.E. e KUHNS, J.L. (1960) On relevance, probabilistic indexing and information retrieval. **Journal of the ACM**, v. 7, n. 3, p.216-244.

MATTELART, A. (2002) **História da sociedade da informação**. São Paulo: Loyola.

MCCULLOCH, W.S. e PITTS, W.H. (1943) A logical calculus of the ideas immanent in nervous activity. **Bulletin of Mathematical Biophysics**, 5:115-133

MCGARRY, K. (1999) **O contexto dinâmico da informação**. Brasília: Briquet de Lemos.

MINSKY, M.L. (1975) A framework to represent knowledge. **The Psychology of Computer Vision**. McGraw-Hill, p.211-277.

MINSKY, M.L. e PAPPERT, S. (1969) **Perceptron:** An introduction to computational geometry. Cambridge: MIT Press

MITCHELL, M. (2002) **An introduction to genetic algorithms**. 8th printing. Cambridge: MIT Press.

MOLINARI, A. e PASI, G. (1996) A Fuzzy Representation of HTML Documents for Information Retrieval Systems. **Proceedings of IEEE International Conference on Fuzzy Systems**, New Orleans, p.8-12.

MOOERS, C. (1951). Zatocoding applied to mechanical organization of knowledge. **American Documentation**, v. 2, n. 1, p.20-32.

MORGAN, J.J. e KILGOUR, A.C. (1996) Personalising on-line information retrieval support with a genetic algorithm. In: Moscardini, A.O. e Smith, P. (Eds.) **Proceedings of PolyModel 16: applications of artificial intelligence**, pp 142-149.

MORRIS, R.C.T. (1994). Toward a user-centered information science. **Journal of the American Society for Information Science**, v. 45, n.1.

MOZER, M.C. (1984) **Inductive information retrieval using parallel distributed computation**. ICS Technical Report 8406. University of California, San Diego.

NEGROPONTE, N. (1995) **A vida digital**. São Paulo: Companhia das Letras.

ORENGO, V.M. e HUYCK, C.R. (2001) A Stemming algorithm for the Portuguese Language. In: **Proceedings of SPIRE'2001 Symposium on String Processing and Information Retrieval**, Laguna de San Raphael, Chile. Disponível em <http://www.cwa.mdx.ac.uk/chris/Search/stemmer.doc>. Acessado em: 16.10.2003.

ORTEGA, C.D. (2002) **Informática Documentária**: estado da arte. São Paulo, 234p. Dissertação (Mestrado em Ciências da Comunicação) - Escola de Comunicação e Artes, Universidade de São Paulo.

OTLET, P. (1934) **Traité de documentation: le livre sur le livre, théorie et pratique**. Bruxelles: Editiones Mundaneum.

PENZIAS, A. (1992) **Idéias e informação**: operando num mundo de alta tecnologia. Lisboa: Gradiva. (Coleção Ciência Aberta, 55).

PESSIS-PASTERMAK, G. (1993) **Do caos à inteligência artificial**: quando os cientistas se interrogam. São Paulo: Editora UNESP.

QUILLIAN, M.R. (1968) Semantic memory. In: Minsky, M.(ed). **Semantic Information Processing**. Cambridge: MIT Press. p.227-270

RAYWARD, W.B. (1997) The Origins of Information Science and the International Institute of Bibliography/International Federation for Information and Documentation (FID). **Journal of the American Society for Information Science**, v. 48, n. 4, p.289-300.

RICH, E. (1988) **Inteligência Artificial**. São Paulo: McGraw-Hill.

RILLOF, E. (1995) Little words can make a big difference for text classification. **Proceedings of the 18th annual international ACM SIGIR conference on research and development in information retrieval**, p.130-136.

ROBERTSON, S.E. (1977) Theories and models in information retrieval. **Journal of Documentation**, 33, p.126-148.

ROBERTSON, S.E. e JONES, K.S. (1976) Relevance weighting of search terms. **Journal of the Americam Society for Information Science**, v. 27, n. 3, p.129-146.

ROBREDO, J. e CUNHA, M.B. (1994) **Documentação de hoje e de amanhã**: uma abordagem informatizada da biblioteconomia e dos sistemas de informação. São Paulo: Global.

ROBREDO, J. (2003) **Da Ciência da Informação revisitada aos sistema humanos de informação**. Brasília: Thesaurus.

ROSENBLATT, F. (1958) The perceptron: a probablistic model for information storage and retrieval in the brain. **Psychological Review**, v. 65, p.386-408.

ROWLEY, J. (2002) **A biblioteca eletrônica.** Brasília: Briquet de Lemos.

RUBIN, R.E. (2000) **Foundations of library and information science**. New York: Neal-Schuman.

RUYER, R. (1972) **A cibernética e a origem da informação**. Rio de Janeiro: Paz e Terra.

SACCONI, L.A. (1999) **Nossa gramática**: teoria e prática. São Paulo: Atual.

SALTON, G. (ed.) (1971). **The SMART retrieval system**: experiments in automatic document processing. Prentice-Hall.

SALTON, G. (1972). A new comparison between conventional indexing (MEDLARS) and automatic text processing (SMART), **Journal of the American Society for Information Science**, v. 23, n. 2, p.74-84.

SALTON, G. (1973). Recent studies in automatic text analysis and document retrieval, **Journal of the ACM**, v. 20, n. 2, p.258-278.

SALTON, G. (1984) **The use of extended Boolean logic in information retrieval**. Technical Report TR 84-588, Cornell University, Computer Science Dept., Ithaca, N.Y.

SALTON, G. e BUCKLEY, C. (1988) Term-Weighting Approaches in Automatic Text Retrieval. **Information Processing and Management**, v. 24, n. 5, p.513-523.

SALTON, G., FOX, E.A., WU, H. (1983) Extended Boolean Information Retrieval. **Communication of the ACM**, v. 26, n. 11, p.1022-1036.

SALTON, G. e LESK, M.E. (1968) Computer evaluation of indexing and text processing. **Journal of the ACM**, v. 15, n. 1, p.8-36.

SALTON, G. e MCGILL, M. J. (1983) **Introduction to Modern Information Retrieval**. McGraw Hill.

SANTOS, D. (1996) Português Computacional. In: Duarte, I., Leiria, I. (ed.). **Actas do Congresso Internacional sobre o Português**. Lisboa: Edições Colibri. p.67-184.

SANTOS, D. (2001) Introdução ao processamento de linguagem natural através das aplicações. In: Ranchhod, E. (ed.) **Tratamento das Línguas por Computador**: Uma introdução à lingüística computacional e suas aplicações, Lisboa: Caminho, p.229-259. Disponível em <http://www.linguateca.pt/Diana/public.html>. Acessado em 01.08.2003.

SARACEVIC, T. (1995) Interdisciplinary nature of information science. **Ciência da Informação**. v. 24, n. 1, p.36-31.

SARACEVIC, T. (1996) Ciência da informação: origem, evolução e relações. **Perspectivas em Ciência da Informação**, v. 1, n. 1, p.41-62.

SARACEVIC, T. (1999) Information Science. **Journal of the American Society for Information Science**, v. 50, n. 12, p.1051-1063.

SCHAMBER, L. (1996) What is a Document? Rethinking the concept in uneasy times. **Journal of the Americam Society for Information Science**, v. 47, n. 9, p.669-671.

SCHULTZ, C. K. (ed.) (1968) **H.P. Luhn: Pioneer of information science** - selected works. New York: Spartan Books.

SETZER, V.W. (2001) **Meios Eletrônicos e Educação**: uma visão alternativa. São Paulo: Escrituras.

SHANNON, C. e WEAVER, W. (1949) **The Mathematical theory of communication**. University of Illinois Press.

SHAW, I.S. e SIMÕES, M.G. (1999) **Controle e modelagem *fuzzy***. São Paulo: Edgard Blücher.

SHERA, J.H e CLEVELAND, D.B. (1977) History and foundations of Information Science. **Annual Review of Information Science and Technology**, v. 12, p.249-275.

SMEATON, A.F. (1997) Information Retrieval: still butting heads with natural language processing. In: PAZIENZA, M.T. (ed.) **Information Extraction: a multidisciplinary approach to an emerging information technology**. Springer-Verlag Lecture Notes in Computer Science, n. 1299, p.115-138.

SMIT, J. (1987) **O que é documentação**. São Paulo: Brasiliense. (Coleção Primeiros Passos, 174).

SMIT, J. (coord.) (1987) **Análise Documentária**: a análise da síntese. Brasília: IBICT.

SMITH, E.S. (1993) On the sholders of giants: from Boole to Shannon to Taube: the origins and development of computerized information from the mid-19th century to the present. **Information Technology and Libraries**, n. 12, p.217-226.

SOWA, J. F. (2000) **Knowledge representation**: logical, philosophical, and computational foundations. Pacific Grove, CA: Brooks/Cole.

STOCKWELL, F. (2001) **A history of information storage and retrieval**. Jefferson: McFarland.

STRATHERN, P. (2001) **Darwin e a evolução em 90 minutos**. Rio de Janeiro: Jorge Zahar.

TÁLAMO, M.F. (1997) Informação: organização e comunicação. **Seminário de Estudos de Informação da Universidade Federal Fluminense**, 1, 1996 Anais... Niterói, Rio de Janeiro : EDUFF, p.11-14.

TEIXEIRA, J.F. (1998) **Mentes e máquinas**: uma introdução à ciência cognitiva. Porto Alegre: Artes Médicas.

TENÓRIO, R.M. (1998) **Cérebros e computadores**: a complexidade analógico-digital na informática e na educação. São Paulo: Escrituras. (Série ensaios transversais).

TONG, R.M. et al. (1985) RUBRIC: An environment for full text information retrieval. **Proceedings of the 8th annual international ACM SIGIR conference on research and development in information retrieval**, p.243-251.

TONG, R.M. et al. (1987) Conceptual Information Retrieval Using RUBRIC. **Proceedings of the 10th annual international ACM SIGIR conference on research and development in information retrieval**, p.247-253.

USCHOLD, M. (2000) Creating, integrating and maintaining local and global ontologies. **Workshop on Applications of Ontologies and Problem-Solving Methods - 14th European Conference on Artificial Intelligence, 2000**. Disponível em <http://delicias.dia.fi.upm.es/WORKSHOP/ECAI00/13.pdf> Acessado em 14.10.2003.

VAN RIJSBERGEN, C.J. (1979) **Information retrieval**. London: Butterworths. Disponível em < http://citeseer.nj.nec.com/vanrijsbergen79information.html>. Acessado em 25.10.2003.

VRAJITORU, D. (2000) Large Population or Many Generations for Genetic Algorithms? Implications in Information Retrieval. In: Crestani, F., Pasi, G. (eds.): **Soft Computing in Information Retrieval. Techniques and Applications**, Physica-Verlag, Heidelberg, p.199-222.

WOODS, W.A. (1975) What's in a link: Foundations for semantic networks. In: Bobrow, D.G. e Collins , A. (eds). **Representation and Understanding: Studies in Cognitive Science**. Academic Press, New York.

WURMAN, R.S. (1991) **Ansiedade de Informação**: como transformar informação em compreensão. São Paulo: Cultura Editores Associados.

YAGER, R.R. (1988) On ordered weighted averaging aggregation operators in multi-criteria decision making, **IEEE transactions on Systems, Man and Cybernetics**, v. 18, p.183-190.

ZADEH, L.A. (1965) Fuzzy sets. **Information and Control**, v. 8, n. 3, p.338-353.