FLAVIA DO CANTO SAKS

BUSCA BOOLEANA: TEORIA E PRÁTICA

CURITIBA
2005

FLAVIA DO CANTO SAKS

BUSCA BOOLEANA: TEORIA E PRÁTICA

Trabalho de Conclusão de Curso apresentado à disciplina Pesquisa em Informação II do Curso de Gestão da Informação, Setor de Ciências Sociais Aplicadas, Universidade Federal do Paraná.

Orientador: Prof. Ulf Gregor Baranow

CURITIBA
2005

SUMÁRIO

LISTA DE ILUSTRAÇÕES

RESUMO

Trata de uma introdução à busca booleana, aplicação da Lógica de Boole a um tipo de sistema de recuperação da informação, em que se combinam termos relacionados com auxílio de operadores lógicos. A elaboração deste trabalho foi motivada pela inexistência de uma fonte de informação introdutória mais abrangente em língua portuguesa sobre os recursos da busca booleana, especialmente destinada para gestores da informação. São descritas as modalidades e mecanismos básicos da busca booleana e sua respectiva representação em tabelas-verdade, assim como alguns recursos associados, utilizados comumente na recuperação da informação. Apresentam-se exemplos de sua aplicação e um estudo de caso realizado em cinco sistemas de busca na Internet. Enquanto levantamento de fontes, o trabalho baseia-se em materiais disponíveis sobre o tema, sendo a maior parte em língua inglesa, obtidos principalmente pela Internet. São apresentadas sugestões para dar continuidade ao tratamento do tema em futuros trabalhos.

1 INTRODUÇÃO

As possibilidades de acesso aos grandes sistemas de recuperação de informação e às bases de dados vieram ampliar significativamente a qualidade das buscas, visto que essas bases proporcionam diversificados pontos de acesso à informação. Esses sistemas de acesso possibilitam o planejamento de estratégias de busca com maior nível de complexidade ao envolver vários conceitos. Permitem, por exemplo, a busca por palavras apenas dos títulos, dos resumos e do próprio texto de documentos, com ou sem o auxílio de linguagens controladas (descritores). Nas expressões de busca, os conceitos podem ser expressos por termos compostos ou simples. Existe a possibilidade de truncagem de palavras e substituição de caracteres no meio das palavras, dentre outros recursos de recuperação.

> "Os conteúdos a serem recuperados por meio dessas técnicas de busca variam, de acordo com a diversidade temática das respectivas bases de dados oferecidas em cada banco de dado. Algumas bases são orientadas para um determinado assunto, enquanto outras são orientadas para a missão da instituição que as desenvolver. Esses e outros fatores devem ser levados em consideração no planejamento das estratégias de busca, assegurando a qualidade na recuperação da informação". (LOPES, 2000, p. 60-61)

Na recuperação da informação, a estratégia de busca pode ser definida como técnica ou conjunto de regras para tornar possível o encontro entre uma pergunta formulada e a respectiva informação armazenada numa base de dados. Dessa forma, a partir de um arquivo, será selecionado um conjunto de itens que constituem a resposta a uma pergunta, apresentada sob a forma de uma questão ou expressão de busca.

Preparar uma estratégia de busca, bem como a seleção de um banco ou uma base de dados a ser consultada para responder a uma pergunta específica, pode exigir do usuário determinados conhecimentos (idiomas, mecanismos de busca do banco de dados, instrumentos de auxílio para identificação da terminologia, lógica booleana e outros recursos disponibilizados). A implementação da estratégia de busca requer, portanto, conhecimentos técnicos e de conteúdo específicos para operacionalização da mesma.

Quando a estratégia de busca for preparada por um profissional de informação para um usuário, este deve fornecer, preferencialmente em formulário específico, os seguintes dados: título sucinto e breve definição do problema; termos apropriados para o tópico de interesse e, eventualmente, uma lista de termos que não são desejados. A partir desses dados, o profissional poderá executar a busca de informação.

Atualmente, o usuário comum, em geral, não pode mais depender de um profissional da informação; ele terá de adquirir conhecimentos básicos sobre os procedimentos de busca adequados a necessidades informacionais específicas. A tecnologia de busca que acabou por ser adotada na grande maioria das bases de dados, inclusive na Internet, é a chamada busca booleana – objetivo da presente monografia.

2 TEMA

Os usuários da Internet podem achar um sistema de busca difícil de manipular. Mesmo aqueles que possuem conhecimentos na área já experimentaram as frustrações de qualquer pessoa que empreende uma pesquisa: falta de bancos de dados na área desejada, resultados de pesquisa aparentemente não relacionados e, freqüentemente, a incapacidade do próprio usuário em combinar termos utilizando procedimentos booleanos-padrão.

Para garantir um resultado satisfatório e a qualidade da informação recuperada, é necessário conhecer elementos de Lógica Booleana, pois uma estratégia de busca adequada exige, por vezes, a elaboração de expressões complexas. O usuário pode igualmente utilizar o conhecimento da Lógica Booleana para compreender e avaliar os resultados obtidos numa busca.

A Lógica Booleana é assim denominada, de acordo com George Boole (1815-1864), matemático e lógico britânico que desenvolveu a teoria da lógica binária, na qual existem somente dois valores possíveis 0 ou 1/ **verdadeiro** ou **falso.** Posteriormente, essa teoria foi aplicada ao funcionamento de circuitos eletrônicos, sendo também fundamental na arquitetura de computadores.

A busca booleana é a aplicação da Lógica de Boole a um tipo de sistema de recuperação da informação, no qual se combinam dois ou mais termos, relacionando-os por operadores lógicos, que tornam a busca mais restrita ou detalhada. As estratégias de busca são baseadas na combinação entre a informação contida em determinados documentos e a correspondente questão de busca, elaborada pelo usuário do sistema.

FERNEDA (2003, p.27) observou que o modelo de busca booleana pode ser considerado o modelo hoje mais utilizado, não só nos sistemas de recuperação de informação e nos mecanismos de busca da *web,* mas também nos sistemas de bancos de dados, onde a busca é expressa através da linguagem SQL. Em geral, encontra-se presente, seja como a principal maneira de formular as expressões de busca, seja como recurso alternativo. Isto, porque este modelo oferece, para usuários experientes, um certo controle sobre o sistema. No caso em que o conjunto de documentos recuperados pelo sistema for muito grande ou muito pequeno, o usuário do sistema saberá quais os operadores a serem aplicados na busca até atingir um resultado satisfatório.

Elaborou-se no presente trabalho uma descrição da busca booleana em suas principais modalidades utilizadas na prática da recuperação informacional. As informações e explicações sobre o assunto encontram-se dispersas, sendo geralmente tratadas de modo bastante sucinto. Direcionados aos usuários da Rede, os *sites* de busca e muitas páginas na Web oferecem apenas aquelas explicações operacionais consideradas indispensáveis para o usuário. Em muitas delas, omite-se qualquer referência à busca booleana, embora o mecanismo de busca básico ainda continua sendo o mesmo.

Falta, portanto, um tratamento mais abrangente da busca booleana, que não se limite apenas a explicações sobre comandos operacionais, mas que não avance tampouco para uma abordagem propriamente informática. Baseando-se em fontes, este enfoque será aqui chamado de "perspectiva do usuário", na qual não se trata de analisar o comportamento do usuário ao utilizar a busca booleana, mas de oferecer conhecimentos básicos para a sua realização. A parte descritiva será complementada pela aplicação prática de um estudo de caso.

3 JUSTIFICATIVA

A principal importância da aplicação dos operadores lógicos booleanos em buscadores na Internet reside na necessidade dos usuários de utilizar esta ferramenta para localizar os documentos que eles desejam recuperar.

O sistema de busca booleana ajuda a percorrer documentos, recuperar informações sobre determinados assuntos e localizar recursos informacionais de interesse. Esses recursos apresentam interfaces com centenas de serviços e bancos de dados da Internet, oferecendo formas mais fáceis de localizar informações.

A busca booleana possibilita a localização de *sites* e páginas que contêm determinados assuntos ou abordam determinados aspectos de um assunto. Faz com que o *site* seja localizado, da maneira mais fácil, pelo usuário-consulente-internauta ou pelo consumidor de um determinado produto divulgado na Internet.

O problema que motivou a elaboração do presente projeto é a inexistência de uma fonte de informação mais abrangente em língua portuguesa, sobre o recurso da busca booleana, a partir da perspectiva do usuário que se utiliza desse sistema em bases de dados em geral, inclusive na Internet.

A escolha do tema da busca booleana na recuperação da informação foi também motivada pela experiência anterior da autora[*], na aplicação da Lógica Booleana a circuitos eletrônicos.

4 OBJETIVOS

O presente trabalho subordina-se a um objetivo geral e a três objetivos específicos.

4.1 OBJETIVO GERAL

Apresentar uma descrição da busca booleana voltado para a perspectiva do usuário de sistemas de informação.

4.2 OBJETIVOS ESPECÍFICOS

a) Explicar as modalidades e mecanismos básicos da busca booleana;
b) Descrever recursos complementares associados à busca booleana;
c) Apresentar uma aplicação da busca booleana.

5 METODOLOGIA

O presente trabalho resultou, basicamente, de uma pesquisa classificada como bibliográfica, isto é, a partir de materiais já publicados sobre o tema (livros, artigos de periódicos, teses etc).

---

[*] Curso de Tecnologia em Eletrônica, realizado pela autora no CEFET de 1999 à 2001.

A abordagem do assunto restringiu-se à descrição dos recursos booleanos básicos na recuperação da informação. A chamada busca booleana estendida ou avançada (cf. KORFHAGE, 1997, p. 65 ss.) não é tratada nesta monografia.

O processo de coleta de informações sobre o tema foi realizado conforme as seguintes etapas:

a) busca e seleção das principais fontes disponíveis sobre tópicos referentes à busca booleana;
b) análise sistemática dos conteúdos levantados;
c) listagem dos principais aspectos sobre o tema;
d) verificação da abordagem dos tópicos nas fontes escolhidas;
e) consolidação e sistematização do conhecimento obtido; e
f) elaboração do texto da monografia.

A coleta das informações a partir das fontes disponíveis em inglês, espanhol e português abrangeu, portanto, publicações impressas, e principalmente fontes obtidas na Internet.

6 OPERADORES DA LÓGICA BOOLEANA E RECURSOS ASSOCIADOS NA RECUPERAÇÃO DA INFORMAÇÃO

A Internet já tem sido comparada a um vasto banco de dados. De fato, seus conteúdos são pesquisados de acordo com as mesmas regras de busca válidas também para bases de dados. Em geral, a busca em bases de dados é realizada de acordo com os princípios da chamada lógica booleana, baseada nas relações lógicas entre os termos de busca (COHEN, 2004).

Sistemas de recuperação booleana surgiram desde a tecnologia dos cartões perfurados. Por essa metodologia era possível separar documentos nos quais existe um determinado termo daqueles que não o têm. Esta operação pode ser facilmente transposta para a lógica de recuperação booleana. Em um sistema desse tipo, as questões de busca não se referem a simples pontos no espaço informacional dos documentos. De acordo com KORFHAGE (1997, p. 81), cada questão de busca constitui uma função lógica de palavras existentes neste espaço. Não se define aqui o documento no sentido tradicional. Como não há semelhança estrutural entre o documento e a questão de busca, esta última é considerada como entidade à parte. A recuperação a partir de uma determinada questão, pode ser vista como uma função característica definida do espaço do documento.

As dificuldades com os sistemas de recuperação booleana foram reconhecidas tão logo passaram a ser utilizados. Há mais de trinta anos (1972) LANCASTER afirmava que o uso da álgebra de Boole para fazer a busca em sistemas de recuperação computadorizados pode ter sido um erro, uma vez que já naquela época existiam propostas de recuperação informacional mais sofisticadas.

Um sistema de recuperação puramente booleano não dispõe de nenhum recurso para julgar semelhanças significativas. Por definição, um determinado documento satisfaz a busca booleana ou não. Em princípio, pode-se considerar o mapeamento definido por uma questão de busca como uma função característica pela qual o espaço documentário é dividido em dois conjuntos distintos: os documentos que correspondem e os documentos que não correspondem à questão de busca. (KORFHAGE, 1997, p. 82)

6.1 OPERADORES DA LÓGICA BOOLEANA

Os operadores booleanos baseiam-se na álgebra de Boole e permitem efetuar operações de caráter lógico-matemático. Estes operadores são: AND (E), OR (OU) e NOT (NÃO), e eles são usados para combinar palavras-chave por ocasião na busca em

bases de dados eletrônicos. O uso destes operadores pode tornar a busca mais enfocada, produzindo resultados mais precisos. No entanto, antes de utilizar os operadores, é necessário entender como eles, de fato, trabalham. (RICH, 2004)

Freqüentemente, os conceitos booleanos são explicados por meio dos diagramas de Venn. No presente caso, os diagramas de Venn podem representar o espaço de informação da Rede. Um círculo contendo uma palavra mostra o subconjunto de documentos da Rede que contêm esta palavra. Quando há dois conceitos em um único diagrama, a interseção entre os círculos representa aqueles documentos que contêm ambos os conceitos. (WEB ARCHITECTS, 2004)

Outra representação gráfica dos operadores booleanos se dá através da tabela-verdade, a qual consiste no diagrama onde são representados e analisados todos os resultados possíveis de uma decisão complexa, baseada em vários fatores. É utilizada como ferramenta de análise de sistema e planejamento de programas de computador. É uma forma alternativa de apresentação de um algoritmo para resolução de determinado tipo de problema. Cada operador tem sua tabela-verdade, que o caracteriza. Também é utilizada, ás vezes, em materiais didáticos para representar um processo decisório complexo.

6.1.1 OR

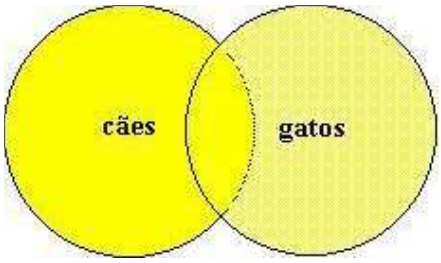
O computador não é um ser pensante; ao pedir-lhe uma informação sobre “casa” ele não pode por si mesmo buscar também outros termos sinônimos como: “vivenda”, “edifício”, “habitação”, “morada”, etc. É preciso que estes termos sejam pedidos de forma explícita.

Assim, para aumentar as possibilidades de recuperar a maior quantidade de informação, é necessário utilizar sinônimos e termos relacionados ou similares ao termo de busca original. Desse modo, será possível recuperar o maior número possível de documentos, independentemente de qual dos termos foi utilizado na busca. Neste caso, relacionam-se os termos com o operador OR.

Os termos não devem ser necessariamente sinônimos em qualquer contexto, mas simplesmente naquele que interessa ao usuário naquele momento.

Tomemos o conjunto de documentos com o termo “cães”, e unimos este com outro conjunto de documentos com o termo “gatos”, formando um terceiro conjunto – a união de ambos – que conterá documentos com o primeiro, o segundo ou ambos os termos.

FIGURA 1 – USO DE ... OR ... [CÃES OR GATOS]



Nesta busca vamos recuperar registros, nos quais PELO MENOS UM dos termos de busca estará presente. Estamos procurando os termos “cães” e também “gatos”, visto que os documentos que contêm qualquer uma destas palavras podem ser relevantes para minha pesquisa.

Pode ser ilustrado do seguinte modo:

a) círculo colorido com a palavra “cães” representa todos os registros que contém esta palavra;
b) círculo colorido com a palavra “gatos” representa todos os registros que contêm a palavra “gatos”;
c) a área de interseção representa todos os registros que contêm ao mesmo tempo “cães” e “gatos”. (COHEN, 2004)

Eis um exemplo de como pode funcionar o operador OR:

TABELA 1 – RESULTADO DA BUSCA OR

Termos de busca	Resultados
Cães	1.490.000
Gatos	1.690.000
Cães OR gatos	1.880.000

Por meio do operador OR, recuperam-se todos os registros que contêm um dos dois termos ou ambos.

Quanto mais termos ou conceitos combinarmos em uma busca com o operador OR, mais registros vamos recuperar. Por isso, de acordo com ROWLEY (2003, p. 121), o operador OR desempenha uma função aditiva, tendo como resultado uma soma lógica.

Exemplo:

TABELA 2 – RESULTADO DA BUSCA OR COM TRÊS TERMOS

Termos de busca	**Resultados**
Cães	1.490.000
Gatos	1.690.000
Pulgas	166.000
cães OR gatos OR pulgas	1.960.000

Vejamos um outro exemplo: para recuperar um documento, é necessário que este contenha o termo de busca. Assim, no caso de se assegurar a recuperação de todo e qualquer registro que mencione “América Central” ou qualquer país dessa região, é necessário que se utilize a seguinte expressão de busca:

América Central OR Centroamérica OR Guatemala OR Honduras OR Panamá OR Costa Rica.

Portanto, se não estivermos seguros de que se tenha utilizado o termo América Central, no documento procurado, teremos de formar o conjunto desejado com a união dos conjuntos individuais para garantir a recuperação. Esta característica pode causar buscas prolongadas, quando não se quer perder informação.

Atualmente, os bancos de dados costumam indicar quantas referências estão disponíveis para cada termo de busca. Ao combinar estes termos com o operador OR, cria-se um novo conjunto formado pela união dos conjuntos individuais. Isto se deve ao fato de existirem documentos – e por tanto também conjuntos de documentos – que contenham simultaneamente dois ou mais termos, por exemplo tanto X como Y.

Por isso, pode dar-se o caso de que um mesmo documento apareça no conjunto X e também no conjunto Y. Mas tratando-se de um só documento, ele aparecerá uma só vez no conjunto X OR Y.

Esta propriedade permite fazer uma busca com maior número de termos, sem que o resultado da união inclua documentos repetidos – o operador OR os descarta.

Ao buscar informações armazenadas em um sistema, as buscas são feitas comparando caractere por caractere do termo solicitado com os termos armazenados. Um só caractere distinto significará para o sistema uma palavra diferente e, portanto, não desejada para a busca em questão.

Por outro lado, os sistemas com capacidade de fazer uma expansão ou o truncamento de termos permitem localizar, também, termos armazenados com erros de ortografia. Por isso, naqueles casos em que se pode prever diferenças de ortografia ou a possibilidade de termos no singular ou plural, o operador OR pode servir para evitar recuperações incompletas. (MARBÁN, 1987, p.12)

Em relação ao uso da tabela-verdade (ver página 9), o operador OR pode ser representado pela expressão Z = (cães OR gatos). A razão para a terminologia adotada reside em que Z = V, se "cães" = V ou "gatos" = V, ou ainda, se ambos "cães" e "gatos" = V, como se verifica na tabela verdade. (TAUB, 1984, p.6)

TABELA 3 – TABELA-VERDADE PARA O OPERADOR OR

Cães	gatos	Z = cães OR gatos
F	F	F
F	V	V
V	F	V
V	V	V

Quando em um *site* da Internet não consta explicitamente a opção de trabalhar com o operador OR, deve-se consultar a respectiva página sobre "Pesquisa Avançada", onde, freqüentemente, se encontram disponíveis uma ou mais modalidades de busca para qualquer um dos termos desejados. No caso dos mecanismos de busca que não utilizam OR, os símbolos | ou + são outras alternativas. Deve-se estar atento,

porque alguns mecanismos de busca executam uma busca com OR por *default.* (SULLIVAN, 2004).

Resumindo:

a) operador OR requer pelo menos um dos termos ligados por este operador em algum lugar no documento, em qualquer ordem;
b) ao utilizar OR, amplia-se a busca, porque um número maior de documentos vai satisfazer esse critério; qualquer um dos termos será suficiente para o documento recuperado;
c) quanto mais palavras entram conectadas por OR, mais documentos serão obtidos.

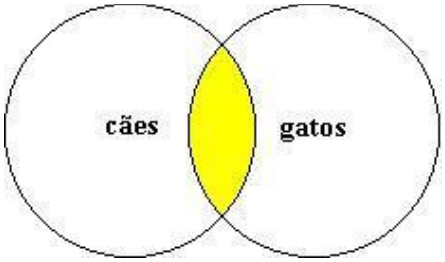
6.1.2 AND

Vimos que, ao realizar-se uma busca booleana utilizando o operador OR, recuperam-se todos aqueles documentos existentes no sistema que correspondem a qualquer um dos termos usados nesta busca (ver seção 6.1.1).

Mas o volume de documentos recuperados pode ser excessivo. Assim, devem-se estabelecer prioridades ou agrupar os termos de busca em subconjuntos segundo diferentes aspectos. Normalmente, busca-se então a informação em um nível mais específico.

Suponhamos que se deva restringir a busca àqueles documentos relacionados com “cães”, tentando-se localizar os documentos que simultaneamente contenham os termos “cães” e “gatos”. Daí resulta uma intersecção de diferentes conjuntos. Esta intersecção se realiza com o operador AND, obtendo-se assim um novo conjunto.

FIGURA 2 – USO DE ... AND … [CÃES AND GATOS]



Nesta busca, vamos recuperar os registros nos quais AMBOS os termos de busca estiverem presentes. Isto é ilustrado pela área colorida na interseção dos dois círculos que representa todos os registros que contêm ao mesmo tempo a palavra "cães" e a palavra "gatos". Note-se que não recuperamos nenhum registro que contém só a palavra "cães" ou só a palavra "gatos". (COHEN, 2004)

Eis um exemplo de como funciona o operador AND:

TABELA 4 – RESULTADO DA BUSCA AND

Termo de busca	**Resultados**
Cães	1.490.000
Gatos	1.690.000
cães AND gatos	612.000

Quanto mais termos ou conceitos combinarmos em uma busca com o operador AND, menos registros vamos recuperar. Conforme ROWLEY (2003, p. 121), o operador AND executa um tipo de busca conjuntiva, levando a um produto lógico. Este é fundamentalmente diferente de um produto resultante de simples soma, como poderia sugerir o uso da proposição E (AND) na linguagem comum.

Por exemplo:

TABELA 5 – RESULTADO DE BUSCA AND COM TRÊS TERMOS

Termos de busca	**Resultados**
Cães	1.490.000
Gatos	1.690.000
Pulgas	166.000
cães AND gatos AND pulgas	1.000

Ao formar um novo conjunto por intersecção – de todos os documentos relacionados com o termo X, extraímos somente aqueles que explicitamente contenham o termo Y. Desta forma, de todos os documentos relacionados com Y, recuperamos unicamente os que também mencionam o termo X.

Por esta razão, o conjunto resultante nunca poderá ser maior em número de registros que o menor dos conjuntos interseccionados; em geral será menor.

Recorde-se que no caso do operador OR o termo **pode** estar presente, mas com o operador AND o termo **deve** estar presente. Com outras palavras, com o operador AND recuperamos apenas aqueles documentos que contenham ambos os termos. (MARBÁN, 1987, p. 20)

Resumindo, cada vez que se agrega um termo ao operador AND, está-se impondo maiores limitações ao conjunto de busca, o que geralmente reduz a quantidade de documentos que cumprem simultaneamente todas as condições formuladas na busca.

O operador AND pode ser definido pela tabela-verdade (ver p. 9), utilizando a expressão: Z = (cães AND gatos), por exemplo. A razão para a terminologia adotada reside em que Z = V (V representa que documentos serão recuperados na busca) somente quando “cães” e “gatos” são ambos verdadeiros, isto é, essas palavras estão contidas nos documentos recuperados. Este operador sugere que Z é o resultado de uma “multiplicação” na qual “cães” e “gatos” são os fatores. (TAUB, 1984, p. 5)

TABELA 6 – TABELA-VERDADE PARA O OPERADOR AND

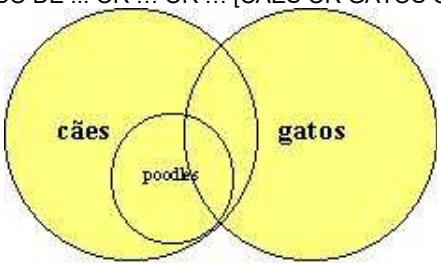
cães	gatos	Z = cães AND gatos
F	F	F
F	V	F
V	F	F
V	V	V

Em alguns mecanismos de busca que não utilizam explicitamente AND, o símbolo & é a alternativa. É preciso estar atento, pois alguns mecanismos de busca executam a busca AND por *default* (SULLIVAN, 2004).

Às vezes acontece de se construírem buscas com termos redundantes. Não são realmente prejudiciais, mas é importante entender o que está acontecendo. Assim, se a busca não recuperar aqueles documentos que se deseja, deve-se reeditar a formulação da busca de maneira mais adequada.

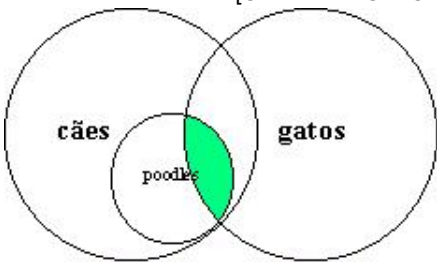
Suponhamos que se esteja fazendo uma busca no espaço informacional representado no diagrama a seguir:

FIGURA 3 – USO DE ... OR … OR … [CÃES OR GATOS OR POODLES]



Suponhamos ainda que se necessite de documentos tratando de *poodles* e gatos criados juntos; trata-se da área colorida no diagrama a seguir.

FIGURA 4 – USO DE … AND … AND … [CÃES AND GATOS AND POODLES]



Seria possível estabelecer a busca: cães AND gatos AND *poodles*, mas a palavra "*poodles*" é redundante, pois todos os documentos sobre *poodles* são também sobre cães. Só a busca gatos AND *poodles* delimitará a área correta.

Uma regra geral aconselhável, pelo menos para iniciantes, é elaborar primeiro uma expressão de busca simples, passando às expressões mais complicadas somente quando não se encontrar a informação desejada.

Resumindo:

a) o operador AND requer que todos os termos (palavras ou frases entre aspas) ligados por este operador constem em algum lugar no documento, em qualquer ordem;
b) ao utilizar AND, a busca é afunilada, pois com este operador aumenta-se o número de palavras necessárias em um determinado documento para que satisfaça o critério de busca;
c) quanto mais palavras forem conectadas por AND, menor será o número de documentos recuperados.

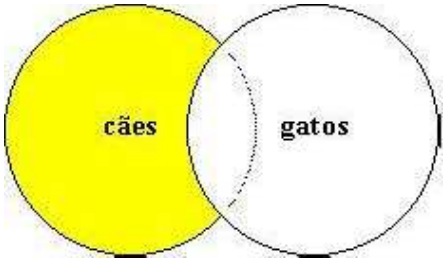
6.1.3 NOT

Por vezes, necessita-se excluir determinados registros, seja porque se referem a aspectos que já se conhecem ou porque, no momento, não interessam ao pesquisador. Esta exclusão se consegue pelo operador NOT.

Este operador é útil para trabalhar em sistemas que permitem selecionar e excluir explicitamente campos não necessariamente temáticos. Entretanto, sendo este o operador mais restritivo dos três, é preciso utilizá-lo com cuidado.

Suponhamos que se deseja apenas informações sobre “cães”, mas não sobre “gatos”. Nesta busca vamos recuperar registros nos quais apenas um desses termos estará presente. Isto é ilustrado pela área colorida, representando todos aqueles registros que contêm a palavra “cães”. Nenhum registro em que consta a palavra “gatos” é recuperado, mesmo se nele estiver também a palavra “cães” (COHEN, 2004).

FIGURA 5 – USO DE NOT [CÃES NOT GATOS]



Segue um exemplo de como funciona o operador NOT:

TABELA 7 – RESULTADO DA BUSCA NOT

Termos de busca	**Resultado**
Cães	1.490.000
Gatos	1.690.000
Cães NOT gatos	9.350

O operador NOT exclui determinados registros dos seus resultados de busca. É preciso ter muito cuidado ao empregar o NOT, pois o termo e, portanto, o documento que se quer recuperar, pode ter uma presença destacada em documentos que também contêm a palavra que se excluiu.

Note-se que “X OR Y” resulta no mesmo que “Y OR X”, e da mesma forma “X AND Y” equivale a “Y AND X”. Isto porém, não acontece com o operador NOT, pois o resultado da expressão “X NOT Y” é totalmente diferente de “Y NOT X”. Na terminologia da lógica booleana diz-se que os operadores OR e AND são comutativos, enquanto que o operador NOT não o é (MARBÁN, 1987. p. 24-26).

Trata-se, portanto, de um operador com ação restritiva substitutiva, conforme ROWLEY (2003, p. 121). Sendo o conjunto resultante menor ou igual ao conjunto do qual foram eliminados os termos, este operador produz uma diferença lógica.

O operador NOT possui uma “função inversora”, isto é, em uma expressão Z = (NOT cães), quando “cães” = V, terá como resultado Z = F (isto é, documentos com tal palavra não serão mencionados na busca) e vice versa (TAUB, 1984. p. 5).

TABELA 8 – TABELA-VERDADE PARA O OPERADOR NOT

Cães	Z = NOT cães
F	V
V	F

Caso o mecanismo de busca não utilize explicitamente o operador NOT, os símbolos – ou ~ são alternativas.

Resumindo:

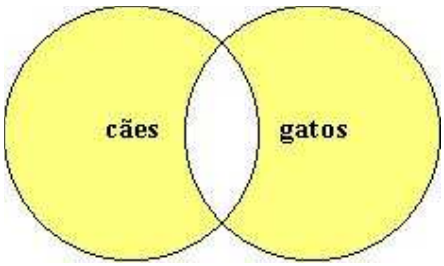
a) este operador exclui todos os documentos que contêm a(s) palavra(s) que vêm depois de NOT;
b) utilizando NOT, limita-se uma busca, porque este operador desqualifica documentos, não importa se eles satisfazem ou não outro critério da busca;
c) o operador NOT geralmente é utilizado após ter executado uma busca, examinado os resultados e concluído que não se necessita de determinadas páginas ou documentos recuperados em função de alguma palavra ou frase.

6.1.4 XOR

Vimos que em uma busca booleana, o uso de AND requer que ambos os termos estejam presentes nos documentos recuperados. Já no caso de OR exige-se que pelo menos um dos termos esteja presente no resultado da busca. Neste caso, trata-se de um uso inclusivo de OR. Isto significa ser aceitável que ambos os termos estejam presentes no resultado de busca (KORFHAGE, 1997, p. 54). Entretanto, alguns sistemas de bases de dados permitem o operador "OR exclusivo" ou XOR, isto é, ou um ou outro termo, mas não ambos. Neste caso, a construção da busca será mais complexa, pois o operador NOT, obviamente, requer que o termo especificado esteja ausente em todo e qualquer documento recuperado.

Suponhamos que se deseja recuperar documentos que contenham a palavra "cães" e também "gatos", mas não se tem interesse em documentos que possuem ambas as palavras simultaneamente. (MARBÁN, 1987, p. 36-37)

FIGURA 6 – USO DE XOR [CÃES XOR GATOS]



O nome deste operador (EXCLUSIVE-OR) origina-se do fato que Z = (cães XOR gatos) = V quando uma das palavras “cães” ou “gatos”, na exclusão da outra, estiver mencionada no documento. Assim, Z = V se “cães” = V ou se “gatos” = V, mas não se ambos forem V. (TAUB, 1984. p. 5)

TABELA 9 – TABELA-VERDADE PARA O OPERADOR XOR

Cães	Gatos	Z = cães OR gatos
F	F	F
F	V	V
V	F	V
V	V	F

6.2 OPERADORES BOOLEANOS COMBINADOS

Ao buscar informação em bancos de dados, pode-se usar vários operadores, mesmo que repetidos. Trata-se então, de expressões de busca mais complexas, com as quais se visa chegar a resultados mais sofisticados.

Ao combinar operadores, os sistemas o fazem de acordo com determinadas prioridades previamente estabelecidas, com ligeiras variações de um sistema para outro.

Note-se que os operadores AND e NOT, geralmente, limitam a busca (diminui o número de documentos recuperados), enquanto o operador OR vai ampliá-la (aumenta o número de documentos recuperados). Observe-se, portanto, as seguintes estratégias (RICH, 2004):

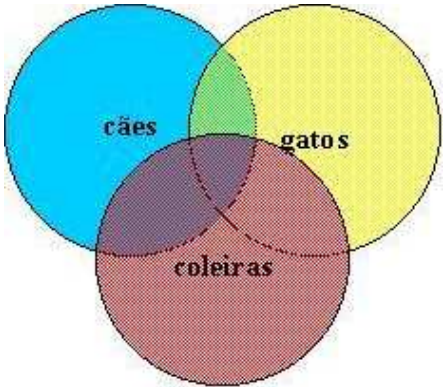
a) se foram localizados registros demais em uma busca, deve-se acrescentar um outro termo de busca, com o operador AND;
b) se foram localizados muitos registros sobre um assunto não selecionado, eliminar uma ou mais palavras com o operador NOT;
c) se foram localizados muito poucos registros em uma busca, pode-se acrescentar um outro termo de busca com o operador OR.

6.2.1 Parênteses

Para aperfeiçoar as buscas, dependendo do sistema, pode-se introduzir um conceito utilizado em Álgebra. Trata-se de parênteses para fazer o computador "entender" o que se pretende obter com a questão de busca. Se quisermos procurar mais de dois termos, isto é, ao utilizarmos AND e OR numa busca, temos de comunicar ao computador, qual a parte da busca a ser executada primeiro. Assim como na álgebra, os ítens dentro de parênteses são sempre interpretados e executados em primeiro lugar.

Exemplo: ao fazer uma busca de artigos sobre o uso de coleiras em cães ou o uso de coleiras em gatos, por meio de expressões "gatos OR cães AND coleiras", possivelmente não vou encontrar os documentos desejados. Quando houver mais de um elemento entre parênteses, a seqüência será da esquerda para a direita. É o chamado "aninhamento" (BARKER, 2004). A maioria dos sistemas de busca interpreta primeiro o operador NOT, sendo seguido pelos operadores AND e OR. Portanto, o que na realidade obtemos pela busca acima são documentos que tratam de cães com coleiras, além de todos os documentos sobre gatos. Isto pode ser ilustrado com diagramas de Venn, em nosso espaço informacional imaginário :

FIGURA 7 – USO DE ... OR ... OR ... [CÃES OR GATOS OR COLEIRAS]



Ao interpretar a expressão gatos OR cães AND coleiras, do mesmo modo como o computador, teríamos de executar primeiramente o operador AND para os círculos correspondentes a cães e coleiras (área colorida). Em seguida, seria aplicado o operador OR à área resultante, com o círculo referente a gatos.

FIGURA 8 – USO DE ... OR ... AND ... [CÃES OR GATOS AND COLEIRAS]

Para resolver este problema, fazemos o computador trabalhar primeiramente com o operador OR, usando a expressão de busca (cães OR gatos) AND coleiras. Disso resultaria o espaço informacional a seguir representado (em amarelo):

FIGURA 9 – USO DE (... OR ...) ...AND ....[(CÃES OR GATOS) AND COLEIRAS]

Mesmo no caso em que se supõe que o computador vai "entender" a busca pretendida, será sempre mais seguro usar parênteses, especialmente quando houver qualquer possibilidade de confusão. Para o próprio usuário o uso de parênteses também ajuda a tornar mais inteligíveis as expressões de busca por ele elaboradas.

Mas em alguns casos não há mesmo necessidade de se utilizar parênteses. Vejamos alguns exemplos (WEB ARCHITECTS, 2004):

a) usando apenas AND: cães AND pêlos AND moscas AND coleiras AND tapetes;
b) usando apenas OR: pulgas OR mosquitos OR carrapatos.

É possível fazer buscas booleanas mais complicadas, com parênteses dentro de parênteses. No caso da busca booleana na *web*, no entanto, isto não é recomendado, pois fazer buscas mais simples é, em geral, melhor do que criar uma busca perfeita, mas complicada. Com a formulação mais simples de uma questão de busca, também é mais fácil fazer alterações para melhorar os resultados. Por outro lado, com uma formulação mais complicada é difícil determinar qual a parte da expressão de busca que encontrou uma determinada página (BARKER, 2004).

Resumo:

a) com o uso de parênteses se possibilita que os termos e as operações dentro dos mesmos sejam processados em primeiro lugar;
b) os parênteses devem ser utilizados para agrupar os termos ligados por OR, quando ainda houver um outro operador na expressão de busca;

6.3 OPERADORES ASSOCIADOS À BUSCA BOOLEANA

Trata-se de recursos desenvolvidos para tornar a busca booleana mais eficiente, sem subordinar-se, porém, à sua lógica subjacente.

6.3.1 Busca por NEAR

No caso do operador AND, os termos no documento encontrado podem constar em qualquer lugar do mesmo. Dentro de um documento longo, muitas palavras poderão gerar combinações que realmente não são objeto do documento. Por exemplo, no caso da busca “vermelho AND celeiros”, um certo documento pode ter “celeiro branco” no primeiro parágrafo e “vagões vermelhos” vinte parágrafos adiante. Assim, vou recuperar este documento, embora não tenha nada a ver com “celeiros vermelhos” ou celeiros que sejam pintados de vermelho, que era o objeto da busca.

Por isso, em lugar do resultado dessa busca quero ter a certeza de recuperar “vermelho NEAR celeiro”. Isto quer dizer, documentos com o termo “vermelho” perto do termo “celeiros”. Com isto, obterei documentos com frases como “Pedro foi até o celeiro vermelho”, mas também “tiramos algumas caixas vermelhas do celeiro”. A tolerância de NEAR varia de um sistema de busca para outro, normalmente de 9 a 15 palavras. Às vezes, esse operador é expresso por WITHIN (dentro de), indicando a distância de um termo de busca até o outro. (WEB ARCHITECTS, 2004)

Resumo:

a) o operador NEAR requer que o respectivo termo esteja dentro de uma certa proximidade em relação à palavra de busca;

b) juntando palavras com NEAR, serão recuperados menos documentos do que utilizar apenas AND, porque NEAR requer que as palavras estejam mais próximas umas das outras;

c) o operador NEAR é utilizado, quando se deseja que determinados termos apareçam no mesmo período gramatical ou parágrafo do documento.

6.3.2 Busca por Truncamento

Entende-se por truncamento a redução da extensão de uma palavra, cujo o resultado pode ou não coincidir com a raiz lexical etimológica.

Existem dois tipos de truncamentos: aberto e fechado. O truncamento aberto permite a substituição de alguns caracteres, possibilitando a recuperação de documentos que contenham termos com a mesma raiz (semântica), por exemplo:

bibliotec*

Neste caso, com o truncamento à direita, serão recuperados: biblioteca, bibliotecas, biblioteconomia, bibliotecário, bibliotecárias, biblioteconômico etc.

No caso do truncamento, à esquerda, como em *metro, o sistema vai recuperar: barômetro, termômetro, manômetro...

No caso do truncamento **fechado** substitui-se um único caractere, por exemplo, no caso de se precisar as duas grafias de:

planejamento (forma brasileira) e

planeamento (forma portuguesa).

Para ter a certeza de recuperar todos os documentos que contem esse contexto na forma brasileira ou portuguesa, faz-se um truncamento fechado: plane?amento, normalmente, assinalando a lacuna (particularmente) por meio de um ponto de interrogação.

6.3.3 Busca por Frase Completa

Na pesquisa por frases, isto é, substantivos compostos ou expressões, pode-se pesquisar utilizando marcas de citação. Com as palavras entre aspas (“deste jeito”) elas aparecerão juntas em todos os documentos retornados. As buscas por frases utilizando aspas são úteis para encontrar substantivos compostos, nomes específicos, provérbios, expressões mais longas (GOOGLE, 2004).

6.4 MODALIDADES DA BUSCA BOOLEANA

Com respeito à Internet, a lógica booleana tem sido utilizada nas modalidades plena, implícita e sob forma de linguagem pré-determinada. Seguimos aqui a explicação de COHEN (2005).

6.4.1 Busca Booleana Plena

São muitas as ferramentas de busca que oferecem a opção da busca booleana plena, exigindo o uso dos respectivos operadores lógicos booleanos.

Exemplos:

a) Questão: Eu preciso de informações sobre gatos.

Lógica de Boole: OR

Busca: gatos OR felinos

b) Questão: Estou interessado em dislexia em pessoas adultas.

Lógica de Boole: AND

Busca: dislexia AND adultos

c) Questão: Estou interessado em radiação, mas não em radiação nuclear.

Lógica de Boole: NOT

Busca: radiação NOT nuclear

d) Questão: Quero me informar sobre o comportamento de gatos.

Lógica de Boole: OR, AND

Busca: (gatos OR felinos) AND comportamento

Neste último exemplo os parênteses servem para reforçar a seqüência do processamento. Colocaram-se entre parênteses as palavras conectadas pelo operador OR, de modo que o sistema vai processar primeiramente os dois termos relacionados. Em seguida, o sistema de busca combinará este resultado com a última parte da busca, que envolve a palavra seguinte. Usando este método, tem-se a certeza de que os termos semanticamente relacionados pelo operador OR estão sendo tratados como unidade lógica única.

6.4.2 Busca Booleana Implícita

A busca booleana "implícita" refere-se a um tipo de busca, no qual são utilizados símbolos para representar os operadores lógicos booleanos. Neste tipo de busca na Internet, mesmo a ausência de um símbolo pode também ser significante. Assim, o espaço entre as palavras é automaticamente direcionado para o operador OR ou então para AND. Muitas ferramentas de busca tradicionalmente eram direcionados à lógica do operador OR. Atualmente, essa prática está sendo abandonada a favor do operador AND (COHEN, 2004).

A Lógica de Boole implícita tornou-se tão comum na Rede que pode ser considerada hoje um padrão de fato.

Exemplos:

a) Questão: Eu preciso de informações sobre gatos.

Lógica de Boole: OR

Busca: gatos + felinos

Este exemplo aplica-se aos sistemas de busca que interpretam o espaço entre as palavras-chave como correspondendo ao OR booleano. Para encontrar qual a lógica que o sistema de busca está usando como padrão, deve-se consultar os arquivos de Ajuda. Atualmente, há poucos sistemas de busca que ainda se utilizam da lógica de OR como forma padrão (COHEN, 2004).

b) Questão: Estou interessado em dislexia em pessoas adultas.

Lógica de Boole: AND

Busca: dislexia & adultos

c) Questão: Estou interessado em radiação, mas não em radiação nuclear.

Lógica de Boole: NOT

Busca: radiação –nuclear

d) Questão: Quero aprender algo sobre o comportamento dos gatos.

Lógica de Boole: OR, AND

Busca: gatos + felinos & comportamento.

6.4.3 Linguagem Pré-determinada

Alguns sistemas de busca oferecem formulários de busca que permitem ao usuário escolher o operador booleano a partir de um *menu*. Normalmente, o operador lógico é expresso de forma descritiva (coloquial), e não pelo respectivo operador booleano.

Exemplos:

a) Questão: Preciso de informações sobre gatos

Lógica de Boole: OR

Busca: qualquer uma dessas palavras/ pode conter as palavras

b) Questão: Estou interessado em dislexia em adultos.

Lógica de Boole: AND

Busca: todas estas palavras/ deve conter as palavras

c) Questão: Estou interessado em radiação, mas não em radiação nuclear.

Lógica de Boole: NOT

Busca: não deve necessariamente conter as palavras/ não deveria conter as palavras

d) Questão: Quero aprender algo sobre o comportamento de gatos.

Lógica de Boole: OR, AND

Busca: combine as opções acima, se o formulário permite expressões de busca múltipla

TABELA 10 – BUSCA BOOLEANA PLENA, BUSCA BOOLEABA IMPLÍCITA E LINGUAGEM PRÉ-DETERMINADA

	Busca Booleana Plena	**Busca Booleana Implícita**	**Linguagem pré-determinada**
OR	Colégio OR universidade	Colégio + Universidade (veja nota abaixo)	- qualquer uma dessas palavras - pode conter as palavras
AND	Pobreza AND crime	Pobreza & crime	- todas as palavras - deve conter as palavras
NOT	Gatos NOT cães	gatos –cães	- não deve conter as palavras - não deveria conter as palavras

FONTE: COHEN, 2005

Nota: esta expressão de busca será resolvida pela lógica do AND em sistemas de busca que utilizarem AND como padrão. Hoje em dia, a maior parte dos sistemas de busca adotou o padrão AND. Entretanto, para ter certeza, o usuário deve consultar "Arquivos de Ajuda" do respectivo *site* para saber qual a lógica adotada como padrão.

6.5 NORMALIZAÇÃO DE QUESTÕES DE BUSCA BOOLEANA

O conteúdo do presente subcapítulo, que não se situa propriamente na perspectiva do usuário comum de busca booleana, apresenta os pressupostos lógicos dessa técnica. Baseia-se na tradução e adaptação abreviada de KORFHAGE (1997).

Ao processar uma questão booleana, o usuário é livre para construir questões de busca bastante complexas. Estes exigem o desenvolvimento de respostas parciais a serem juntadas na resposta final à questão de busca. Toda a questão de busca booleana pode ser reformulada seja sob *forma disjuntiva normal* (FDN), seja sob *forma conjuntiva normal* (FCN). Cada qual oferece uma forma-padrão fácil de processar. Isto pode ser feito automaticamente, sem exigir que o usuário, ele próprio leve a cabo o processo. O valor do FDN ou FCN para uma questão de busca refere-se à respectiva questão de busca original, sendo no final recuperado o mesmo conjunto de documentos.

Em seguida, ambas as modalidades são escritas, seguindo a exposição de KOFHAGE (1997, p. 57-62).

6.5.1 Forma Disjuntiva Normal (FDN)

Há três níveis de expressões em uma forma disjuntiva normal (FDN):

a) termos, que são palavras simples ou compostas que ocorrem ou naturalmente ou sob forma negativa. Por exemplo, concerto e NOT teatro são dois termos válidos;
b) conjunções, que são termos unidos por AND. Por exemplo, jantar AND show AND NOT teatro é um conjunto válido;
c) disjunções, que são conjuntos unidas por OR. Uma questão de busca booleana em FDN é constituída por uma ou mais disjunções. Por exemplo a expressão de busca

(*concerto* AND *jantar* AND NOT *teatro*) OR
(*voleibol* AND *tênis*) OR
(*natação* AND NOT *futebol*)

constitui uma expressão do tipo FDN válida. Uma vantagem da modalidade FDN é que uma questão sob esta forma pode ser dividida em questões menores, cada qual constituída de um dos conjuntos. Assim, a questão no exemplo acima pode ser tratada

sob a forma de três questões separadas. Em seguida, os resultados podem ser juntados para produzir a resposta à questão de busca inicial. Alguns autores insistem que cada conjunto deve conter todos os termos possíveis em uma expressão de busca. Assim, em lugar de

(A AND B) OR (A AND NOT C),

estes autores ampliariam cada termo, resultando na expressão:

(A AND B AND C) OR (A AND B AND NOT C)
OR (A AND NOT B AND NOT C)

A forma completamente expandida é chamada de *forma disjuntiva normal plena*.

6.5.2 Forma Conjuntiva Normal (FCN)

Uma questão de busca do tipo *forma conjuntiva normal* (FCN) é definida de modo semelhante, com os papéis de AND e OR invertidos. Os termos são unidos por OR para formar disjunções, e estes são unidos por AND para formar conjuntos. Uma questão típica nesta modalidade poderia ser:

(*música popular* OR *jantar* OR NOT *teatro*) AND
(*natação* OR *tênis*) AND
(*voleibol* OR NOT *futebol*).

Esta não é uma forma diferente da modalidade FDN citada acima, mas é uma questão de busca totalmente diferente que recuperará um conjunto também totalmente diferente.

6.5.3 Normalização

O processo de transformar uma questão de busca booleana nas modalidades FDN ou FCN é chamado *normalização*. A normalização de uma questão de busca booleana qualquer é feita com auxílio das chamadas tabelas-verdade. Em uma tabela-verdade, cada coluna tem o valor de *verdadeiro* ou *falso*. Ao construir o FDN completo para uma questão de busca, são utilizadas somente as colunas “verdadeiras”

da tabela-verdade. Estas constituem justamente os termos na forma completa normal disjuntiva para a expressão. Como exemplo, considere-se a seguinte questão de busca:

(A OR B) AND (C OR NOT D) AND (D OR B),

expandindo-a para uma tabela-verdade.

A FDN plena para a questão é formada, tomando-se as colunas “verdadeiras” da tabela da verdade e combinando-as com OR:

Coluna 1: A AND B AND C AND D

Coluna 2: A AND B AND C AND (NOT D)

Coluna 4: A AND B AND (NOT C) AND (NOT D)

Coluna 5: A AND (NOT B) AND C AND D

Coluna 9: (NOT A) AND B AND C AND D

Coluna 10: (NOT A) AND B AND C AND (NOT D)

Coluna 12: (NOT A) AND B AND (NOT C) AND (NOT D)

Cada um destes conjuntos pode ser processado separadamente, ou eles podem ser combinados na FDN plena para a questão de busca inicial,

(A AND B AND C AND D)

OR (A AND B AND C AND (NOT D))

OR (A AND B AND (NOT C) AND (NOT D))

OR (A AND (NOT B) AND C AND D)

OR ((NOT A) AND B AND C AND D)

OR ((NOT A) AND B AND C AND (NOT D))

OR ((NOT A) AND B AND (NOT C) AND (NOT D)).

TABELA 11 – TABELA - VERDADE PARA A FORMA DISJUNTIVA NORMAL PLENA

Coluna	A	B	C	D	A OR B	C OR NOT D	D OR B	Expressão
1	V	V	V	V	V	V	V	V
2	V	V	V	F	V	V	V	V
3	V	V	F	V	V	F	V	F
4	V	V	F	F	V	V	V	V
5	V	F	V	V	V	V	V	V
6	V	F	V	F	V	V	F	F
7	V	F	F	V	V	F	V	F
8	V	F	F	F	V	V	F	F
9	F	V	V	V	V	V	V	V
10	F	V	V	F	V	V	V	V
11	F	V	F	V	V	F	V	F
12	F	V	F	F	V	V	V	V
13	F	F	V	V	F	V	V	F
14	F	F	V	F	F	V	F	F
15	F	F	F	V	F	F	V	F
16	F	F	F	F	F	V	F	F

A FDN plena para uma questão envolve, freqüentemente, várias frases (conjuntos) alguns dos quais podem ser combinados. Neste exemplo, visto que as colunas 1 e 2 são ambas incluídas, está claro que para estas colunas não importa aplicar D ou NOT D. Em ambas as colunas, A, B e C são verdadeiras. Por isso, a expressão A AND B AND C é verdadeira para estas duas colunas, mas para nenhuma outra. Esta única expressão cobre ambas as colunas e pode ser substituída para as duas expressões que envolvem D e NOT D. Várias técnicas existem para *minimizar* uma expressão de busca booleana, isto é, reduzi-la à forma mais simples possível. Aplicadas ao presente exemplo, estas técnicas levam à seguinte expressão de FDN mais simples:

(A AND C AND D) OR (B AND C) OR (B AND (NOT D)).

Os três conjuntos cobrem todas as colunas verdadeiras (e nenhuma das falsas) da tabela:

Coluna 1: (A AND C AND D), (B AND C)

Coluna 2: (B AND C), (B AND (NOT D))

Coluna 4: (B AND (NOT D))

Coluna 5: (A AND C AND D)

Coluna 9: (B AND C)

Coluna 10: (B AND C), (B AND (NOT D))

Coluna 12: (B AND (NOT D))

A FCN plena para uma questão de busca pode ser obtida de modo semelhante. Começa formando a FCN plena, usando as colunas falsas da tabela. Esta é claramente a FDN para a negação da questão. Pela negação, e aplicando o Teorema de DeMorgan, ela é convertida para a FCN no lugar da questão original. Pelo Teorema de DeMorgan a negação é movida para os termos individuais, intercambiando AND e OR no processo:

NOT (A AND B) = (NOT A) OR (NOT B),

NOT (A O B) = (NOT A) AND (NOT B).

Neste processo também é usada a *Lei de Negação Dupla*:

NOT (NOT A) = A

Usando um exemplo mais breve, se o FDN para a negação de uma questão de busca for:

(A AND B AND NOT C) OR (NOT A AND C) OR (B AND C),

então o FCN para questão de busca é determinada, negando esta expressão e expandindo-a para:

NOT ((A AND B AND C) OR NOT A AND C) OR (B AND C))

= NOT (A AND B AND NOT C)

AND NOT (NOT A AND C)

AND NOT (B AND C)

= (NOT A OR NOT B OR NOT (NOT C))

AND (NOT (NOT A) OR NOT C)

AND (NOT B OR NOT C).

O resultado final é a expressão FCN:

(NOT A OR NOT B OR C) AND (A OR NOT C) AND (NOT B OR NOT C).

Cada disjunção em um FDN produz um conjunto de respostas para a expressão de busca, que são então juntados para desenvolver um conjunto pleno. Isto não acontece com o FCN. Enquanto cada conjunção produz um conjunto de candidatos em resposta à questão de busca, estes candidatos ainda devem ser validados, satisfazendo outras conjunções. Assim, em nosso exemplo, qualquer documento que satisfaz NOT A OR NOT B OR C, também tem que satisfazer A OR NOT C AND NOT B OR NOT C. Entretanto, recorde-se a forma de FCN como sendo basicamente uma questão conjuntiva:

A AND B AND C AND D,

onde cada uma das conjunções A, B, C e D foi substituída por uma lista de sinônimos ou termos alternativos. O uso de um tesauro, por exemplo, pode expandir facilmente uma determinada questão do tipo FCN para uma questão mais abrangente de busca de FCN. Bibliotecários de referência, familiarizados com os termos usados em uma determinada coleção de literatura, freqüentemente se utilizam de questões de busca do tipo FCN.

Quantitativamente, o processamento de uma questão de busca booleana pode ser reduzido, desde que haja algum conhecimento prévio da base de dados. A começar com um determinado conjunto de documentos, ao processar cada conjunção (AND), reduz-se o tamanho do conjunto, pois eliminam-se aqueles documentos que não satisfazem a conjunção. Assim, o conjunto de documentos recuperados em resposta à questão de busca A AND B deve ser, no melhor dos casos, menor que o número de documentos que contêm A e o número daqueles que contêm B e, provavelmente, será menor que cada um deles. Consequentemente, se os tamanhos dos vários conjuntos satisfazendo os termos em um questão de busca forem conhecidos, o processamento desses conjuntos resultará em ter de lidar com conjuntos cada vez menores.

7 DIFICULDADES E LIMITAÇÕES DA BUSCA BOOLEANA

Apesar da simplicidade e do sucesso da busca booleana, especialmente em sistemas comercializados e na Internet, ela apresenta vários problemas ou dificuldades que são explicitados a seguir, seguindo KORFHAGE (1997).

A **primeira dificuldade** é que numa questão de busca puramente booleana, não há como atribuir pesos aos termos conforme sua importância. Um termo está presente ou ausente. Assim, o usuário tem pouco controle sobre a importância por ele atribuída a um termo numa questão de busca. O usuário de um banco de dados sobre música, por exemplo, não poderia facilmente formular uma busca booleana para "música de Beethoven, preferivelmente uma sonata". As expressões de busca booleana mais simples seriam *Beethoven* AND *sonata* (o que eliminaria qualquer outra música de Beethoven) e *Beethoven* OR *sonata* (que incluirá também sonatas de outros compositores). A expressão de busca (*Beethoven* AND *sonata*) OR *Beethoven* poderia alcançar o resultado desejado, mas a maioria dos sistemas booleanos não distinguiria entre sonatas de Beethoven e outras músicas de *Beethoven*. Em outras palavras, a simples questão de busca com o termo "Beethoven" alcançaria um resultado semelhante. Entretanto, ao longo do tempo desenvolveram-se sistemas booleanos com acréscimo de critérios estatísticos que trabalham com atribuição de pesos aos termos para superar essa limitação.

A **segunda dificuldade** com a busca booleana decorre do fato de que ela pode produzir resultados errôneos por causa de uma questão de busca mal formulada. Este problema envolve a interpretação incorreta dos conectivos booleanos AND e OR. Pessoas não bem familiarizadas com as convenções lógicas, e tendo usado estes conectivos só informalmente de modo intuitivo, tendem a utilizá-los erroneamente aqui. O uso das expressões que contenham os operadores AND e OR em lógica booleana não corresponde ao uso intuitivo, cotidiano. Por exemplo, uma pessoa que pesquisa as possibilidades de diversão para sábado à noite especificou seus interesses na expressão de busca:

jantar AND *futebol* AND *música popular*

A escolha de eventos em que haja simultaneamente *jantar* e *futebol* e *música popular* não se aplica; provavelmente a pessoa queria expressar:

jantar OR *futebol* OR *música popular*,

ou ainda

jantar AND (*futebol* OR *música popular*).

O sistema de recuperação booleana por si só não pode proporcionar uma resposta satisfatória a uma questão desse tipo. Isto traz uma dificuldade para o usuário não iniciado em lógica de Boole. Por essa razão, as ferramentas utilizadas hoje na Internet acabaram por eliminar esse problema com a busca implícita.

A **terceira dificuldade** com sistemas de recuperação booleana consiste na ordem de precedência dos conectivos lógicos. São permitidos dois padrões diferentes para a ordem de precedência. Ambos se utilizam de parênteses para agrupar os termos: a combinação dentro de parênteses é processada como uma unidade, antes de ser combinação com os termos fora dos parênteses. Em uma das modalidades, NOT é aplicado primeiro dentro dos parênteses, seguido por AND, que é seguido por OR. Neste uso, há uma precedência da esquerda para direita para os respectivos operadores do mesmo tipo. Mas em outros sistemas segue-se uma ordem de precedência da esquerda para a direita, sem levar em conta os operadores. Assim, a questão de busca:

A OR B AND C

seria interpretada como

A OR (B AND C)

no primeiro tipo de sistema, mas como

(A OR B) AND C

no segundo tipo de sistema. Em qualquer um dos casos, é preciso utilizar parênteses, quando se pretende obter uma interpretação diferente daquela que o sistema assume automaticamente.

Note-se que neste exemplo, a primeira interpretação poderá recuperar um documento que contém somente o termo A. Já a segunda interpretação não, pois exige a presença do termo C. Observou-se que as pessoas tendem a interpretar esta ambigüidade de uma maneira ou de outra, dependendo das relações semânticas entre os três termos. Por exemplo, as pessoas interpretam a expressão de busca "*café* AND *pão de queijo* OR *rosca*" diferentemente da expressão de busca "*capa de chuva* AND *guarda-chuva* OR *óculos*". No segundo uso, evidentemente, a pessoa certamente não escolherá entre guarda-chuva ou óculos.

O operador NOT pode causar um outro problema, embora de mais fácil solução. Viu-se que NOT recupera todos os documentos que não contenham um determinado termo específico. Desse modo, uma questão de busca com NOT, vinculado a uma palavra inexistente, corre o risco de ter de recuperar virtualmente a base de dados inteira. Uma maneira de solucionar isto é restringir o uso de NOT a situações, em que este operador se aplica apenas a uma quantidade pequena de documentos. Por exemplo, suponha-se que a questão de busca seja:

(NOT A) AND B AND C.

Em lugar de começar a interpretação da questão de busca com NOT A, que vai recuperar todo documento que não contém o termo A, começa-se com B AND C. Então, a condição NOT A é aplicada a um conjunto muito menor de documentos resultantes, isto é só aqueles que contêm ambos B e C.

A imensa maioria dos usuários de sistemas de informação e, em especial de Internet, naturalmente, não é treinada em Álgebra de Boole. O problema da aprendizagem de interpretar corretamente os operadores booleanos e suas regras de precedência, combinado com o fato de que muitos usuários não têm acesso a sistemas de recuperação de informação regularmente, tem sido a principal barreira para o uso efetivo de sistemas de recuperação booleanos.

A **quarta dificuldade** em sistemas de recuperação booleana está em controlar a extensão e composição dos conjuntos recuperados de documentos. Tecnicamente, o sistema deveria apresentar todos os documentos que satisfazem a

questão de busca. Porém, da busca pode resultar um número muito reduzido ou muito grande. No segundo caso, o sistema pode apresentar ao usuário várias centenas ou milhares de documentos para exame, sem nenhuma ordem específica. Uma solução seria começar novamente com uma questão de busca mais restrita. O usuário pode em seguida formular uma nova questão de busca, ou imediatamente, ou após examinar os primeiros documentos recuperados. Poderá acrescentar mais termos à questão original ou construir uma questão de busca com um conjunto de termos completamente novo. Há sistemas que restringem automaticamente o número de documentos apresentados ao usuário. Às vezes, este número pode ser fixado pelo sistema ou pelo usuário. Mas, a menos que o sistema faça uma ordenação dos documentos pelo número de termos de busca encontrados, pode acontecer que o corte arbitrário resulta na retenção de documentos relativamente fracos, enquanto descarta os melhores encontrados.

Quando os documentos são ordenados pela quantidade de termos encontrados, tem-se inicialmente uma ordenação mais ou menos grosseira. Isto pode ser melhorado em um sistema de busca booleana, utilizando, por exemplo, um tesauro para ampliar a quantidade dos termos de busca. Entretanto, com isto não se resolve satisfatoriamente o problema da importância relativa dos termos individuais na questão de busca.

Por último, existe ainda uma dificuldade de natureza estrutural com as questões de busca booleana que KORFHAGE (1997, p. 62) chama de “problema filosófico”. Ao executar uma recuperação, o sistema compara um documento com uma questão. Um documento pode ser representado facilmente por uma lista de termos que ele contém. Mas por tratar-se de uma lista, e não de uma expressão booleana, a introdução de conectivos booleanos numa questão de busca parece produzir uma forma de busca menos parecida com o documento do que seria uma lista de termos. Por causa desta diferença estrutural entre a questão de busca e o documento, pode-se encarar a recuperação booleana antes como um processo de mapeamento e não propriamente como um processo de verificação de coincidências (*matching process*) entre termos de busca e termos contidos nos documentos.

Apesar dessas restrições, os sistemas de recuperação booleana continuam bastante difundidos e razoavelmente eficientes. Sua facilidade de uso certamente contribui para isto. O usuário especifica uma lista de termos combinados com AND, OR e NOT, talvez incrementada por operadores de proximidade ou por outros operadores. Não há necessidade de considerar a possível importância que determinado termo possa ter. As dificuldades de modelar a questão de busca com precisão lógica, não tem causado maiores impactos práticos. As pessoas tendem a usar apenas dois ou três termos de cada vez, evitando construir questões de busca complexas. Finalmente, outros modelos de recuperação, embora teoricamente mais interessantes, não têm alcançado, na prática, resultados superiores àqueles dos sistemas booleanos. Se isto permanecerá assim, enquanto os sistemas de recuperação estão migrando para sistemas de texto inteiro e documentos multimídia, ainda é difícil de se prever (KORFHAGE, 1997, p. 62-63).

O que se tem conseguido nos últimos anos, especialmente na Internet, foi implementar mecanismos de busca “híbridos”, com a implementação inclusive de métodos estatísticos, tornando o processo de recuperação mais “amigável” para os usuários em geral. Na mesma tendência, a própria busca booleana tem-se tornado cada vez mais “implícita”, dispensando, na maioria dos casos, o conhecimento técnico sobre seu mecanismo.

8 ESTRATÉGIAS DE BUSCA

Estratégia de busca é o meio pelo qual o pesquisador se comunica com o sistema, e é muitas vezes a chave para uma busca bem sucedida. As técnicas de busca são métodos, táticas, estratégias ou planos que podem ser usados nas buscas em sistemas de informação convencionais ou eletrônicos.

Portanto, estratégia de busca é o procedimento por meio do qual se procuram documentos ou informações sobre um determinado assunto. Pressupõe sempre a existência de um sistema de recuperação da informação, consistindo um elemento essencial a função de *output* (saída) a informação nesse sistema. Para que se possa recuperar uma informação, é preciso que haja uma informação anteriormente armazenada, e é preciso que haja uma perfeita identidade entre a linguagem adotada pelo sistema na entrada dos dados e aquela utilizada durante a busca (saída dos dados). Segundo ROGERS (1980, p. 72) o processo de busca inclui os seguintes passos:

1. esclarecer a necessidade de informação;
2. estabelecer os parâmetros de busca baseados na necessidade;
3. identificar o sistema(s) onde será feita a busca;
4. traduzir (indexar) a necessidade para a linguagem do sistema;
5. realizar a busca;
6. fornecer a informação.

Em outras palavras, isso significa que é preciso, em primeiro lugar, estabelecer claramente qual a necessidade de informação do cliente/usuário que deverá ser atendida, procurando-se, portanto, aproximar ao máximo possível a demanda expressa da necessidade a ser identificada. No segundo passo, o pedido do cliente é analisado, procurando-se reconhecer as várias facetas dos conceitos envolvidos. Como terceiro passo, é preciso identificar qual ou quais sistemas de recuperação da informação são mais indicados para efetuar a busca da informação solicitada.

Depois da análise conceitual da demanda de informação e da escolha do sistema de recuperação mais adequado à busca pretendida, é preciso traduzir os termos resultantes dessa análise conceitual para a linguagem adotada pelo sistema escolhido. Portanto, há dois passos distintos na elaboração de uma estratégia de busca:

- A análise conceitual daquilo que é realmente desejado, e
- A tradução desta análise para conjunto de termos usados para representar os conceitos procurados em determinada base de dados a ser utilizada.

Esses passos são semelhantes ao processo de indexação dos documentos que entram numa base de dados.

As estratégias de busca podem ser bastante simples ou muito complexas, variando conforme o tipo de demanda, os recursos de busca oferecidos, o grau de sofisticação da indexação e a armazenagem dos dados no respectivo sistema de recuperação da informação. Entretanto, em qualquer tipo de sistema (bases de dados, Internet) as estratégias são basicamente as mesmas.

A busca que utiliza linguagem documentária pela qual o documento foi previamente indexado é bem mais precisa do que aquela que se utiliza da linguagem natural, pois pode-se ter certeza da existência ou não de assuntos procurados. Neste caso, pode-se utilizar um único descritor, que pode resultar num número grande e indesejável de recuperações, ou então utilizar combinações de descritores, segundo a lógica booleana, que permite delimitar e tornam mais precisa uma busca por assunto. Não há, geralmente, restrições para o número de termos e operadores lógicos incluídos numa estratégia de busca. Muitos outros recursos existem, sendo que alguns podem ser utilizados tanto nas buscas por vocabulário controlado como naquelas por linguagem natural.

O grau de sucesso nas buscas de informação depende igualmente do processo de indexação dos documentos, da linguagem de indexação, da interface usuário-sistema e das estratégias de busca empregadas. Consequentemente, caso sejam detectadas falhas na recuperação da informação, é importante avaliar qual desses fatores deve ser responsabilizado e corrigi-lo, se possível. Muitas vezes, a eficiência do sistema de recuperação da informação pode ser imediatamente aumentada através de uma melhor comunicação com os usuários e aperfeiçoamento nas estratégias das busca utilizadas.

Os usuários, obviamente, esperam que o sistema seja capaz de recuperar a quantidade e qualidade de documentos que contribuem para satisfazer alguma necessidade de informação (documentos relevantes).

9 ESTUDO DE CASOS

No presente capitulo é apresentado o resultado de uma pesquisa temática em língua portuguesa realizada em cinco sistemas de busca da Internet. Será demonstrada a capacidade de cada sistema na recuperação da informação desejada, utilizando operadores booleanos e refinando ao máximo a busca.

Uma análise prévia do assunto que se pretende pesquisar permitiu determinar, quais os termos mais adequados e os operadores indicados para associar esses termos. Caso o resultado pretendido não fosse alcançado, seriam acrescentados sinônimos com o operador OR, ou mais um termo referente ao assunto com o operador AND, ou ainda eliminados os termos sem interesse, por meio do operador NOT.

O objetivo pretendido com a presente questão de busca, utilizando os operadores lógicos booleanos, é demonstrar as características dos respectivos sistemas e como são utilizados os operadores nos mesmos. A pergunta apresentada é a seguinte: "Desejo obter informações detalhadas sobre as civilizações da Mesopotâmia".

a) Alta Vista (www.altavista.com.br)

- Características: aceita o uso de todos os operadores
- Questão de busca: Mesopotâmia AND (Caldeus OR Assírios OR Babilônios OR Sumérios) AND política AND cultura AND economia AND religião AND escrita NOT Iraque
- Documentos recuperados: 53

b) Google (www.google.com.br)

- Características: AND = *default*
  OR = "+"
  NOT = "–"
- Questão de busca: Mesopotâmia (Caldeus + Assírios + Babilônios + Sumérios) política cultura economia religião escrita -Iraque
- Documentos recuperados: 3

c) Cadê? (www.cade.com.br)

- Características: aceita todos os operadores booleanos
- Questão de busca: Mesopotâmia AND (Caldeus OR Assírios OR Babilônios OR Sumérios) AND política AND cultura AND economia AND religião AND escrita NOT Iraque
- Documentos recuperados: 56

d) Aonde? (www.aonde.com.br)

- Características: AND = &
  OR = |
  NOT = ~
- Questão de busca: Mesopotâmia & (Caldeus | Assírios | Babilônios | Sumérios) & política & cultura & economia & religião & escrita ~Iraque
- Documentos recuperados: site com problemas

e) Radar Uol (www.radaruol.com.br)

- Características: AND = &
  OR = |
  NOT = ~
- Questão de busca: Mesopotâmia & (Caldeus | Assírios | Babilônios | Sumérios) & política & cultura & economia & religião & escrita ~Iraque
- Documentos recuperados: 4

Constatou-se nesta pesquisa temática que a forma de aplicação dos operadores booleanos e a quantidade de resultados variam conforme o *site.* Constatou-se, também, que todos os itens recuperados pelos sistemas de busca corresponderam ao que se esperava como resposta à pergunta inicial.

10 CONCLUSÃO

Foram apresentadas neste trabalho as modalidades básicas de uso da busca booleana, a qual é utilizada hoje pela maioria dos sistemas de recuperação de informação da Internet. Foram descritos alguns aspectos problemáticos, além da aplicação dessa ferramenta e avaliado seu uso em cinco sistemas de busca.

Na tentativa de se obter conhecimentos básicos sobre a matéria, foi realizada, primeiramente, uma pesquisa bibliográfica acerca do assunto, o qual foi apresentado, numa linguagem de fácil entendimento para usuários iniciantes de sistemas de busca que utilizam os operadores lógicos booleanos.

Um dos problemas referentes ao uso do modelo booleano é que o usuário leigo tem dificuldade em colocar suas necessidades informacionais na forma de uma expressão de busca. O desconhecimento dos rudimentos da Álgebra Booleana acentua as dificuldades. O usuário precisa memorizar símbolos de conetivos e verificar prioridades no uso dos operadores e de parênteses. Como conseqüência, os resultados, muitas vezes, não são satisfatórios.

É grande a variedade de consultas que podem ser feitas com a busca booleana. Entretanto, o sistema, em muitos casos, pode oferecer respostas não relevantes, quando não houve uma estratégia de busca adequada na consulta do usuário. Além disso, a avaliação é muito sensível, sendo que documentos irrelevantes recuperados ou documentos relevantes não recuperados afetam a precisão e a cobertura do resultado.

Apesar disso, grandes volumes de informação, hoje acessíveis na Internet, são disponibilizados por intermédio de ferramentas baseadas na busca booleana.

Pôde-se confirmar, no decorrer deste trabalho, que há ainda poucos textos em português sobre este assunto. Para futuros trabalhos, sugere-se ampliar a descrição sobre arquitetura, processos, técnicas e ferramentas baseadas na Lógica de Boole, bem como elaborar um conjunto de exercícios progressivos para Gestores da Informação.

REFERÊNCIAS

BARKER, Joe. **Boolean searching for the web.** Disponível em: <http://www.lib.berkeley.edu/TeachingLib/Guides/Internet/Boolean.pdf> acesso em: 25 ago. 2004.

BRANSKI, R. M. **Localização de informação na Internet**: características e formas de funcionamento dos mecanismos de busca. Disponível em: <http://www.eco.unicamp.br/cefi/localizacao.doc> acesso em: 04 abr. 2005.

CARDOSO, S. H.; SABATINI, R. M. E. **Pesquisando na Internet**: como usar os recursos avançados, 1998. Disponível em: <http://www.nib.unicamp.br/metodologia/biblio.htm> acesso em: 29 set. 2004

COHEN, Laura. **Boolean Searching on the Internet**. Disponível em: <http://library.albany.edu/internet/boolean.html> acesso em: 24 mar. 2005.

DAGLIAN, J. **Lógica e álgebra de Boole**. São Paulo: Atlas, 1986. 155p.

ENCICLOPEDIA MIRADOR INTERNACIONAL. São Paulo: Britannica do Brasil, 1991.

FERNEDA, E. **Recuperação da informação**: análise sobre a contribuição da Ciência da Computação para a Ciência da Informação. São Paulo, 2003. 147f. Tese (Documento em Ciência da Comunicação) – Escola de Comunicação e Arte, Universidade de São Paulo.

GILSTER, P. **Como encontrar informações na internet**. São Paulo: Makron Books, 1995.

GOOGLE. **Refinando sua busca**. Disponível em: <http://www.google.com.br/intl/pt-BR/help/refinesearch.html> acesso em: 27 set. 2004.

HELP CENTRAL ROBO. **Boolean search**. Disponível em: <http://www.abebooks.com/docs/HelpCentral/RoboHelp/bookbuyerhelp/Searching_and_Browsing_for_Books/Boolean_Search.html> acesso em: 15 set. 2004

KORFHAGE, R. R. **Information storage and retrieval.** New York: Wiley Computer Publishing, 1997, 349p.

LAQUEY, T; RYER, J. C. **O manual da internet**: um guia introdutório para acesso às redes globais. Rio de Janeiro: Campus, 1994.

LOPES, I. L. Estratégia de busca na recuperação da informação: revisão da literatura. **Ciência da Informação**, Brasília, v. 3, n. 2, p. 60-71, mai./ago. 2002

MARBÁN, R. M. **Operadores booleanos en la recuperación de información**. Guatemala: OEA-ICAITI, 1997, 43p.

MOREIRO GONZÁLEZ, J. A. **Manual de documentación informativa**. Madrid: Signo e Imagen, 2000. p.458.

MOURA, G. A. C. **Sistemas de busca da web**: diretórios e mecanismos de busca. Disponível em: <http://www.quatrocantos.com/tec_web/sist_busca/sb_sum.htm> Acesso em: 31 jan. 2004.

RICH, Linda. **Boolean Operators.** Disponível em: <http://www.bgsu.edu/colleges/library/infosrv/lue/boolean.html> acesso em: 24 jan. 2005.

ROGERS, S. J. Research strategies: bibliographic instruction for undergraduates. **Library Trends**, Urbana, v. 29, n. 1, p. 69-80, 1990.

ROWLEY, J. **Informática para bibliotecas**. Brasília: Briquet de Lemos, 1993.

SOMATEMATICA. **Biografias:** George Boole. Disponível em: <http://www.somatematica.com.br/biograf/boole.php> acesso em: 03 jun. 2005.

SULLIVAN, Danny. **Boolean Searching.** Disponível em: <http://searchenginewatch.com/facts/article.php/2155991> acesso em: 24 jun. 2004.

TAUB, Herbert. **Circuitos digitais e microprocessadores**. São Paulo: McGraw-Hill do Brasil, 1984, p. 509.

TRILLO, C. D. P. **Recuperação de vídeos indexados por conceitos.** São Paulo, 2005. 98f. Dissertação (Mestrado em Ciência da Computação) – Instituto de Matemática e Estatísitca, Universidade de São Paulo.

WEB ARCHITECTS. **Boolean tutorial.** Disponível em: <http://florin.syr.edu/webarch/searchpro/boolean_tutorial.html> acesso em: 24 jun. 2004.

APÊNDICES

APÊNDICE 1 – BIOGRAFIA DE GEORGE BOOLE (1815-1864)

APÊNDICE 1 - BIOGRAFIA DE GEORGE BOOLE (1815-1864)

George Boole nasceu em Lincoln, na Inglaterra, a 2 de novembro de 1815 e faleceu de pneumonia em Cork, na Irlanda, a 8 de dezembro de 1864. Casou-se em 1855 com Mary Everest, com quem teve cinco filhas.

Boole, cujo pai era sapateiro, não tinha condições de estudar em um colégio mais afamado. Estudou na Escola Primária Lincoln, e mais tarde, numa Escola Comercial. Dedicou-se, como era usual, ao aprendizado do grego e do latim, tomando aulas particulares com um livreiro local.

Dos 16 aos 34 anos, ensinou em escolas elementares, dirigindo, por vários anos, a que ele próprio fundou. Estudou Matemática por conta própria, sem nenhuma formação acadêmica, e foi até encorajado a estudar na Universidade de Cambrigde. Contudo, não pôde aceitar, pois seus pais necessitavam de sua ajuda.

Em 1840, foi eleito para ocupar o lugar de professor de Matemática no Queen´s College em Cork (Irlanda), onde permaneceu o resto da vida. Em 1844, lançou um trabalho sobre a aplicação de métodos algébricos na solução de equações diferenciais, recebendo uma medalha de Ouro da Royal Society. Publicou a “Analise da Matemática Lógica” em 1847, inaugurando sua carreira como um dos iniciadores da moderna Lógica Simbólica.

Em sua investigação sobre as “Leis do Pensamento” (1854) estão cimentadas as Teorias da Lógica e das Probabilidades. Tornou-se conhecido até hoje pela “Álgebra de Boole”, onde abordou a Lógica de forma a reduzi-la a uma Álgebra simples, inserindo a Lógica na Matemática.

Boole recebeu títulos das Universidades de Dublin e Oxford e, em 1857, foi eleito membro da Royal Society.

A Álgebra de Boole, embora existindo há mais de cem anos, não teve qualquer utilização prática até 1937, quando foi feita sua primeira aplicação na análise de circuitos réles. Atualmente, é utilizada em computadores digitais, tendo-se tornado a base convencional de busca na maioria dos sistemas computadorizados. *

* Compilado a partir das seguintes fontes:
ENCICLOPÉIA MIRADOR INTERNACIONAL, 1991, p. 1473
DAGHLIAN, 1986, p. 18
SOMATEMATICA, 2005

APÊNDICE 2 – SISTEMAS DE BUSCA E METAPESQUISADORES

SUMÁRIO

1 SISTEMAS DE BUSCA

Acrescentou-se o presente Apêndice como complemento ao tema desta monografia. As fontes utilizadas foram, principalmente, MOURA (2004) e BRANSKI (2005).

Entende-se por sistema de busca um conjunto organizado, constituído de computadores, índices, bases de dados e algoritmos – tudo isso reunido com a missão de:

a) analisar e indexar as páginas da *web*, e

b) armazenar os resultados dessa análise e indexação numa base de dados.

Quando de uma consulta de um usuário, o sistema de busca vai pesquisar a(s) sua(s) base(s) de dados e fornecer os resultados da pesquisa ao usuário.

Todas essas funções realizam-se em um *site* da *web* cuja página de abertura é, geralmente, um portal.

Existem duas classes de sistemas de busca: os **diretórios** e os **mecanismos de busca**, sendo que ambos têm a mesma finalidade. Do ponto de vista do usuário-consulente-internauta ambos possibilitam a localização de *site*s e páginas (*homepages*) que contêm determinado assunto ou abordam determinado aspecto de um assunto. A partir do ponto de vista do proprietário-dono-autor de uma página, esses sistemas fazem com que o seu *site* seja localizado, da maneira mais fácil possível, pelo próprio usuário-consulente-internauta. Este pode ser eventualmente um consumidor do próprio produto divulgado no *site*.

As denominações **diretório** e **mecanismo de busca** ainda não estão bem consolidadas no Brasil, havendo diversos termos em uso para designar os mesmos conceitos, por exemplo: sistema de busca, ferramenta de busca, ferramenta de procura, motor de busca, motor de procura, indexador, catálogo, site de busca, programa de busca, serviço de busca, engenho de busca etc.

1.1 MECANISMOS DE BUSCA

Entende-se por **mecanismos de busca** sistemas (de busca) baseados no uso exclusivo de programas de computador para a indexação das páginas da *web*. De uma forma simplificada, esses mecanismos de busca apresentam três componentes principais:

a) um programa de computador denominado robô que “visita” os *sites* ou páginas armazenadas na *web*. Ao chegar num *site*, o programa robô "pára" em cada página do mesmo, criando uma cópia ou réplica do texto contido na página visitada. Essa cópia ou réplica vai compor a sua base de dados;

b) o segundo componente é a **base de dados** constituída pelas cópias obtidas pelo robô. Às vezes também denominada de índice ou catálogo, o resultado dessa busca fica armazenada no computador. Este que também é chamado de servidor do mecanismo de busca;

c) o terceiro componente é o programa de busca propriamente dito, acionado toda vez que alguém realiza uma pesquisa. Nesse instante, o programa sai percorrendo a base de dados do mecanismo em busca dos endereços - os URLS (*Uniform Resource Locators*) - das páginas que contêm as palavras, expressões ou frases informadas na consulta. Em seguida, os endereços encontrados são apresentados ao usuário.

1.2 DIRETÓRIOS DE BUSCA

Diretórios são sistemas de busca, nos quais a indexação das páginas da *web* é realizada por humanos.

Ao realizar uma pesquisa, quer através de um mecanismo de busca quer através de um diretório, não se está pesquisando diretamente a *web*. Está-se pesquisando uma base de dados localizada num *site* da *web*. Nessa base de dados,

encontra-se uma cópia dos *sites* e páginas existentes na *web*. Esse diretório tem dois componentes principais a saber, uma base de dados, também chamada índice ou catálogo, e um programa de computador que faz a pesquisa na base de dados.

Portanto, a montagem ou criação da base de dados de um diretório é realizada por humanos, que fazem a análise e indexação dos *sites* da *web*. Dessa forma, nos diretórios não existem robôs para a catalogação e a indexação da *web*.

Enquanto os mecanismos de busca copiam todo o conteúdo das páginas que encontram pela frente, mantendo tudo em suas bases de dados, nos diretórios mantêm-se apenas resumos do conteúdo dos *sites* catalogados. Muitas vezes, esse resumo que fica na base de dados do diretório, contém apenas o título do *site* mais duas ou três frases sobre o assunto nele contido. Esse resumo pode ser elaborado pelo autor da página ou por um editor, dependendo do diretório.

O **diretório** tem a mesma finalidade do mecanismo de busca: indexação e recuperação de páginas da *web*. Eles têm a mesma finalidade, apesar das diferenças fundamentais entre eles.

Nas ferramentas de pesquisa mais comuns submete-se as palavras a um único banco de dados, recebendo-se uma relação dos documentos onde constam as palavras pesquisadas. Os resultados obtidos em diferentes pesquisadores podem variar bastante, mas também podem conter resultados duplicados.

2 METAPESQUISADORES

Os metapesquisadores buscam, simultaneamente, informações em vários mecanismos de busca. Não possuem banco de dados próprio, funcionando como um agente intermediário que repassa a pesquisa. Obtêm as respostas dos pesquisadores individualmente e, então, apresentam um resultado unificado, extraído das diversas fontes. Em poucos segundos os metapesquisadores compilam os resultados obtidos, economizando tempo e fornecendo uma visão geral do tipo de documentos armazenados em cada ferramenta.

A utilização dos metapesquisadores não elimina a necessidade de se conhecer as características individuais dos diversos mecanismos de busca. Quanto mais se conhece sobre as formas de funcionamento das ferramentas que os alimentam, melhor será a capacidade de avaliar a confiabilidade dos resultados obtidos Se, por exemplo, a pesquisa exige determinados refinamentos não processáveis pelas ferramentas que constituem o metapesquisador, pode haver resultados inadequados ou mesmo erros.