Arquitetura de um Mecanismo para Busca Especializada na Web

Fernando Renier Gibotti
Gilberto Câmara
Instituto Nacional de Pesquisas Espaciais
{gibotti, gilberto}@dpi.inpe.br

Resumo

O desenvolvimento acelerado da Internet e o aumento de conteúdos digitais disponíveis conduziram o desenvolvimento de mecanismos de busca que facilitassem a recuperação de informações na Web. Entretanto, estes mecanismos apresentam limitações principalmente quando se trata da recuperação de conteúdos especializados. Dados geográficos não são encontrados na Web por mecanismos convencionais, pois tais mecanismos não estão preparados para evidenciar este conteúdo. Neste contexto, este trabalho apresenta uma arquitetura para acesso e recuperação de dados geográficos na Web, discorrendo sobre suas principais características, arquitetura, tecnologias envolvidas e performance.

Palavras-chave: mecanismo de busca, dados geográficos, web.

1. Introdução

Desde o seu aparecimento no final da década de 60, a Internet cresceu rapidamente [Howe 1996] e, com sua efetiva utilização, passou de um projeto de pesquisa a uma vasta coleção de documentos heterogêneos. A web é uma rede, cuja arquitetura não foi projetada, composta por bilhões de páginas multimídia interligadas (imagens, sons, texto, animações etc) desenvolvidas de forma descoordenada por milhares de pessoas. Nos últimos anos o aumento expressivo na quantidade de informações publicadas e a inexistência de estruturação do conteúdo de documentos disponibilizados contribuíram para dificultar o processo de recuperação de dados.

Assim, mecanismos de busca como Google (www.google.com), Altavista (www.altavista.com), Yahoo (www.yahoo.com) surgiram com o objetivo de melhorar a procura e análise de informações na web [Glover 2002]. Entretanto, estes mecanismos possuem algumas limitações, que variam da cobertura dos documentos indexáveis na web [Lawrence and Giles 1998] à quantidade de respostas indesejadas retornadas por uma pesquisa, dificultando o acesso à informação.

Para atender às necessidades da comunidade Web frente a essas limitações, surgiram os motores de metabusca que proporcionam maior cobertura da Web, tais como o Inquirus do NEC Research Institute [Lawrence and Giles 1999]. Técnicas de metabusca podem melhorar a eficiência de buscas na Web combinando os resultados de múltiplos mecanismos de busca e implementando novas funcionalidades, tais como extração dos termos consultados no contexto dos documentos e filtragem de links quebrados; porém estes motores apresentam sérias limitações quanto a classificação (ranking) da lista de resultados e limitação no número de resultados que podem ser obtidos, além de não possuírem suporte a todas características das linguagens de consulta de cada mecanismo.

Todos esses esforços para a criação de mecanismos e métodos para a recuperação de informações na Web, embora satisfatórios a uma geração de usuários, ainda trazem limitações quando o conteúdo pesquisado trata-se de dados geográficos. A WEB possui grande quantidade de dados geográficos, porém as ferramentas de busca mais poderosas não são especializadas para recuperá-los, criando uma grande lacuna entre estes dados e os usuários que deles necessitam.

Muitos trabalhos relacionados à recuperação de dados geográficos disponíveis na *Web* propõem criar uma estrutura para que clientes possam depositar os dados de forma estruturada em repositórios e desenvolver ferramentas específicas para a recuperação destes dados. Porém para maximizar o poder da busca, é interessante utilizar a atual estrutura da *Web* para recuperar informações.

Este artigo apresenta um mecanismo de busca especializado para acesso e recuperação de dados geográficos na *Web*. Nossa proposta inclui uma estrutura rápida e distribuída de rastreamento, algoritmos eficientes para analisar e recuperar informações de interesse, utilização otimizada de espaço de armazenamento dos dados e seus índices e uma ferramenta para fornecer fácil acesso aos dados indexados.

2. Recuperação de dados geográficos

Mecanismos de busca tradicionais não possuem rastejadores e analisadores especializados em dados geográficos. Estes são preparados para buscar documentos hipertexto e *hyperlinks*. Esta seção discorrerá sobre os principais aspectos observados para a implementação de nosso mecanismo.

2.1. Encontrando dados geográficos

A primeira questão é como encontrar dados geográficos. Nossa escolha foi considerar que dados geográficos estão normalmente distribuídos em um conjunto de formatos predefinidos associados a sistemas GIS. Podemos citar como exemplo o formato *shapefile* criado pelo software *ArcView* que é uma forma de troca de dados geográficos bem aceita pela comunidade *GIS*. Desta forma, nosso mecanismo tenta recuperar arquivos *GIS* mais comumente utilizados. Em nosso protótipo atual, são recuperados arquivos *shapefile*, mas o mecanismo pode ser facilmente estendido para recuperar outros formatos *GIS*.

2.2. Dados sobrepostos

A existência de cópias de dados geográficos em diferentes sítios da *Web* gera um problema para a indexação dos arquivos pois pode introduzir duplicações na base de dados. A detecção de dados similares é possível pela análise de sua estrutura e de alguns atributos intrínsecos dos arquivos, tais como nome, data da criação, tamanha e tipo. Através destes atributos intrínsecos, nosso mecanismo identifica arquivos similares e evita o armazenamento redundante.

2.3. Classificação dos produtores de dados

Mecanismos de busca tradicionais utilizam diferentes formas para classificar os principais sítios da *Web*. O mecanismo *Google*, utiliza o método de *PageRank* para priorizar os resultados de busca por palavras-chave [Page, Brin, Motwani et al. 1999] [Gerhart 2002]. Similarmente ao método para classificar páginas utilizando a informação dos links, o mecanismo *Citeseer* [Giles, Bollacker and Lawrence 1998] classifica artigos científicos como *hubs* e autoridades baseado no gráfico de citações.

O objetivo de um mecanismo de busca é possibilitar às pessoas recuperar dados geográficos de forma segura quanto à qualidade e origem dos dados. O mecanismo desenvolvido identifica produtores de dados baseado em três aspectos: quantidade de dados geográficos disponível no sítio da *Web*; quantidade de arquivos capturados pelos usuários a partir dos arquivos disponíveis no sítio; e pela indicação se o sítio é um *hub* ou uma autoridade. Nesta abordagem, *hubs* são sítios que recomendam outros sítios que contenham dados geográficos e autoridades são sítios que são recomendados por muitos *hubs*.

Ao retornar uma lista de dados geográficos para o usuário, o mecanismo desenvolvido apresenta, para cada dado, seu produtor (por exemplo, IBGE), a quantidade de dados geográficos produzidos por este produtor (por exemplo, 1080 *shapefiles*) e a quantidade de *downloads* executados a partir do sítio deste produtor (por exemplo, 10.112 *shapefiles* requisitados pelos usuários).

3. Arquitetura do sistema

O mecanismo de busca desenvolvido tem algumas características que auxiliam no processo de busca de dados geográficos e que melhoram significativamente os resultados retornados. Primeiro ele classifica os produtores de dados geográficos. Segundo, ele utiliza informações adicionais presentes nos *hyperlinks* para descrever o conteúdo dos dados geográficos e finalmente ele identifica dados sobrepostos disponíveis em diferentes sítios da *Web*.

3.1. Texto âncora e texto âncora estendido

Normalmente os atributos dos arquivos (nome, data da criação e última modificação, tamanho e tipo) não oferecem descrições detalhadas do conteúdo do arquivo. A falta de informações adicionais torna a indexação destes arquivos uma tarefa difícil e algumas vezes os resultados obtidos não são os desejáveis.

Um sítio da *Web* pode ser composto por multimídias tais como sons, imagens, textos, arquivos e pelas conexões a outros sítios ou páginas, os *hyperlinks*. A estrutura criada por estas conexões está sendo pesquisada e utilizada para melhorar a eficiência dos rastejadores [Cho, Garcia-Molina and Page 1998] e o processo de classificação de páginas utilizada pelos mecanismos de busca, para descobrir comunidades *Web* e para organizar os resultados da pesquisa em *hubs* e autoridades. Um *hyperlink* contém a URL para a página que ele referencia e um texto âncora que descreve a ligação. O texto âncora pode oferecer excelentes descrições das páginas que ele referencia. Estes textos âncoras podem ser úteis para descrever e auxiliar na recuperação de páginas não indexadas, que contém elementos como imagens, arquivos de banco de dados e dados geográficos, por mecanismos de busca tradicionais.

A idéia de utilizar texto âncora foi inicialmente implementada na World Wide Web Worm [Mcbryan 1994] especialmente porque ele auxilia na busca de informações não textuais. O texto âncora permite conectar palavras (e contexto) a um conteúdo específico (por exemplo, Clique aqui para obter o mapa da cidade de São José dos Campos na escala 1:20.000).

O mecanismo desenvolvido utiliza o conceito de texto âncora para auxiliar na descrição do contexto e nos resultados da busca. Para melhorar os resultados obtidos utiliza também o texto âncora estendido. Neste caso, além do texto do *link*, as palavras e frases próximas dos links são consideradas para classificar os dados. A figura 1 ilustra os conceitos de texto âncora e texto âncora estendidos.

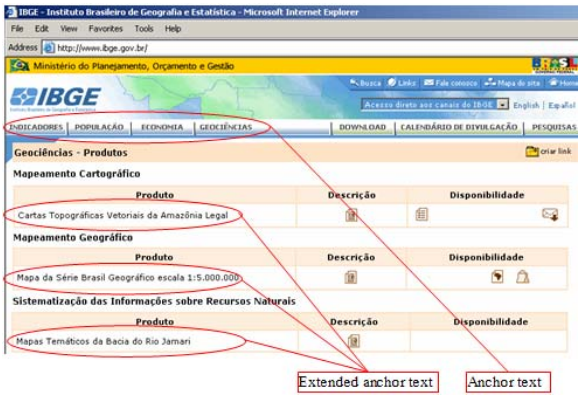

Figura 1. Texto âncora e texto âncora estendido.

Devido ao tamanho dos arquivos de dados geográficos, normalmente estes arquivos estão disponíveis na *Web* em formatos compactados na forma de arquivos zip, arj, rar e outros. Estes formatos não estão no foco de nosso mecanismo. Neste caso, o texto âncora tem outra importante função: ajudar os rastejadores localizar arquivos compactados de dados geográficos a partir da análise do contexto das páginas.

3.2. Processamento distribuído

Para suportar todas as requisições de busca e consulta, diferentemente de todos os mecanismos de busca existentes, nosso protótipo está baseado em processamento distribuído com um servidor de aplicações centralizado com *Web services*. Este servidor é responsável pelo gerenciamento das requisições dos clientes. Atualmente os principais mecanismos de busca têm dezenas de bilhões de páginas indexadas em seu banco de dados, desta forma, o processo de extração, análise e indexação dos sítios *Web* é lento. Outro problema é o tempo de revisita que acontece entre 30 e 60 dias, tornando o banco de dados desatualizado. Utilizar processamento distribuído com usuários colaboradores pode tornar estes processos mais eficientes, rápidos e exigir menores investimentos em infra-estrutura física.

3.3. Usuários colaboradores

Usuários colaboradores são clientes que ajudam no processamento quando seus computadores estão ociosos. Eles contribuem com o mecanismo desenvolvido rastejando a *Web* e executando a análise sintática para encontrar evidências de dados geográficos. Para se tornar um colaborador, o usuário precisa instalar o módulo que gerencia as tarefas relacionadas ao mecanismo de busca.

Dentre as vantagens de trabalhar com usuários colaboradores destacam-se a redução de investimentos para manter o projeto, uma vez que o processamento distribuído torna desnecessária a utilização de servidores poderosos para executar as funções de busca e análise e a capacidade de crescimento sustentável à medida que novos usuários colaboradores aderem ao projeto.

3.4. Outras características

Outra importante característica do mecanismo proposto é que seus rastejadores respeitam o código de ética dos rastejadores. Para evitar que um sítio específico seja indexado, são observadas algumas meta-tags tais como <*meta name = "robots" content = "noindex, nofollow"*>. Esta meta-tag é incluída no cabeçalho das páginas e o valor *robots* pode ser alterado pelo nome de um mecanismo específico (por exemplo *google* ou *yahoo*). O valor *noindex* estabelece para o robô que a página não pode ser indexada. O valor *nofollow* determina que os *links* eventualmente existentes na página não podem ser seguidos. Qualquer arranjo dos valores *index/noindex, follow/nofollow* é permitido.

4. Funcionamento do sistema

O mecanismo desenvolvido é implementado em C# e o servidor de banco de dados utiliza o SQL Server 2000. Existem duas arquiteturas principais: Servidores e Clientes. O papel básico do primeiro é gerenciar a distribuição de URLs, receber, organizar e armazenar os dados e arquivos capturados, e disponibilizar uma interface par a o usuário executar consultas na web e visualizar os resultados obtidos. Os clientes, executados nos computadores dos usuários colaboradores, os papéis são requisitar uma lista de URLs para serem visitadas, rastejar os sítios indicados, capturar as páginas, analisar o conteúdos das páginas, extrair os dados de interesse e enviar o conteúdo encontrado para o servidor. A figura 2 demonstra o funcionamento do sistema.

O tráfego de dados entre os clientes e o servidor é executado utilizando XML (*eXtended Markup Language*) e *Web services* para possibilitar a utilização da aplicação por computadores protegidos por *firewall* e *proxy*.

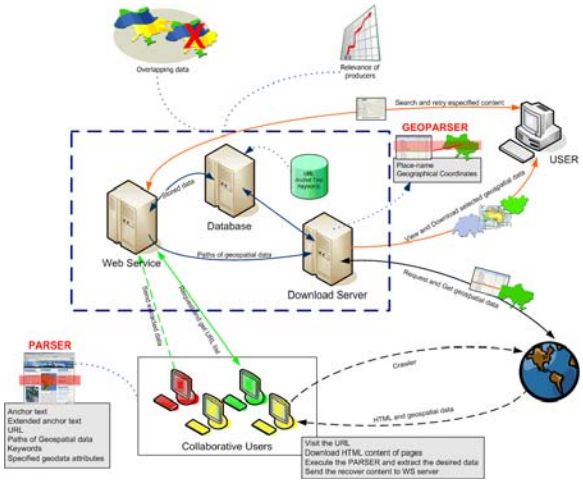

Figura 2. Funcionamento do mecanismo de busca.

O processo para descobrir dados geográficos é iniciado quando o computador de um usuário colaborador está ocioso. O cliente solicita uma lista de URLs para serem visitadas. O servidor de *Web service* (WS) envia um lista ordenada de URLs para o cliente. O cliente utiliza o rastejador para encontrar os sítios indicados e recuperar o conteúdo HTML das páginas visitadas. Então, seu conteúdo é analisado visando encontrar dados geográficos e extraindo os dados desejados (URLs, endereços dos dados geográficos, palavras chave, texto âncora e texto âncora estendido), e enviando estes dados para o WS que armazena os dados organizados no servidor de banco de dados (BD). Antes de serem armazenadas as URLs relativas são convertidas para URLs absolutas.

Novas URLs são ordenadas para serem visitadas posteriormente. A função de indexação é executada pelo indexador. O indexador executa várias funções: faz a leitura do repositório e o analisa, analisa também todos os *links* das páginas e armazena informações importantes sobre eles em arquivos específicos. Estes arquivos contêm informação suficiente para determinar para onde cada *link* aponta e o texto do *link*.

O servidor BD está conectado ao servidor de *downloads* (SD) por uma rede interna. O SD monitora os caminhos de arquivos incluídos no servidor de banco de dados e inicia o processo de *download* para cada inclusão. O SD busca o arquivo em seu local original, o captura e o armazena em um diretório. Então o arquivo é compactado para otimizar o espaço de armazenamento. Para a compactação e descompactação dos arquivos é utilizada a classe *GZipStream* disponível no .NET *framework* 2.0.

Sobre os arquivos armazenados é realizada a análise específica para os dados geográficos, buscando a projeção e o nome do local nas colunas dos arquivos DBF. Esta informação é ordenada e armazenada em um repositório do servidor BD e posteriormente é utilizada no processo de busca. Informações adicionais relevantes presentes nas colunas também são armazenadas.

4.1. Rastejador

O rastejador é responsável por visitar os endereços armazenados no servidor BD. A primeira tarefa executada pelo rastejador é a verificação do arquivo robots.txt no diretório raiz do servidor que hospeda o sítio visitado para certificar-se que não exista nenhuma regra que impeça a indexação das páginas. Informações detalhadas sobre o arquivo robots estão disponíveis em http://www.robotstxt.org/wc/norobots.html.

Após executar esta tarefa e certificar-se que a página pode ser indexada, o rastejador faz uma cópia de seu conteúdo por meio de um protocolo http. O rastejador utiliza classes que retornam um *stream* de bytes dos dados recebidos. Estes dados são convertidos para caracteres ASCII e o conteúdo da página é reconstruído de forma que o analisador possa executar o processo de análise e extração das informações.

O rastejador é uma aplicação complexa pois demanda a interação com milhares de servidores Web que extrapolam o controle do sistema [Brin and Page 1998]. Para visitar milhões de sítios da Web, o mecanismo desenvolvido tem um sistema distribuído de rastejadores que roda nos computadores dos usuários colaboradores. Atualmente cada rastejador visita 60.000 páginas por dia. Este é um excelente número uma vez que nosso mecanismo pode envolver centenas de usuários colaboradores aumentando consideravelmente a cobertura das páginas disponíveis na *Web*.

4.2. Analisador

O analisador é responsável pela extração de todo o conteúdo das páginas. Nestas páginas estão os endereços que serão armazenados na base de dados e posteriormente visitados pelos rastejadores. Assim que os rastejadores executam o *download* das páginas, todos caracteres são convertidos para minúsculos. O analisador verifica todo o conteúdo HTML para extrair algumas informações relevantes para a indexação da página. Entre este conteúdo estão as meta-tags: *<title></title>* que descreve o título da página, *<META NAME="Description" Content="">* que descreve o

conteúdo da página, <*META NAME=”Keywords” Content=””*> que armazena palavras-chave relacionadas ao conteúdo da página, <*a href=””*> que indica *links* para outras páginas e arquivos de dados geográficos.

O analisador remove todas as *tags* HTML, *scripts* e outras marcas mantendo o texto puro. As palavras extraídas são armazenadas em uma tabela de palavras no servidor BD. Para cada palavra um código *hash* é gerado para melhorar a velocidade no processo de busca.

4.3. Armazenador de endereço

O armazenador de endereço é responsável pelo armazenamento, no servidor BD, dos endereços extraídos pelo analisador e pela verificação do correto armazenamento dos endereços. Os endereços são utilizados pelo servidor WS para distribuir as URLs que serão visitadas e analisadas pelos usuários colaboradores.

4.4. Busca

O processo de busca é focado na qualidade dos resultados obtidos. A interface com o usuário é amigável e roda em navegadores tradicionais.

Para a execução da busca, o servidor WS verifica no servidor BD as palavras-chave relacionadas com as informadas pelo usuário. Quando o servidor WS encontra resultados que satisfaçam a consulta, ele retorna para o usuário uma lista de arquivos que atendam o conteúdo de sua busca. A classificação dos arquivos na lista é calculada considerando a proximidade dos termos de consulta com os termos encontrados no servidor BD e pela relevância do produtor.

Para cada item presente na lista de arquivos, são fornecidas informações adicionais tais como relevância do produtor, quantidade de dados geográficos disponíveis no sítio deste produtor, quantidade de *downloads* solicitados do arquivo. Estas informações auxiliam o usuário a conhecer melhor os dados geográficos antes de iniciar o *download*.

Os passos para a busca são:

1. O usuário acessa a interface do mecanismo e digita as palavras-chave para a busca.
2. O servidor WS analisa as palavras.
3. O servidor BD é percorrido até que exista um arquivo de dados geográficos que satisfaça as palavras-chave.
4. A classificação dos arquivos selecionados é calculada e os arquivos são ordenados.
5. O servidor WS envia uma lista de arquivos classificados e informações adicionais para o usuário.

O download dos arquivos pode ser executado através do servidor de downloads ou a partir de seu local de origem.

5. Conclusões

Este mecanismo foi projetado para tornar melhorar o acesso e recuperação de arquivos de dados geográficos disponíveis na Web. Ele foi implementado em um ambiente distribuído com usuários colaboradores. Os usuários colaboradores são muito interessantes pois auxiliam nos processos de recuperação e análise das páginas aumentando significativamente a cobertura da *Web*. Este cenário apresenta algumas vantagens: redução de investimentos para manter o mecanismo, uma vez que torna-se desnecessário servidores poderosos e grande capacidade de expansão do mecanismo à medida que novos usuários façam parte do projeto.

Para melhorar os resultados das buscas foram utilizadas técnicas para análise e recuperação de dados geográficos, dados sobrepostos, descrição de relevância dos produtores de dados, texto âncora e texto âncora estendido. O mecanismo apresentado possui funções para coletar, indexar e executar buscas em dados geográficos.

Algumas funções estão sendo desenvolvidas para completar o projeto, dentre elas a utilização de *gazetteers* para comparar nomes de lugares; a ampliação dos rastejadores a fim de reconhecer outros formatos GIS tais como arquivos *spr* e *geotiff*, a inclusão de ontologias para melhorar os resultados retornados à consulta dos usuários, a implementação de um sistema amigável de pré-visualização dos arquivos de dados geográficos através do navegador.

6. Referências

[1] Brin, S. and L. Page (1998). The anatomy of a large-scale hypertextual Web search engine. Seventh International World Wide Web Conference, Brisbane, Australia.

[2] Cho, J., H. Garcia-Molina and L. Page (1998). Efficient Crawling Through URL Ordering. Seventh International Web Conference (WWW 98). Brisbane, Australia.

[3] Gerhart, A. (2002). "Understanding and Building Google PageRank." from http://www.searchengineguide.com/orbidex/2002/0207_orb1.html

[4] Giles, C. L., K. Bollacker and S. Lawrence (1998). CiteSeer: An automatic citation indexing system. Digital Libraries 98 - The Third ACM Conference on Digital Libraries, Pittsburgh, PA, ACM Press.

[5] Glover, E. J. T., K.; Lawrence, S.; Pennock, D.; Flake, G. W (2002). Using Web Structure for Classifying and Describing Web Pages. WWW2002, Honolulu, Hawaii, USA.

[6] Howe, W. (1996). "When did the Internet start? A brief capsule history." from http://intranet.canacad.ac.jp/instruction/internet_skills/brief_history/!history.html

[7] Lawrence, S. and C. L. Giles (1998). "Searching the World Wide Web." Science **280**(5360): 98-100.

[8] Lawrence, S. and C. L. Giles (1999). Text and Image Metasearch on the Web. International Conference on Parallel and Distributed Processing Techniques and Applications, PDPTA 99.

[9] Mcbryan, O. A. (1994). GENVL and WWWW: Tools for Taming the Web. First International Conference on the World Wide Web, Geneva, CERN.

[10] Page, L., S. Brin, R. Motwani, et al. (1999). "The PageRank Citation Ranking: Bringing Order to the Web." from <http://dbpubs.stanford.edu/pub/1999-66>.