Um estudo sobre o uso de agentes de internet em buscas (Junho 2010)

Alexandre Fatayer Canova, RA 107214, UNICAMP

Agentes de internet são parte integrante da web na forma como conhecemos e usamos a intenet atualmente. Neste artigo, serão consideradas suas aplicações, possibilidades e limitações, juntamente com uma implementação para demostrar alguns dos conceitos discutidos.

***Termos*—Agentes, Internet, Rastreadores, Pesquisa**

I. INTRODUÇÃO

Atualmente, a World Wide Web[1] é baseada principalmente em documentos HTML[2] , uma linguagem para descrever, com ênfase na apresentação visual, um corpo estruturado de texto juntamente com objetos multimídia, como imagens e formulários interativos.

Os agentes de internet constituem uma parte importante da Web, porque mesmo que indiretamente, seus serviços são freqüentemente utilizados por todos. Toda vez que realizamos uma pesquisa na Internet, através de um sistema de busca, como Google ou Yahoo, por exemplo, fazemos uso de um índice que foi baseado no conteúdo obtido de um agente.

Os sistemas de pesquisa utilizam os agentes de internet que funcionam como rastreadores para pesquisar a Web e os resultados obtidos são indexados.

Um rastreador é um software que percorre a estrutura de links da Web automaticamente, recuperando um documento e recursivamente inter-relacionando todos os documentos a que ele possui vínculos descritos em sua parte interna

II. POSSIBILIDADES DE APLICAÇÃO

A tecnologia baseada no rastreamento é útil para uma ampla variedade de aplicações baseadas na Web, como por exemplo:

A. Indexador de conteúdo de pesquisas

Promove a possibilidade de pesquisas rápidas através de palavras-chaves encontradas durante o rastreamento de endereços.

B. Detector de links quebrados

Utilizado para varrer todo o conteúdo de um site e apresentar um relatório dos links com problemas, útil para tarefas de manutenção de sites.

C. Comparador

Utilizado para varrer determinados sites em busca de alguma característica (como o menor preço) de algum item. Um representante de grande expressão nesta categoria é o Buscapé (http://www.buscape-inc.com/).

D. Arquivador

Utilizado para salvar todas as páginas de sites, para permitir a visualização offline, fazer backups ou indexar o conteúdo para permitir pesquisas mais rápidas.

III. PESQUISA DE CONTEÚDO

A. Sistema de busca

Os sistemas de busca atuais normalmente consistem de três partes distintas, cada qual rodando em uma rede distribuída de milhares de computadores de baixo custo, provendo desta forma processamento rápido de forma paralela. Estes três processos podem ser definidos como:

- Um agente Web, também conhecido como robô ou web crawler.
- Um indexador, que indexa cada palavra de cada página e armazena o resultado das palavras indexadas em um grande banco de dados.
- Um processador de consultas, que compara o termo a ser pesquisado com o índice existente, recomendando os documentos considerados mais relevantes.

Este processo é executado pela maioria dos serviços de busca existentes na Web.

[1] Rede de computadores na Internet que fornece informação na forma de hipertexto.

[2] HyperText Markup Language – Linguagem de marcação de hipertexto.

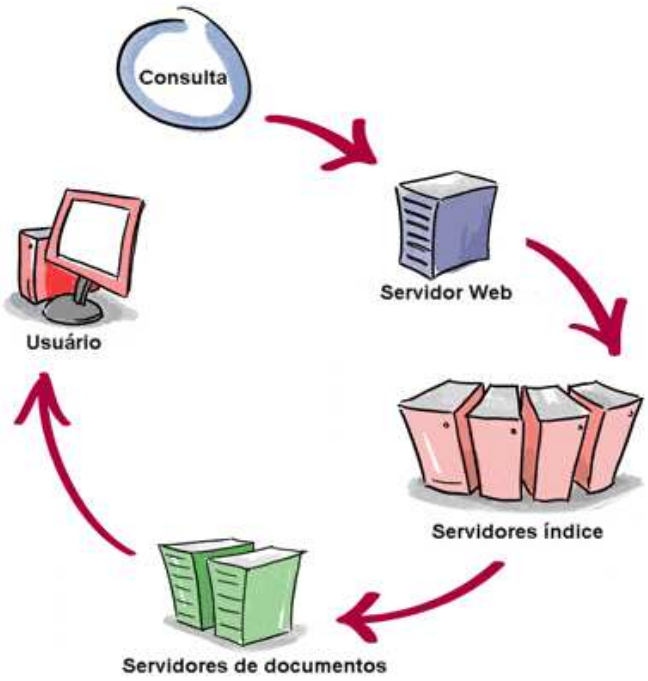

Fig. 1. Funcionamento da pesquisa na Web

O servidor Web envia a consulta para os servidores de índice. O conteúdo armazenado por estes servidores é semelhante ao índice encontrado no final dos livros, pois indica quais páginas contém as palavras que coincidem com o termo da consulta. A consulta então é transferida para os servidores de documentos, que recuperam os documentos armazenados. Neste momento são geradas pequenas descrições para cada resultado obtido. Os resultados são retornados para o usuário em uma fração de segundo.

A funcionalidade de um agente Web que atua como rastreador é como a de um Web browser, enviando uma requisição ao servidor por uma página Web e fazendo o download da página inteira. Geralmente o processo consiste de diversos computadores fazendo requisições e obtendo páginas muito mais rapidamente que um usuário com um Web browser. De fato, o rastreador possui capacidade para trabalhar com centenas de páginas diferentes simultaneamente.

B. Algoritmos

Na essência, todos os rastreadores Web são fundamentalmente os mesmos. O funcionamento dos rastreadores se baseia no seguinte processo:

- É realizado o download da página Web.
- A seguir é realizada a análise sintática da página descarregada e recuperado todos seus links.
- Por fim, para cada link recuperado, é repetido novamente o processo para construir novas buscas.

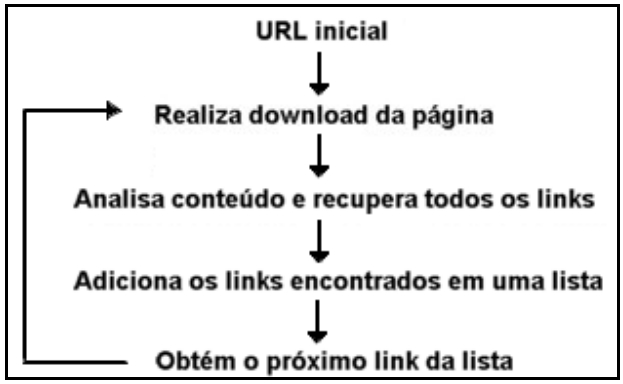

Fig. 2. Algorítimo de busca

No primeiro passo, o rastreador, através de uma URL inicial, faz o download da página Web. Freqüentemente, a página descarregada é salva em um arquivo em disco ou colocada em um banco de dados. Salvar a página permite ao rastreador ou outro software voltar mais tarde e manipular a página para diversos fins.

No segundo passo, o rastreador analisa sintaticamente a página descarregada e recupera os links para outras páginas. Cada link da página é definido com um tag de âncora de HTML, como por exemplo:
<A HREF=http://www.nomedoservidor.com/pasta/arquivo.html>Link</A>

Depois que o rastreador recupera os links da página, cada link é adicionado a uma lista de links a serem varridos.

O terceiro passo para a varredura da Web repete o processo. Os links podem ser varridos *primeiro em profundidade* ou *primeiro em largura*. A varredura primeiro em profundidade segue cada caminho possível até sua conclusão antes de outro caminho ser tentado. Ela funciona localizando o primeiro link na primeira página. Ela varre, então, a página associada àquele link, localizando o primeiro link da nova página, e assim por diante, até que o fim do caminho seja alcançado. O processo continua até que todas as ramificações de todos os links sejam esgotadas.

A varredura primeiro em largura verifica cada link de uma página antes de prosseguir até a próxima página. Portanto, ela varre cada link da primeira página e depois varre cada link do primeiro link da primeira página, e assim por diante, até que cada nível de links seja esgotado. Por exemplo, nos endereços:

Fig. 3. Ordem de varredura

Primeiro são verificados todos os links presentes nas áreas “dir1”, “dir2” e “dir3”, pois pertencem ao mesmo nível. Depois é iniciado o processamento no nível seguinte, que envolve as áreas “subdir1”, “subdir2” e “subdir3”.

C. Protocolo Robot

Ao se fazer a varredura de um Web site, pode-se gerar uma grande sobrecarga sob os recursos do servidor Web, pois muitas requisições podem ser feitas em um curto espaço de tempo.

Em geral, algumas poucas páginas são descarregadas por vez de um Web site, e não centenas ou milhares sucessivamente. Os Web sites também freqüentemente possuem áreas restritas que os rastreadores não devem varrer. Em uma tentativa de padronizar a utilização dos rastreadores, muitos Web sites adotaram o protocolo Robot, que estabelece as diretrizes que os rastreadores devem seguir. Com o tempo, o protocolo tornou-se um padrão na Internet para os rastreadores e é considerada boa prática utilizá-lo (THE WEB ROBOTS PAGES, 2010).

O protocolo Robot define que um arquivo chamado *robots.txt* deve existir no diretório raiz do Web site. Dentro deste arquivo pode-se definir as áreas cujo acesso é restrito para alguns ou todos os agentes. Durante o funcionamento do rastreador Web, este arquivo deve ser consultado, determinando quais partes do site não são autorizadas para varredura.

Exemplo de um arquivo robots.txt:

```
#robots.txt para http://nomedoservidor.com/
User-agent: *
Disallow: /cgi-bin/
Disallow: /registration
Disallow: /login
```

Na primeira linha, há um comentário, indicado pelo uso do caractere #. O rastreador Web ignora os comentários no arquivo.

Na terceira linha do arquivo de exemplo, é especificado o agente do usuário ao qual as regras se aplicam. Agente do usuário é um termo utilizado pelos programas que acessam um Web site, como um Web browser. Os rastreadores também enviam, em geral, um valor de agente do usuário junto com cada solicitação a um servidor Web. O uso de agentes do usuário do arquivo robots.txt permite a Web sites configurarem regras individualmente. Entretanto, os Web sites geralmente não querem autorizar o acesso de todos os rastreadores. Isso especifica que todos os agentes do usuário estão desautorizados pelas regras.

A desautorização de todos os agentes do usuário não impede os Web browsers de funcionar, já que estes softwares não seguem o protocolo Robot.

As linhas seguintes à linha do agente do usuário são chamadas instruções de não-autorização (disallow statements). As instruções de não-autorização definem os caminhos do Web site a que os rastreadores não têm permissão de acesso. Por exemplo, a primeira instrução de não-autorização do arquivo de exemplo instrui os rastreadores a não varrer qualquer link que inicia com “/cgi-bin/”.

Portanto, os URLs http://nomedoservidor.com/cgi-bin/, http://nomedoservidor.com/cgi-bin/register são ambos inacessíveis aos rastreadores de acordo com essa linha. As instruções de não-autorização são para caminhos e não para arquivos específicos; assim, qualquer link solicitado que contenha um caminho na lista de não-autorização é recusado.

D. Implementação

Para exercitar os conceitos discutidos, foi criado um projeto com enfoque prático, utilizando o protocolo Robots, onde questões como busca por palavras chave, processamento em paralelo, técnicas de varredura são abordadas.

A implementação do algoritmo de rastreamento relacionado com a estrutura de links da Web foi desenvolvida na linguagem de programação Java, escolhida por funcionar em diferentes ambientes e possuir muitos recursos integrados, como acesso à rede, threads e logging.

A aplicação que demonstra a utilização deste algoritmo fornece uma tela onde é possível informar diversos parâmetros, como endereço inicial, termos de pesquisa, número de consultas simultâneas, etc, (Figura 4).

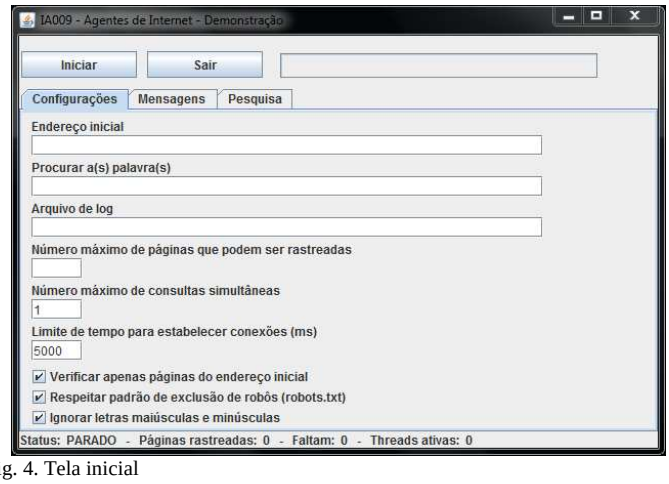

Fig. 4. Tela inicial

O processo de rastreamento de links começa a partir do endereço inicial fornecido pelo usuário. A página apontada por este endereço é descarregada, e então é feita a análise de seu conteúdo, recuperando uma lista de todos os links encontrados. Para cada link encontrado, o processo é repetido.

O algoritmo não faz uso de recursividade para evitar a sobrecarga dos recursos do sistema, ou seja, não segue cada caminho possível até sua conclusão antes de outro ser tentado, pelo contrário, é feita a verificação de cada link dentro de uma página antes de prosseguir até a próxima.

A medida que os links são encontrados são adicionados em uma lista de links a processar, e assim que são processados, são removidos. Também é mantida uma outra lista contendo os links já processados, para evitar que o rastreador entre em um círculo (site A aponta para site B que aponta novamente para site A).

O uso do termo de pesquisa é opcional, e não influência no processo de rastreamento dos links encontrados em cada página. Quando utilizado, uma etapa a mais é realizada, que consiste em separar as palavras encontradas e verificar se coincidem com que o usuário forneceu.

Geralmente ao iniciar a execução do rastreador, o tempo e o número de links a serem processados são imprevisíveis, por isso, o processo de rastreamento pode ser limitado pelo número máximo de páginas rastreadas ou verificar apenas links referentes ao servidor inicial fornecido.

Um dos itens que mais influenciam na performance de rastreamento é o tempo de espera para o estabelecimento de conexões e download das páginas. Pode-se definir um limite de tempo em milisegundos para se obter a conexão com o endereço a ser verificado, antes deste ser descartado.

Com a utilização de apenas uma thread , o rastreador deve esperar o estabelecimento de conexões, download e análise das páginas antes de seguir para o próximo link disponível. Através do campo “Número máximo de consultas simultâneas”, pode-se informar o número máximo de threads disponíveis para a execução do algoritmo de rastreamento, resultando em um melhor desempenho.

As threads entram em execução apenas quando existem links a serem varridos, e entram em um estado “adormecido” quando não há links suficientes para todos.

Muitas vezes os sites possuem áreas em que os administradores não desejam que sejam rastreados. O rastreador respeita estes limites através do padrão de exclusão de robôs (robots.txt), desde que a opção correspondente esteja selecionada. As áreas cujo acesso é restrito são armazenados em um map , cuja chave é o nome do servidor. Através desta chave, é armazenada uma lista de áreas onde o acesso é proibido. Desta maneira, evita-se a pesquisa constante ao arquivo robots.txt, pois o arquivo é verificado apenas no primeiro acesso ao servidor, melhorando a performance.

A partir da barra de status disponível, é possível verificar a situação atual do rastreador (parado, parando ou rastreando), o número de páginas já rastreadas e que ainda faltam ser verificadas e o número de threads ativas no momento.

Com a utilização do mecanismo de logging do Java, (disponível no pacote java.util.logging), todas as operações executadas, informações e advertências são registradas e ficam disponíveis na aba Mensagens (Figura 5).

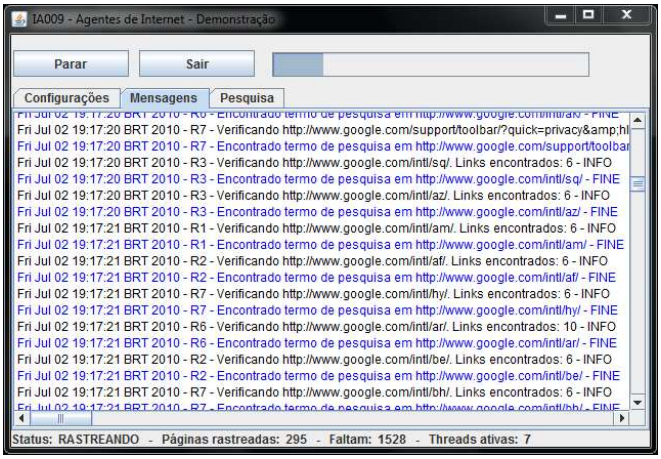

Fig. 5. Log das operações realizadas

Opcionalmente pode ser criado um arquivo de log adicional, que é informado pelo usuário através do campo “Arquivo de log”.

O formato do log é composto pela data e horário da ocorrência, número da thread execução (nome R + número da thread) e a mensagem. Mensagens na cor preta representam operações normais de processamento, em vermelho estão as advertências gerais (arquivo não encontrado, limite de tempo ultrapassado, etc), e em azul ficam os resultados das pesquisas pelas palavras-chave fornecidas pelo usuário. Estes resultados também são registrados em uma aba própria chamada Pesquisa (Figura 6).

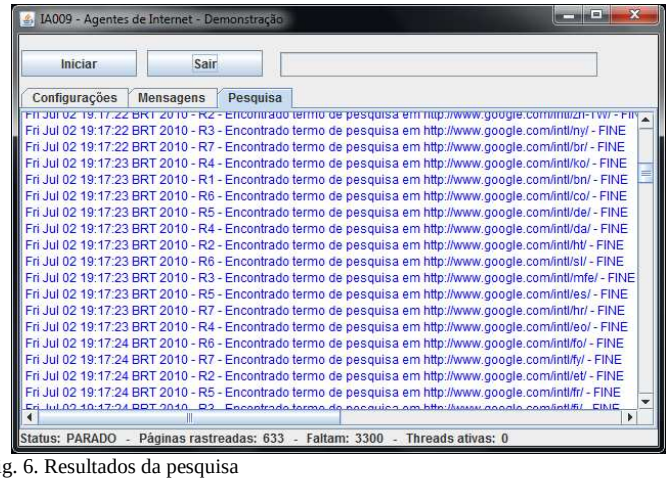

Fig. 6. Resultados da pesquisa

A implementação deste algoritmo se restringe à demonstração do conceito de rastreamento de links da Web, não oferecendo uma solução viável para utilização em um ambiente real.

Uma das limitações desta implementação está na maneira de como os dados são armazenados. Todo o controle interno (como links já rastreados e a rastrear) é feito através de dados

em memória, que invariavelmente tende a se esgotar com o tempo (se os limites da varredura não foram especificados), devido ao crescimento exponencial do número de links a serem processados.

Uma possível solução para este problema seria a utilização de um banco de dados para a persistência das informações de controle, tornando o uso da memória estável durante todo o processamento.

Além disso, tornaria possível também o armazenamento da informação que é descarregada de cada página, que pode ser útil para diversos fins, como indexação e arquivamento.

IV. LIMITAÇÕES E ALTERNATIVAS

O HTML possui uma capacidade limitada para fazer a classificação de blocos de texto em uma página, além das regras que se aplicam na organização do documento, e isto muitas vezes se reflete nas pesquisas realizadas na Web, que retornam resultados fora do contexto correto.

A Web Semântica corrige esta limitação, usando tecnologias para permitir a atribuição de descrições e significado ao conteúdo na Web, facilitando desta forma a pesquisa realizada pelos computadores.

Para atingir estes objetivos, a Web Semântica define as relações entre os dados através de ontologias, que estabelecem um conjunto de termos de conhecimento, e faz uso de linguagens que permitem expressar ao mesmo tempo o significado e as regras de raciocínios sobre eles.

A base destas tecnologias são o XML, que permite o uso de documentos contendo uma estrutura clara e precisa da informação que é armazenada, e o RDF, que permite o processamento de metadados e a troca de informações compreensíveis na Web. Juntos, XML e RDF se completam, enquanto um define a estrutura, o outro permite expressar o significado associado aos dados.

V. CONCLUSÃO

Atualmente os agentes de internet desempenham um papel importante para o sucesso da Internet. Praticamente todos os serviços de busca da Web (um dos tipos de aplicações mais utilizados) fazem uso de suas informações coletadas.

Podem ser utilizados como base para outros serviços, como a comparação de preços entre diversos fornecedores e o arquivamento de páginas.

REFERÊNCIAS

[1] DEITEL, H.M.; DEITEL, P.J., Java, Como Programar, Bookman. 4º ed. 2003. Brasil.
[2] HEATON, J., Programming Spiders, Bots, and Aggregators in Java, Sybex. Fev. 2002. USA.
[3] HAROLD, E. R., Java Network Programming, O'Reilly, 3º ed. 2004. USA.
[4] SCHILDT, H., Java 2 – The Complete Reference, Osborne. 5º ed. USA.
[5] The Web Robots Pages, disponível em: http://www.robotstxt.org/wc/robots.html
[6] BLUM, T; KEISLAR, D.; WHEATON, J., Writing a Web Crawler in the Java Programming Language, disponível em: http://java.sun.com/developer/technicalArticles/ThirdParty/WebCrawler/index.html
[7] RAPPOPORT, A., Checklist for Search Robot Crawling and Indexing, disponível em http://www.searchtools.com/robots/robot-checklist.html.
[8] OLIVEIRA, R. M. V. B., Web Semântica: Novo Desafio para os Profissionais da Informação, disponível em: http://www.sibi.ufrj.br/snbu/snbu2002/oralpdf/124.a.pdf
[9] Google Guide, disponível em: http://www.googleguide.com