Políticas eficientes de revisitação de páginas com Web Crawlers

Elson Silva Costa *e-mail*: escosta@hotmail.com
Tiago Mesquita de Araujo Cunha. *e-mail*: tiagomac@gmail.com
Instituto Federal da Bahia - IFBA.
Grupo de Pesquisa em Sistemas Distribuídos, Otimização, Redes e Tempo Real.

***Resumo*—Milhares de páginas são criadas ou modificadas diariamente na internet. Em um cenário dinâmico como a internet se apresenta o conteúdo gerado pode ficar perdido em um espaço onde não existem referências. Para solucionar esse problema os Web Crawlers foram criados para buscar e indexar esse material não referenciado. Para o correto funcionamento os Web Crawlers seguem políticas adotadas em seu algoritmo para que a revisitação de páginas seja realizada de modo adequado, não sobrecarregando servidores e realizando a correta organização desses dados.**

***Index Terms*—Web Crawlers, indexação, políticas eficiêntes, algoritmos**

I. Introdução

O OBJETIVO deste artigo é apresentar políticas eficientes de indexação de páginas na WEB partindo do princípio básico dessa técnica e abordando os mecanismos existentes para o funcionamento correto, assim como algoritmos eficientes de pesquisa.

Diariamente na internet é gerado um grande volume de dados dispersos de modo que mecanismos auxiliares como indexadores se tornam ferramentas necessárias para levar esse conteúdo ao usuário final. Segundo dados de [1] é possível observar que em apenas sessenta segundos 571 novos sites são criados, 20 mil fotos são enviadas ao serviço Tumblr, 3.600 ao serviço Instagram e 347 novos posts são gerados em Wordpress. Para indexar e categorizar todo esse conteúdo serviços como o Google realizam 2 milhões de pesquisa a cada sessenta segundos. Todo esse volume de dados apresentado gera um problema de organização e referência, uma vez que o conteúdo gerado não esteja de fácil acesso ao usuário final ele se torna obsoleto na internet. Para resolver o problema relacionado à organização, busca e categorização de todo o conteúdo gerado foram criados os *Search Engine*, sendo esses mecanismos divididos basicamente nas categorias de Web Crawling, Indexação e Busca. Serviços como o Cadê! funcionam no sistema de indexação, onde o usuário acessa a página do serviço e fornece os dados de seu conteúdo. Os *Web Crawling*, objetivo de estudo desse trabalho apresentam um mecanismo de pesquisa onde através de palavras chaves e links realiza a busca e ordenação de conteúdo disposto na internet, desse modo podemos categorizar um *web crawler* como um programa que coleta conteúdos da rede mundial de computadores de forma metódica e automatizada, tendo seu funcionamento similar ao de um robô. Para tanto os *web crawlers* montam uma árvore de páginas por onde encontram outras páginas através das quais novos links são pesquisados. Na figura 1 é possível visualizar uma representação gráfica do funcionamento de um *web crawler*

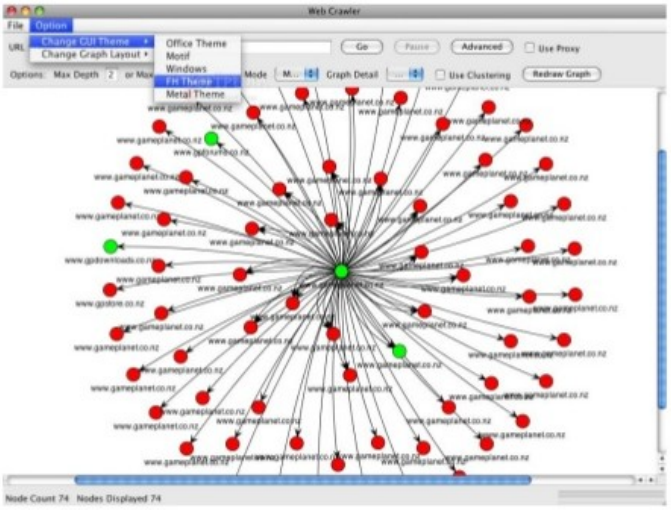

Figura 1. Representação gráfica de um *Web Crawler*. Fonte: [2]

Na seção II desse trabalho será apresentada uma introdução aos *web crawlings*, na seção III e suas subseções serão abordadas as políticas de funcionamento eficiente e na seção IV a conclusão.

II. Introdução a Web Crawling

A técnica de *web crawling* consiste em realizar uma busca contínua de links em páginas na Web. A cada nova página encontrada uma nova busca por links é iniciada no conteúdo dessa página, para tanto os *web crawlings* fazem uso de um *Parser* responsável pela extração e armazenagem dos dados extraídos em uma base de dados. Após a extração dos dados os links entram em uma fila para que suas respectivas páginas sejam obtidas através de um componente de *download*. Do funcionamento básico de um *web crawler* podemos entender que o conteúdo mais importante da página, além das palavras chaves são as referências a *links* capturados através da tag html href. Na figura 2 é possível visualizar os componentes

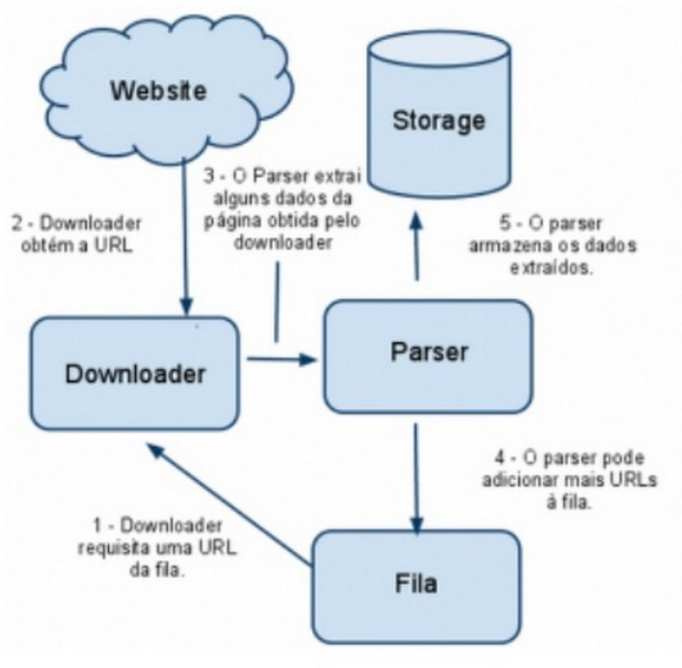

Figura 2. Representação gráfica dos componentes de um*Web Crawler*.Fonte: [2]

básicos de um *web crawler*. [2]
Serviços baseados em *search engine* fazem uso de duas técnicas de pesquisa, sendo elas a vertical e a horizontal. Na técnica de pesquisa vertical o conteúdo buscado é específico por categoria ou tipo de mídia, na busca horizontal o conteúdo não segue rigor de critério. São exemplos de pesquisa vertical os serviços Google Video e Google Images, enquanto o serviço Google Search mescla atributos de uma pesquisa horizontal e vertical, através dos *snippets* avançados que podem trazer além das informações básicas também imagens e vídeos de algumas consultas. Um *web crawler* se identifica ao servidor Web através do campo *User-Agent* de um *HTTP request*. Em alguns casos o *web crawler* pode omitir sua identificação não informando ou alterando esse valor. Pelo alto volume de informações obtidas no processo de *crawling* é necessário priorizar downloads de páginas mais importantes, nesse contexto é necessária a adoção de políticas que tornem a consulta e busca de páginas o mais eficiente possível. [2]

III. Políticas eficientes para *Web Crawling*

Com o surgimento dos *crawlers* diversas questões começaram a ser levantadas, como quais domínios um *crawler* deve ou não visitar, qual o conteúdo ele deve pesquisar e até mesmo questões técnicas que implicam na disponibilidade dos serviços pesquisados. Dessa forma, conforme [3] foram criadas políticas de eficiência na implementação de um *crawler* que tratam desses questionamentos. Nas subseções seguintes serão apresentadas as políticas de seleção, revisitação, cordialidade e paralelização.

A. Política de Seleção

Devido à limitação existente na capacidade dos *crawlers* de baixar e analisar todas as páginas existentes na WEB é desejável que o *crawler* possua um sistema de *rank* capaz de identificar os domínios e as páginas mais importantes a serem analisadas. O *rank* gerado por um *crawler* podem seguir o conceito de *path-ascending crawler*, onde os caminhos plausíveis de uma dada URL são verificados como um todo, esse formato de pesquisa também é conhecido como *Harvester* devido a sua capacidade de cavar domínios e subdomínios de um determinado endereço. Determinados crawlers de busca vertical utilizam de uma técnica chamada *focused crawler* ou *tropical crawlers* para buscar conteúdo específico. O principal problema dos *crawlers* baseados nessa técnica é a necessidade de realizar todo o download do conteúdo da página antes da análise. A performance desses crawlers depende muito do detalhe das descrições de links.

B. Política de Revisitação

A política de revisitação serve para indicar quando cada página será atualizada e gerencia a inclusão de páginas a serem visitadas pelo *crawler*. Uma política ideal de revisitação deve manter essa lista sempre atualizada, mas de maneira a não sobrecarregar a lista de páginas a serem visitadas pelo *crawler*. As funções mais adotadas na política de revisitação são as de atualidade e idade, sendo a de atualidade a medida binária que indica quando a cópia local está ou não atualizada e a idade a medida que indica a diferença de tempo entre a data de download e a última atualização da página se essa for maior que a data de download. Duas políticas simples de revisitação estudadas por [4] e [5] são a política uniforme, onde todas as páginas na coleção são revisitadas com a mesma frequência, desconsiderando desse modo a frequência de atualização, já na política proporcional as páginas são revisitadas conforme a sua taxa de atualização, logo as páginas que sofrem mais atualizações são rapidamente revisitadas. Em testes realizados por [4] a visitação uniforme se mostrou mais eficaz em ambiente simulado e real, isso ocorre pelo fato de que páginas com pouco índice de atualização sofrem menos revisitação de modo a ficarem desatualizadas.

C. Política de Cordialidade

As revisitações constantes em páginas na Web podem gerar um consumo excessivo de banda e recursos operacionais dos domínios que as hospedam. Desse modo os *crawlers* devem seguir uma política de cordialidade onde essa revisitação seja corretamente administrada. Koster [6] sugere um conjunto de normas de comunicação através de um arquivo texto denominado Robots Exclusions, se o *crawler* seguir esse protocolo ele pode ser advertido pelo domínio a não praticar revisitação invasiva ou mesmo serem instruídos a não baixar determinados conteúdos onde o tamanho pode carregar o domínio. A primeira proposta de Koster de intervalo entre revisitação foi de 60 segundos, porém considerando

um domínio com 100.000 páginas em uma conexão ideal com baixa latência e banda infinita ainda levaria 2 meses para que um *crawler* pudesse indexar todas as páginas do domínio, desse modo outros estudos indicam que o *crawler* deve considerar aspectos técnicos do domínio onde busca informações para não sobrecarrega-los em suas consultas.

D. Política de Paralelização

A política de paralelização trata da divisão de responsabilidades entre os *crawlers* de uma mesma rede para visitação de páginas na Web. No que diz respeito à política de paralelização podemos definir essa política em dois modos distintos: associação dinâmica ou associação estática. Na primeira a lista de sites a visitar pelo *crawler* ficam em um coordenador, responsável em repassar essa lista para os *crawlers* responsáveis pela visitação. Na estática os *crawlers* ficam responsáveis em realizar a visita com base em critérios previamente estabelecidos, por exemplo, o *crawler* A deve acessar URLs que iniciam pela letra "A", o *crawler* B deve acessar URLs que iniciam com a letra "B", sendo esse apenas um exemplo e sabendo que outros processos de distribuição de responsabilidades podem ser adotados. Apesar de no processo estático não existir um coordenador os *crawlers* podem trocar informações entre si.

IV. Conclusão

Nas seções desse trabalho fizemos uma introdução aos *web crawlers* e suas políticas de eficiência.

Sabendo da crescente utilização e surgimento de mecanismos de pesquisa ou *search enginer* com características de *crawling* podemos observar que os *crawlers* já fazem parte da internet e que a incorreta utilização deles pode gerar problemas para os administradores de domínios, conteúdo e usuários. Para uma correta convivência ainda é necessário um estudo mais aprofundado em pesquisas de impacto de sua utilização. Algumas soluções já foram elaboradas no que tende a padronizações e normas de funcionamento. Koster. M em [6] elaborou em 1993 algumas instruções que seguidas podem minimizar o impacto causado pela utilização dos *crawlers*, essas instruções devem ser seguidas por desenvolvedores de *crawlers* e mantenedores de domínios e ficam descritas em um arquivo robots.txt no domínio visitado. Nesse arquivo o *crawler* recebe instruções do administrador do domínio para realizar, restringir ou anular a pesquisa do *crawler*. Algumas normas também devem ser seguidas pelo *crawler*, conforme [6] o *crowler* deve se identificar através do campo HTTP *user-agent* para o domínio que realiza a visitação assim como notificar autoridades quanto ao seu funcionamento em redes como comp.infosystems.www.providers. No entanto, apesar de todo o esforço na criação de políticas de eficiência e padrões na utilização dos *crawlers* ainda é comum encontrar serviços que não seguem essas normas para obter vantagem sobre seus concorrentes. Uma das formas utilizadas por *crawlers* mal intencionados é a omissão de sua identificação ou substituição de sua identificação pela de um programa conhecido que realiza requisições, como um *browser*, se passando desse modo por um usuário comum. Para solucionar esse e outros problemas administradores de domínio criam políticas de segurança que terminam por afetar os usuários, como a utilização de CAPTCHA (*Completely Automated Public Turing test to tell Computers and Humans Apart*) que são testes realizados para identificar se a requisição parte de um usuário comum ou de um robô.

Ainda existe muito a ser estudado sobre os *crawlers* e seu impacto, mas já é perceptível a sua importância e na medida em que ocorre seu crescimento a Web como um todo se adapta ao seu funcionamento de modo a não ser gravemente afetada.

Referências

[1] V. WOOLLASTON. (2013) Revealed, what happens in just one minute on the internet: 216,000 photos posted, 278,000 tweets and 1.8m facebook likes. MailOnline. [Online]. Available: http://goo.gl/WDbl7R

[2] D. S. Bonafé. (2005, Oct.) Noções de search engine. [Online]. Available: http://goo.gl/D9IM9B

[3] P. V. and M. M. (2011) Crawling na web pública e na web escondida. [Online]. Available: http://goo.gl/EIb8ja

[4] V. Cothey, *Web-crawling reliability*. Journal of the American Society for Information Science and Technology, 2004, no. 1228-1238.

[5] S. Raghavan and H. Garcia-Molina, *Web-crawling reliability*, 2001. [Online]. Available: http://www.vldb.org/conf/2001/P129.pdf

[6] M. Koster. (1993) Guidelines for robot writers. [Online]. Available: http://www.robotstxt.org/guidelines.html