Hython – Web Crawler

Jéssica do Nascimento Piroupo Francisco, Luana Fares, Simone de Oliveira, Wander Camilo

Bacharelado em Ciência da Computação, Senac São Paulo
São Paulo – SP – Brasil

jessicapiroupo@hotmail.com, luanafares@hotmail.com,
nlmony@hotmail.com, wander_camilo@hotmail.com

Resumo: *O projeto apresenta um WebCrawler, desenvolvido em linguagem Python 3.2. A interface, desenvolvida com HTML5, CSS3, JavaScript e JQuery, facilita a utilização do programa e apresenta os dados retornados com mais clareza e interatividade.*

1. Introdução:

Um Web Crawler consiste em um programa que navega pela internet de forma metódica, automatizada e organizada.

Foi proposto um projeto que tem como objetivo desenvolver um Web Crawler em linguagem Python e integrá-lo a uma interface desenvolvida em HTML5, com recursos como CSS3, JavaScript e JQuery.

Os dados obtidos com o programa devem ser apresentados de forma clara para o usuário, e para dar início ao processo, é necessário que uma URL seja informada.

Em posse da URL, o programa irá retornar as informações do site, como: links internos, links externos, imagens e documentos.

Em um arquivo .txt, foram inseridas manualmente algumas palavras-chaves que serão utilizadas para categorizar o site e os links obtidos, caso algum site não possuir meta tags em seu código fonte.

2. Materiais e Métodos:

Para desenvolvimento do projeto, foram feitas algumas exigências. Para a codificação do Web Crawler, deve-se utilizar a linguagem Python, em uma versão 3.x. A versão escolhida da linguagem Python foi a 3.2.2. Deve-se ter também, uma interface para visualização dos dados obtidos, feita em HTML5 e que fossem utilizados recursos como CSS3, JavaScript e JQuery.

Após a pesquisa de conteúdo da linguagem Python, foram pesquisadas as novas bibliotecas da versão 3.2.2 para prosseguir com o código do projeto, em seguida pesquisa sobre diferenças e mudanças do HTML4 para o HTML5 e aprofundamento de conhecimento de CSS3, JavaScript e JQuery.

Ao final foram aplicadas uma série de testes para verificar o funcionamento do programa para a aplicação final do design.

3. Métodos Utilizados

Para a classificação de uma URL, caso não haja meta tags no código fonte, é feita uma leitura do código, onde é efetuada uma análise da frequência em que as palavras contidas aparecem na lista de palavras-chave pré-definidas, e então, é sugerida uma categoria para o site. Por exemplo, a palavra **computador** está relacionada à informática, logo, se ela for a palavra que mais aparecer no texto, pode-se concluir que o site tem uma grande probabilidade de ser relacionado à informática.

Alguns sites apresentam Meta Tags em seu Head, que são palavras-chaves que definem os temas da página para melhor identificação de sistemas de busca. Caso o site possua Meta Tags, a pesquisa é feita nesse próprio campo ao invés de utilização de frequência de palavras em um código,

A partir do HTML, também é extraído todos os links, que são verificados individualmente, classificados de acordo com sua categoria e subdivididos para que possam ser apresentados em um menu organizado na página de resultados.

4. Resultados e Discussão:

Foram obtidos pontos positivos e negativos durante com o processo do WebCrawler. Com um extenso arquivo de configuração de palavras-chaves, fora obtido um bom resultado, sendo que na maioria das vezes a classificação dada ao site chegava bem próximo a realidade do site. Porém, da mesma forma que temos um resultado positivo na eficiência, temos uma grande demora na classificação de todos os links do site, pois o mesmo processo é feito várias vezes.

5. Dificuldades:

Uma das maiores dificuldades encontradas foi trabalhar com a versão do Python exigida para a codificação. Atualmente, a versão Python 2.x é mais utilizada que a versão 3.x, então, houve dificuldade para obter informações sobre a linguagem e encontrar bibliotecas compatíveis com a versão 3.x. Muitas bibliotecas que funcionam perfeitamente nas versões anteriores, não estão completamente modificadas para funcionar na versão 3.x, e as que estão modificadas, ainda pode apresentar alguns problemas.

O fato de ser uma versão relativamente nova faz com que poucos exemplos de funções/utilidades mais complexas, como Threads, sejam encontrados facilmente em buscas feitas em livros ou na internet, e em alguns casos é de difícil entendimento e há poucas pessoas que realmente conhecem a nova versão.

6. Conclusão

O Web Crawler pode demorar um bom tempo para que consiga classificar todos os links que estão dentro de um site pela quantidade de palavras-chaves cadastradas, porém, quanto maior a quantidade dessas palavras, melhor e mais precisa é essa classificação.

Com esse trabalho, foi possível entender como funciona a relação entre Python e o ambiente Web, além de entender como o funcionamento do CSS3 e os novos elementos do HTML5.

7. Referências.

PYTHON, documentação. Disponível em < http://www.python.org/doc/ >, acessado em 17/09/12.

W3SCHOOLS, html5. Disponível em <http://www.w3schools.com/html/html5_intro.asp >, acessado em 23/10/12.