PageRank

Caio de Moraes Braz e José Coelho de Pina Jr.
Bacharelado em Ciência da Computação
Instituto de Matemática e Estatística,
Universidade de São Paulo,
Rua do Matão 1010, 05508–900 SãoPaulo, SP
caiobraz@linux.ime.usp.br
coelho@ime.usp.br

1 de dezembro de 2011

Resumo

Classificar informações por relevância é um grande desafio, principalmente por se tratar de algo subjetivo, que depende do interesse do usuário, do nível de conhecimento relativo ao assunto, entre outros.

Um dos métodos para classificar páginas na Internet é o conhecido PageRank que atribui um valor (`rank`) a cada página, percorrendo sua estrutura de *links*.

Neste trabalho de formatura supervisionado estudamos o PageRank, suas implementações e aplicações.

Sumário

1 Introdução

1.1 Motivação

Classificar informações sempre foi algo importante. Quando pensarmos no contexto da Internet, essa classificação se torna altamente prioritária, tornando-a um excelente alvo de estudos de como resolver este problema, que possui várias dificuldades, como o problema de escala (a Internet é muito grande), problemas de qualidade de informação (muitas das páginas existentes tem conteúdos irrelevantes).

Logo, é muito interessante estudar métodos para classificar a importância relativa entre essas páginas e com isso conseguir distinguir os conteúdos mais relevantes em uma busca, por exemplo.

Um dos métodos existentes para classificar páginas na Internet é o **PageRank**, que classifica as páginas atribuindo um valor (que chamaremos de *rank*) a cada página, calculado a partir da estrutura de links destas páginas imersas na Internet.

1.2 Histórico

O PageRank foi desenvolvido na Universidade de Stanford por Sergey Brin e Lawrence "Larry"Page (daí o nome PageRank) como parte de um protótipo de um mecanismo de busca textual em páginas na Internet de nome *Google* [1], que usa de maneira crucial a estrutura de links das páginas.

Este protótipo é a base para o *Google* que conhecemos hoje. O endereço `google.stanford.edu` ainda existe, e hoje é o mecanismo de busca oficial da Universidade de Stanford.

1.3 Objetivos

Neste trabalho pretendemos estudar o PageRank, suas características, as dificuldades de sua implementação, suas aplicações, assim como produzir um pequeno *software* que aplica o PageRank para domínios específicos, como por exemplo, a rede interna do IME ou a rede Linux.

2 PageRank

Nesta seção, vamos detalhar vários aspectos do PageRank, desde sua modelagem, o algoritmo, a implementação e as aplicações onde pode ser utilizado.

2.1 Modelagem

Para o PageRank, necessitamos primeiro modelar a internet como um digrafo. As páginas web serão representadas por vértices, e seus links serão os arcos deste digrafo.

A ideia do PageRank consiste em usar este grafo da web para categorizar as páginas de acordo com a estrutura dos links, seguindo o seguinte modelo:

- "The Random Surfer"model: consiste em imaginar um usuário da web, que uma vez em uma página, vai escolher a próxima página aleatóriamente dentre os links disponíveis na página atual, repetindo indefinidamente.

Podemos olhar este grafo da web como uma Cadeia de Markov com probabilidades $1/N_j$ em cada ligação, onde L_v é o número de ligações que saem do vértice v

Logo, temos a matriz A que representa esta cadeia:

$$A_{i,j} = \begin{cases} 1/L_j, & \text{se existe um arco ligando } j \text{ a } i \\ 0, & \text{caso contrário} \end{cases}$$

Com isso, podemos dizer que o PageRank é uma distribuição estacionária desta cadeia, onde o valor (rank) de cada página é a probabilidade de estar nela após um número indefinido de mudanças de estado, começando em qualquer página.

2.2 Algoritmo

Agora que temos uma modelagem, podemos ver como é definido o PageRank. Em [2], temos uma primeira definição:

$$PR(u) = c \cdot \sum_{v \in B_u} \frac{PR(v)}{L_v}$$

Onde, B_u é o conjunto de páginas que aponta para u, L_v é o número de links que sai de v e finalmente c é uma constante de normalização.

Nesta definição recursiva já vemos bem a idéia de que a importância de uma página está diretamente ligada à importância das páginas que apontam para ela, porém, teremos um problema quando a cadeia possuir o que se chama de *Rank Sink*.

O *Rank Sink* é um pedaço do grafo no qual há um circuito de onde não é mais possível sair, isso impede o calculo do PageRank de convergir para o resultado esperado.

Para resolver este problema, é introduzido um fator de amortecimento d que representa a probabilidade do usuário continuar clicando em links. Logo, a ação na qual o usuário eventualmente "cansa"de avançar em links e decide acessar uma página qualquer tem uma probabilidade $(1 - d)$.

Assim sendo, a nova versão do PageRank fica:

$$PR(u) = c \cdot ((1-d) + d \cdot \sum_{v \in B_u} \frac{PR(v)}{L_v})$$

Alterando também a matriz A, que fica da forma:

$$A_{i,j} = \begin{cases} d/L_j, & \text{se existe um arco ligando } j \text{ a } i \\ (1-d)/(N-L_j), & \text{caso contrário} \end{cases}$$

Onde N é o número total de vértices.

Garantindo assim que todos as páginas tenham sempre uma chance de ser acessada, fazendo com que o algoritmo funcione.

Com isso, podemos dizer que o PageRank é uma distribuição estacionária da cadeia de markov, onde o valor (rank) de cada página é a probabilidade de estar nela após um número indefinido de mudanças de estado, começando em qualquer página.

Logo, o PageRank é o autovetor da matriz A associado ao autovalor 1, isto é:

$$A \cdot PR = 1 \cdot PR$$

3 Software

Nesta seção falaremos sobre a implementação do software para calcular o PageRank para um domínio específico (no caso, o domínio do IME-USP)

3.1 Ideia

A ideia foi desenvolver um software que dado um domínio, calculasse o PageRank das páginas relativo a este dominio.

Para isso, dividimos o software em 4 partes principais:

- Crawler
- Parser
- Hash e Estrutura de dados
- PageRank

3.2 Crawler

O crawler é a parte do software que percorre a internet, no nosso caso, pegando o código HTML da página e enviando-o para o parser.

3.3 Parser

O parser é o responsável por pegar um código HTML e criar uma lista de links existentes na página. Com isso ele atualiza a fila de páginas que o crawler deve percorrer e tambem envia esta lista para a construção da estrutura de dados que representa o grafo da web

3.4 Hash e Estrutura de Dados

Uma vez com a lista de páginas em mãos, devemos montar a estrutura de dados para que seja possivel o cáclulo do PageRank, tendo o cuidado de não repetir paginas na estrutura, que é o papel da tabela de hash.

3.5 PageRank

Finalmente, a parte que uma vez a estrutura de dados pronta, calcula o PageRank para as páginas.

4 Resultados

A implementação do software está bem modularizada, de modo que as partes fazem seu papel, porém ainda não foi possivel terminar de integra-las de forma a ter um programa funcional.

5 Análise Subjetiva

Uma análise pessoal sobre a relação entre este trabalho com o curso de Ciência da computação em si.

- **Desafios e frustrações encontrados**

  Na parte do trabalho em si, acredito que o maior desafio tenha sido a parte do desenvolvimento do software, em particular o crawler e o parser. O primeiro devido ao fato de que é relativamente complicado testa-lo. O segundo pelo fato de um parser ter inúmeros detalhes que precisam ser levados em conta quando queremos encontrar todos os links em uma página.

  Mas em especial, a maior dificuldade durante este trabalho foi externa, vários problemas, como uma mudança tardia sobre a ideia do trabalho, assim como alguns problemas pessoais, que fizeram com que meu foco e atenção não estivessem neste trabalho durantes vários momentos, e isso foi altamente frustrante pra mim, pois não consegui fazer tudo que poderia ter feito neste trabalho.

O software não está funcional do jeito que deveria, as partes separadas funcionam, mas ainda tem graves problemas na integração.

- **Lista das disciplinas cursadas no BCC mais relevantes para o trabalho**
  - MAC0110 - MAC0122 - MAC0323
    Disciplinas que são a base dos conceitos de programação, todo código escrito neste trabalho, utiliza de alguma maneira vários tópicos abordados inicialmentes nestas disciplinas.
  - MAE0228 - Introdução à Probabilidade e Processos Estocásticos
    Acredito que seja a disciplina mais relevante neste trabalho, dado que é nela que aprendemos sobre Cadeias de Markov, passeio aleatório e estado estacionário, que são modelos importantes para entender como funciona o PageRank
  - MAT0139 - Álgebra Linear para Computação
    Nesta disciplina, um dos tópicos é sobre autovalores e autovetores, que tem uma boa importância neste trabalho, pois o PageRank, sendo uma distribuição estacionária, é um autovetor.
  - MAC0300 - Métodos Numéricos da Álgebra Linear
    Uma vez tendo o conhecimento sobre autovalores e autovetores, nesta disciplina vemos uma técnica para calcular autovetores de uma matriz eficientemente, o método da potência.

- **Futuro do Trabalho**

  Muitas coisas a serem feitas, desde a integração final entre os módulos, possibilidade de utilizar outros dominios, melhorias no parser.

  Além disso, gostaria de estudar mais as consequencias que mudanças no grafo da web geram no PageRank, como por exemplo, o que a remoção de uma página importante faz com o PageRank? Ou encontrar o melhor "lugar"para inserir uma página de modo que ela tenha um rank alto.

- **Agradecimentos**

  Em primeiro lugar, tenho que agradecer à minha família que sempre me apoiou, não importando a situação.

  Agradeço também ao Prof. Coelho, que me orientou durante este trabalho e também foi sempre receptivo e motivador. Sem ele, acho que teria desistido do trabalho devido aos problemas que tive.

  Por último, agradeço ao meus amigos, que estão sempre dispostos a ouvir os problemas e compartilhar ideias e soluções para eles.

Referências

[1] Sergey Brin e Lawrence Page, The anatomy of a large-scale hypertextual web search engine, *Proceedings of the Seventh International World Wide Web Conference*, vol. 30, April 1998, `http://infolab.stanford.edu/~backrub/google.html`, pp. 107–117.

[2] Rajeev Motwani Lawrence Page, Sergey Brin e Terry Winograd, *The pagerank citation ranking: Bringing order to the web*, Tech. report, Stanford University, 1999.