Busca parametrizada para a recuperação ranqueada na Web

Fábio Eduardo Kaspar, Igor Canko Minotto, Ricardo Oliveira Teles
Orientador: Prof. Dr. Joao Eduardo Ferreira
Instituto de Matemática e Estatística da Universidade de São Paulo

Introdução

Com o surgimento da Web, houve um aumento significativo na quantidade de informações publicadas. No entanto, seria inútil manter um grande volume de dados sem a capacidade de priorizar respostas para o usuário com eficiência. Neste contexto, a Recuperação de Informação (RI) vem se tornando uma importante área de pesquisa para acesso a grandes coleções de documentos.

> *"Recuperação de Informação (RI) é encontrar material (geralmente documentos) de uma natureza não estruturada (geralmente textos) que satisfaça uma informação necessária dentro de grandes coleções (geralmente armazenadas em computadores)" [1].*

Possibilitar o ranqueamento dos resultados segundo uma ordem de relevância é uma estratégia para facilitar a recuperação das informações desejadas pelo usuário. Para alguns autores, a relevância de um documento é subjetiva ao usuário, pois é uma percepção dele sobre a proximidade da informação fornecida e a que ele necessita. Um dos principais desafios da área é compreender os objetos de interesse do usuário com base nessa subjetividade. Expressar consultas com parâmetros traz possibilidades de refinar essa ideia, pois permite ao usuário comunicar com maior precisão o que ele deseja encontrar.

São práticas comuns dos sistemas de RI utilizar filtros ou buscas sem parâmetros, o que limita a verbalização do usuário. Nesse caso, ele só visualiza um conjunto de resultados por vez, possivelmente realizando várias buscas até encontrar algo satisfatório, o que pode ser frustrante.

Como alternativa, desenvolvemos um sistema de RI com buscas parametrizadas, a aplicação LookingFor.

Objetivos

- Estudo dos conceitos fundamentais de RI.
- Adaptação do método de ponderação de parâmetros.
- Implementação de um experimento (LookingFor) como caso de uso.

Estrutura de busca

A busca tem por objetivo encontrar documentos nos quais determinados termos estão presentes. Para isso, é necessário utilizar uma estrutura de dados conveniente. Índices invertidos são mais comuns, por serem compactos.

LookingFor

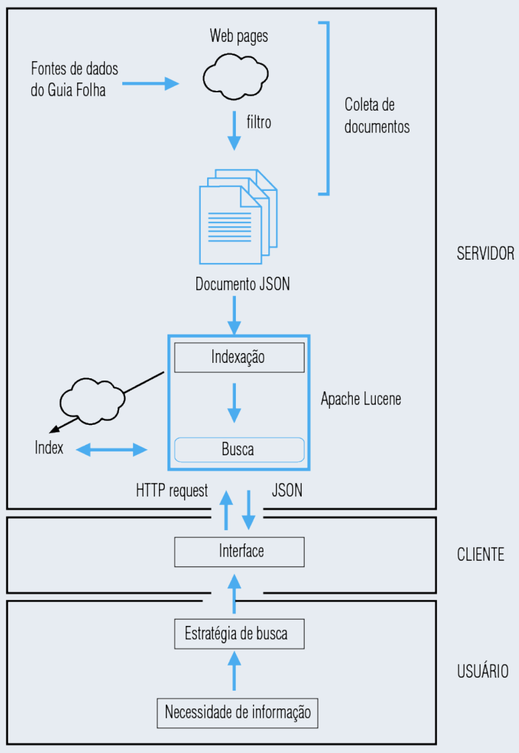

Figure 1: Esquema do aplicativo

O **LookingFor** é um aplicativo web que auxília o usuário na busca por restaurantes de acordo com 3 parâmetros: distância, preço e qualidade.

SERVIDOR

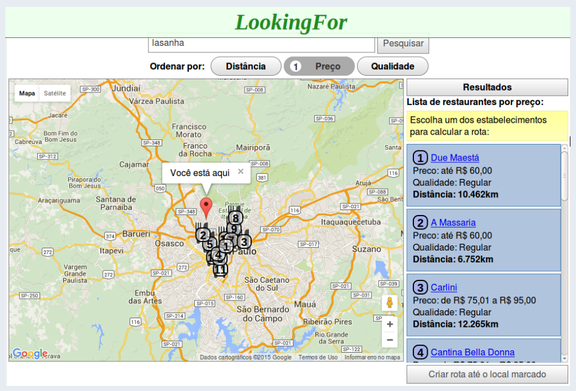

Figure 2: Interface do LookingFor

O resultado da busca é uma lista de documentos composta por referências a cada termo, que são recuperadas por meio dessa estrutura.

Para ranquear a lista retornada, algumas estatísticas são necessárias como a frequência e a raridade. A frequência de um termo em um documento ($tf_{t,d}$) pode indicar textos mais relevantes para a consulta.

Em uma busca com vários termos, um deles pode ser mais frequente que os demais e, consequentemente, ter uma influência maior nos resultados. Os autores definem a frequência de documentos df_t como o número de documentos nos quais esse termo está presente. Por favorecer termos muito frequentes, define-se também a frequência inversa do termo, idf_t. Sendo N o número de documentos na coleção, normaliza-se a df_t e atenua-se o seu crescimento com a função logarítmica:

$$idf_t = log\frac{N}{df_t} \qquad (1)$$

Em outras palavras, $tf_{t,d}$ é a abundância do termo em um documento e idf_t é a raridade do termo na coleção. Unindo esses dois conceitos, os autores definem

$$tf\text{-}idf_{t,d} = tf_{t,d} \cdot idf_t \qquad (2)$$

que é um dos atributos utilizados para ranquear um documento. Por exemplo, ao buscar "aracnofobia aranha" na Web, o termo "aracnofobia" é o mais raro. Logo, um documento com várias ocorrências desse termo deve ter uma colocação melhor que outro com ocorrências apenas de "aranha".

Por fim, define-se a pontuação de um documento d dada uma consulta q por

$$Score(q,d) = \sum_{t \in q} tf\text{-}idf_{t,d} \qquad (3)$$

Em alguns sistemas, as informações necessárias podem estar espalhadas em várias zonas de um documento. Por exemplo, um usuário pode procurar termos em um título ou em um resumo. Um peso p_i é atribuído para cada uma das M zonas disponíveis. Uma nova pontuação s_i é dada para cada zona de um documento e sua pontuação total torna-se:

$$Score(q,d) = \sum_{n=i}^{M} p_i s_i \qquad (4)$$

Este é um caso genérico do parágrafo anterior, que só possuía uma zona. Ou seja, s_i pode ser calculada como na equação 3.

Existem outras informações que podem ser relevantes para a busca, mas que não se encontram nas zonas de um documento. Por exemplo, pode-se associar uma posição geográfica a um documento. Para armazenar essas informações, são criados arquivos de metadados [2]. Assim como na equação 4, podemos utilizá-los para atribuir uma pontuação s_i para cada parâmetro.

Proposta

O LookingFor é o aplicativo desenvolvido neste trabalho. Sua estrutura pode ser vista na figura 1. Utilizou-se a biblioteca Apache Lucene [3] para indexar a coleção de documentos e realizar buscas.

No nosso caso de uso, a coleção foi gerada a partir da listagem dos 954 restaurantes do Guia da Folha para o estado de São Paulo [4].

As páginas Web foram guardadas em disco. Depois, fez-se uma filtragem, extraindo somente os dados necessários para a busca parametrizada do caso de uso, como o preço, a localização e a qualidade dos estabelecimentos. Essas informações foram armazenadas no formato JSON, compondo a coleção de documentos e metadados que serviram de base para a indexação e a busca.

Conclusão

Ao buscar por informações em grandes coleções é conveniente realizar um pré-processamento (indexação) dos documentos, a fim de melhorar a eficiência com que a busca é feita. Outro ponto chave de RI é o cálculo do $tf\text{-}idf_{t,d}$. Ele é a base do ranqueamento de documentos, sendo adaptado conforme cada aplicação.

Por fim, um outro grande desafio de RI é analisar se o método utilizado na busca parametrizada foi satisfatório, uma vez que a relevância de um documento é subjetiva ao usuário, conforme já mencionado.

Trabalhos Futuros

- Monitorar o comportamento dos usuários a fim de validar a qualidade da busca.
- Manter uma estatística dos pesos atribuídos aos parâmetros para a avaliar sua contribuição à experiência do usuário.
- Incluir parâmetros de diversas naturezas, tais como idade, gênero, entre outros.
- Estender o trabalho para outras categorias de coleções, por exemplo livros de uma biblioteca digital.

Referências

[1] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA, 2008.

[2] Ana Maria de Carvalho Moura, Genelice da Costa Pereira, and María Luiza Machado Campos. A metadata approach to manage and organize electronic documents and collections on the web. *Journal of the Brazilian Computer Society*, 8:16 – 31, 07 2002.

[3] Apache Lucene. `https://web.archive.org/web/20151106173103/https://lucene.apache.org/core/`. Accessed: 2015-11-13.

[4] Guia Folha. `https://web.archive.org/web/20151113125208/http://guia1.folha.com.br/busca/restau%20rantes/?sr=1`. Accessed: 2015-11-13.

Contato

- Fábio Eduardo Kaspar
  fabio.kaspar@usp.br
- Igor Canko Minotto
  igor.minotto@usp.br
- Ricardo Oliveira Teles
  ricardo.oliveira.teles@usp.br