**PRÓXIMA ETAPA**

**DATA: até 20/04**

**Criar a *home page* do seu trabalho**: <http://www.linux.ime.usp.br/~seulogin/mac0499/>

**Proposta de trabalho**

A proposta deve incluir uma descrição clara do que se pretende fazer no trabalho. Para tanto, deve apresentar uma contextualização do tema ou problema a ser tratado, incluindo as motivações e justificativas para o estudo do tema, o objeto/problema específico a ser estudado, objetivos do estudo, as principais atividades previstas para atingir o obetivo, juntamente com um cronograma aproximado. Se o desenvolvimento do trabalho já estiver em andamento, pode incluir também uma descrição dos avanços já obtidos.

Converse com o seu supervisor para dimensionar o escopo, definir objetivos e metas, e planejar as principais atividades do trabalho pretendido.

Disponibilizar na *home page* do seu trabalho e enviar email para responsável e supervisor

**TÓPICOS DOS CAPÍTULOS**

**Capítulo 1**

Modelo de rec. booleana: quais termos aparecem em quais documentos da minha coleção.

Duas abordagens:

1. Matriz de incidência termo-documento: os indices das linhas são os termos e os indices das colunas são documentos. Problema: matriz esparsa.
2. Índice invertido: estrutura de dados muito melhor que a anterior e muito utilizada. Permite otimizações de consulta, como o numero de documentos da lista já é gravado na cabeça, para aquele termo. Ordenação alfabética dos termos do dicionário. Ordenação numérica dos ids dos documentos.

Este modelo admite uma linguagem de consulta precisa, na qual os termos são valores booleanos e podem ser conectados por operadores da lógica (AND, OR, …).

Também foi um modelo muito utilizado pelos provedores de informação até a década de 90, com o aparecimento da Web.

É um modelo muito limitado, tendo em vista as necessidades dos usuários.

Já no modelo de rec. ranqueada, a consulta submetida pelo usuário assume formato livre e o sistema deve responder a essas consultas.

Contamos também com algoritmos de busca de intersecção e sua variante otimizada, que começa o processamento com a menor lista de postagens.

(Fábio)

**Capitulo 6:**

Precisamos ordenar ou classificar os documentos que foram retornados pela consulta.

Queremos atribuir um peso para o par (consulta, documento).

**Índice de zona e índice paramétrico:**

Documentos grosseiramente são pensados como conjuntos de termos, mas a maioria deles (documentos digitais) incluem metadados reconhecíveis pelo computador, tais como autor, título e data de publicação. Estes metadados geralmente são campos do dicionário, como data de criaçao, formato, autor, título.

Os campos de dicionário podem ser usados para definir consultas paramétricas (documentos com uma determinada data, ou um determinado autor, etc) e se os valores forem ordenáveis, podemos usar estruturas apropriadas de manipulação, como árvores-B.

Zonas diferem de campos por poderem assumir texto arbitrário e de tamanho ilimitado, ao passo que os campos são de dimensão menor. Por exemplo, resumos, titulos e lista de autores são tratados como zonas.

Enquanto o dicionário do índice paramétrico assume um vocabulario fixo, o dicionário do índice de zona é mais abrangente e deve estruturar vocabulários que derivem do texto da zona.

As zonas podem ser codificadas no dicionário ou nas postagens.

Até agora, assumimos que o peso depende se um termo de uma consulta está ou não presente na zona de um documento.

Esquema de ponderaçao 1 => tf(t,d): nº de ocorrencias do termo t no documento d;

modelo *Bag of Words* (a ordem dos termos é ignorado e o que interessa é apenas o numero de ocorrencias do termo no documento).

Stop-words: palavras que não serão indexadas, pois não contribuem para a pontuaçao;

Não são todas as palavras igualmente importante em um documento.

df(t): número de documentos na coleção que contém t

idf: mede pontuação de um termo. idf(t) = log(N/df(t)). Cresce com N e com a raridade do termo.

tf-idf(t,d) = tf(t,d) \* idf(t) => aumenta a pontuação do documento se o termo aparece muitas vezes; aumenta se o termo aparecer em poucos documentos.

Por fim, podemos atribuir uma pontuação para um documento em função de uma consulta, como sendo a somatória do tf-idf(t,d) de cada termo t da consulta no documento d.

score(q,d) = SUM[ tf-idf(t,d) ]

t em d

wf(t,d) = 1 + log(tf(t,d)), se tf,td >0 ou 0 c.c

wf-idf t,d = wf t,d × idf t .

**Capítulo 6**

Atribui uma pontuação a um par (consulta, documento)

**6.1** **Parametric and zone indexes**

Metadata -> formas especificas de dados sobre um documento (autor, título, data de publicação, data de criação e fomato do documento)

fields -> os possiveis valores para os campos deveriam ser finitos

parametric index -> texto com valores específicos (ex: datas)

zone -> texto livre arbitrário (ex: títulos de documentos)

**6.1.1 Weighted zone scoring (ranked Boolean retrieval)**



si pertence ao conjunto {0,1} e gi é calculado através de machine learning

**6.1.2 Learning weights**

Como é calculado gi? através de machine learning

**6.1.3 The optimal weight g**

Qual é valor ótimo de gi?

**6.2 Term frequency and weighting**

mochila de palavras(bag of words) = a ordem exata das palavras em um documento é ignorada, mas o número de ocorrências do termo é importante

**6.2.1 Inverse document frequency**

tft,d (frequência do termo) = número de ocorrências do termo t em um documento d

cf = o número de ocorrências de um termo na coleção

df = o número de documentos na coleção que contém o termo

idft,d = log (N/df)

**6.3 The vector space model for scoring**

**6.4 Variant tf-idf functions**

**PROPOSTA (Oficial)**

**Alunos:** Fábio Eduardo Kaspar

Igor Canko Minotto

Ricardo Oliveira Teles

**Orientador**: João Eduardo Ferreira

**Co-orientador**: André Casimiro

**Proposta:** Recuperação de Informação para dados não-relacionais na Web

**Objetivo do trabalho:**  conhecer os principais fundamentos de recuperação de informação e aplicá-las em um experimento real.

**Como será trabalhado:** O trabalho é dividido em 3 partes:

1) Estudo dos conceitos fundamentais de Recuperação de informação, no qual geraremos um texto síntese dos principais conceitos do tema proposto;

2) Adaptação ou desenvolvimento de um algoritmo já conhecido na literatura para atender uma lacuna na recuperação de informação;

3) E por último, projeto e implementação de um experimento com as devidas análises.

**PROPOSTA (Rascunho)**

**Alunos:** Fábio Eduardo Kaspar NUsp: 7991166

Ricardo Oliveira Teles NUsp: 7991444

**Orientador**: João Eduardo Ferreira

**Co-orientador**: André

**Proposta:** Recuperação de Informação para dados não-relacionais na Web **(Tema ?)**

**Como será trabalhado:**

O trabalho consistirá em 3 partes: estudos dos conceitos fundamentais de Recuperação de informação, desenvolvimento de uma ferramenta com suporte da implementação de um algoritmo conhecido, e por último, sua aplicação e análise em um problema real, através de experimentos.

O objetivo do trabalho será a recuperação da informação com base em alguma informação de interesse anexada, como perfil do usuário, data, hora, local, etc.