ória¡6¿59248O Modelo de Espaço Vetorial

Sistemas de Recuperação de Informação

https://github.com/fccoelho/curso-IRI

IRI 6: Scores, Ponderação de Termos e Modelos de Espaço Vetorial

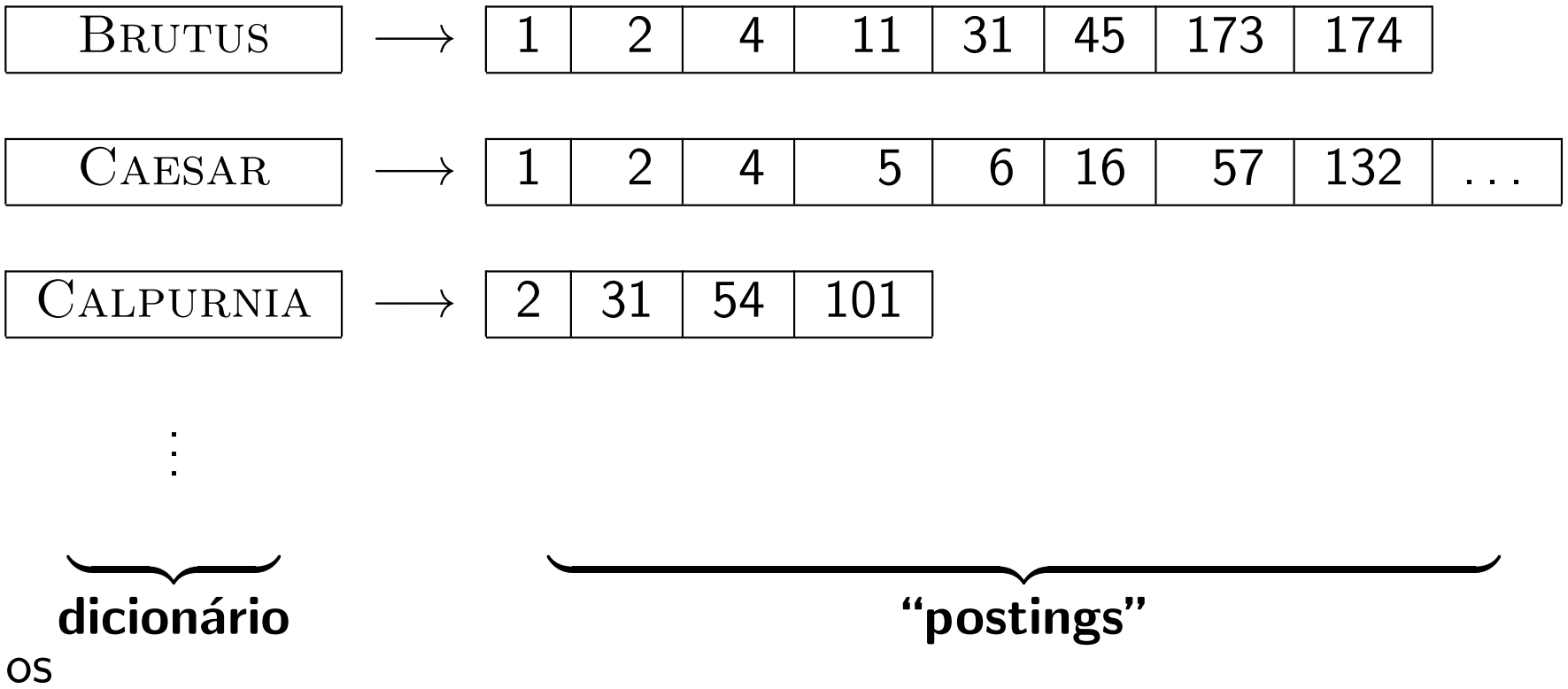
Flávio Codeço Coelho

Escola de Matemática Aplicada, Fundação Getúlio Vargas

Sumário da Aula

Índice invertido

Para cada termo t, armazenamos uma lista de documentos que contém t.



os

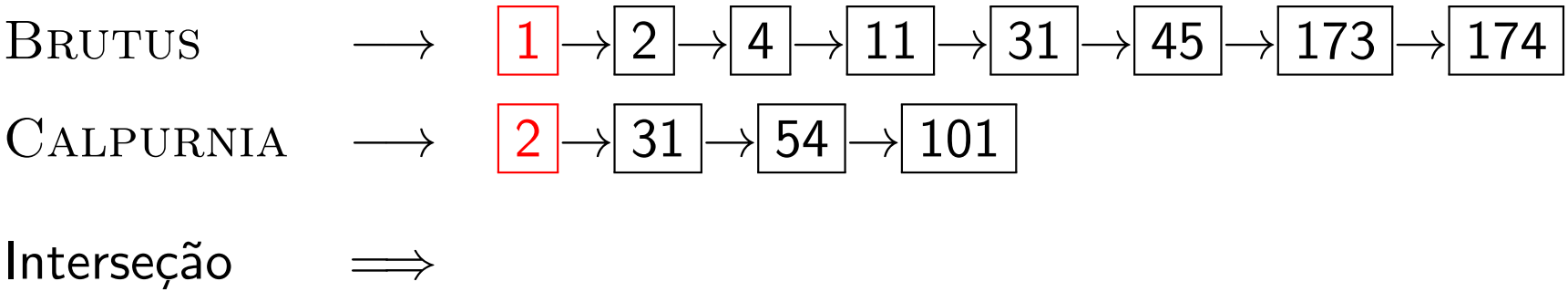
Interseção de duas listas de postings

BRUTUS $\longrightarrow$ 1 → 2 → 4 → 11 → 31 → 45 → 173 → 174

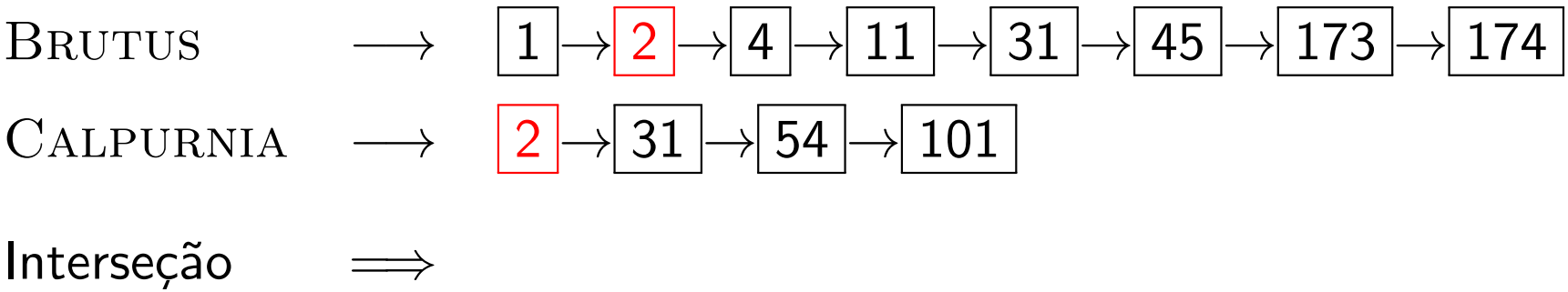
CALPURNIA $\longrightarrow$ 2 → 31 → 54 → 101

Interseção $\Longrightarrow$

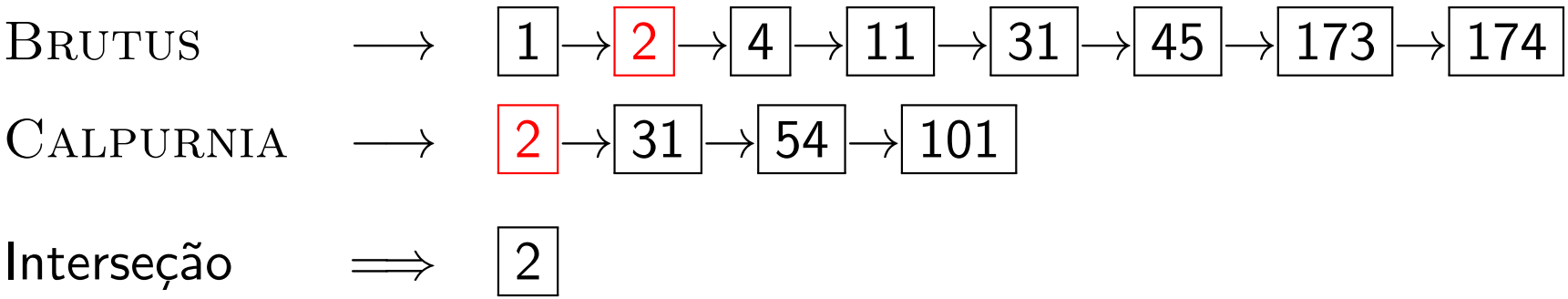
Interseção de duas listas de postings



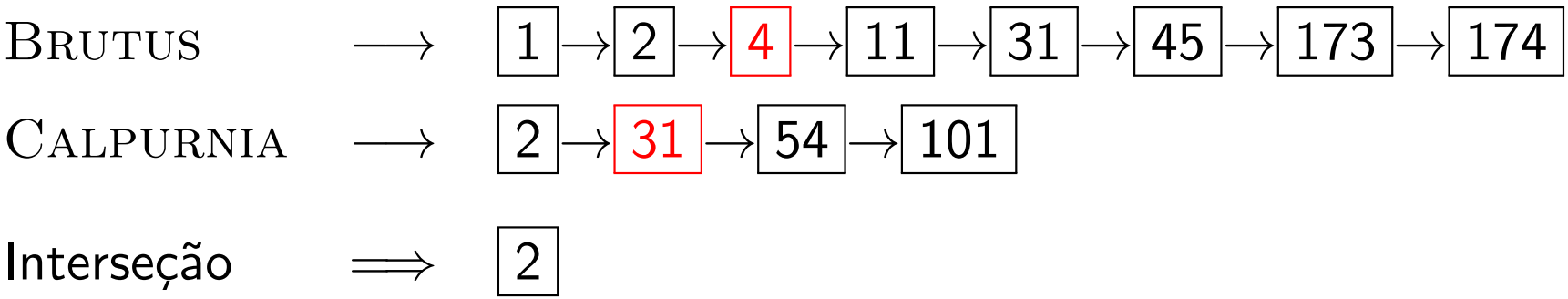
Interseção de duas listas de postings



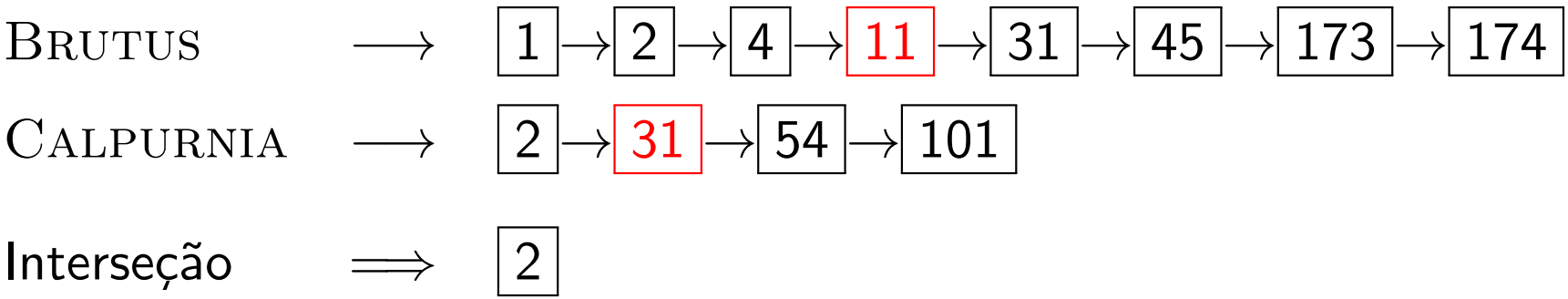
Interseção de duas listas de postings



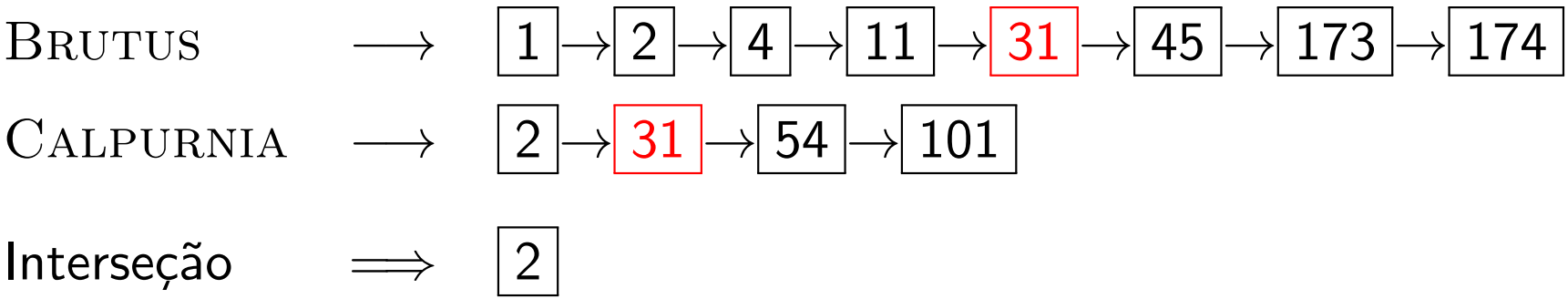
Interseção de duas listas de postings



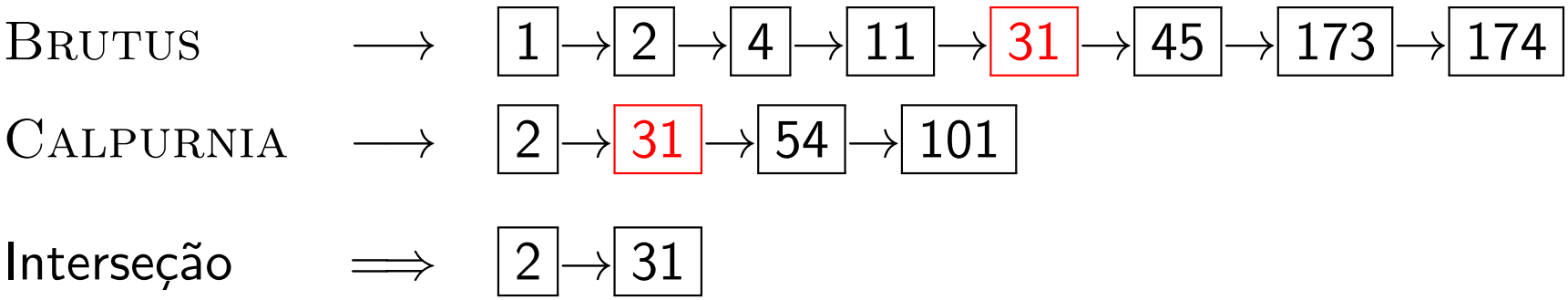
Interseção de duas listas de postings



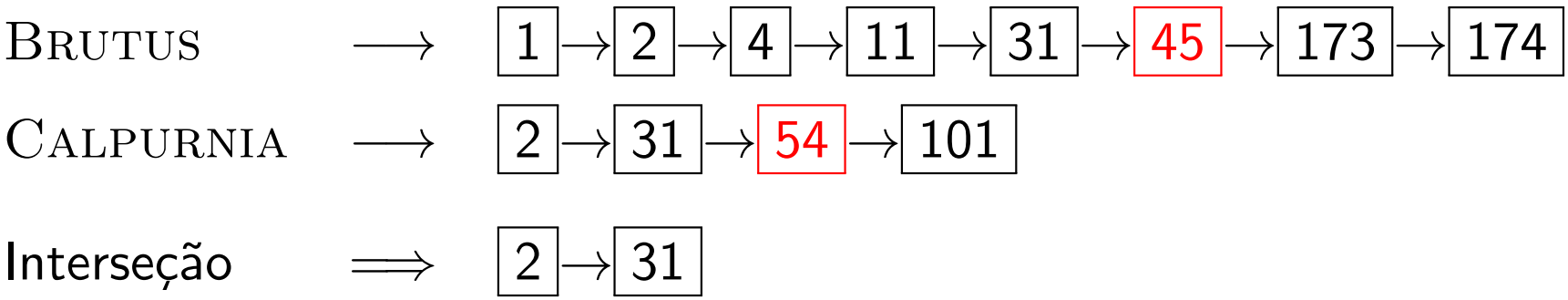
Interseção de duas listas de postings



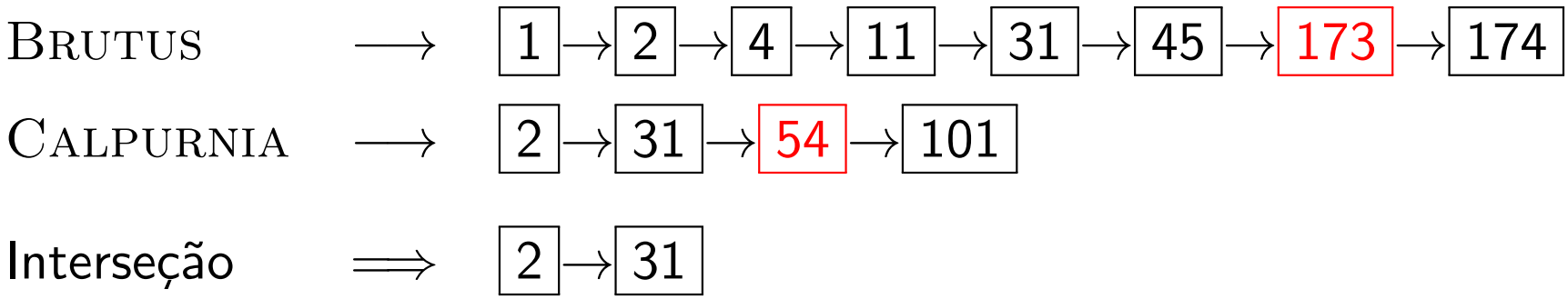
Interseção de duas listas de postings



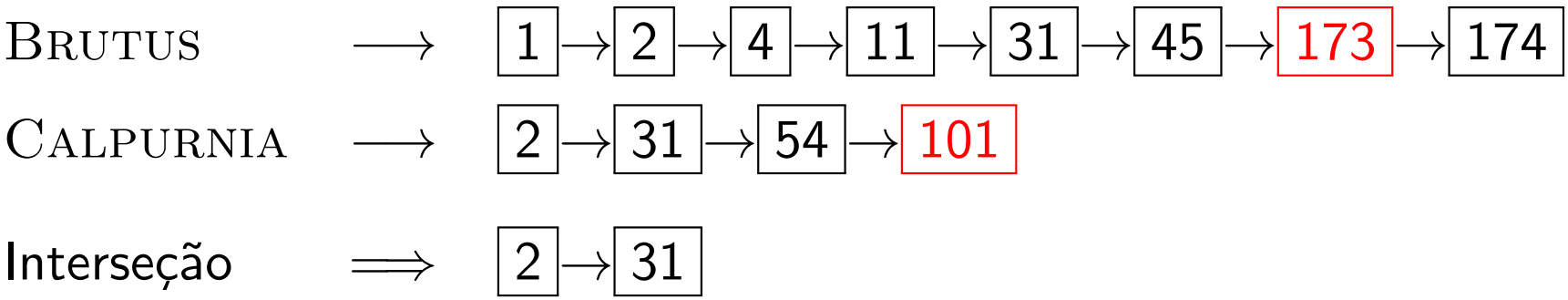
Interseção de duas listas de postings



Interseção de duas listas de postings



Interseção de duas listas de postings



Construindo um índice invertido: Ordenando postings

termo	docID
I	1
did	1
enact	1
julius	1
caesar	1
I	1
was	1
killed	1
i'	1
the	1
capitol	1
brutus	1
killed	1
me	1
so	2
let	2
it	2
be	2
with	2
caesar	2
the	2
noble	2
brutus	2
hath	2
told	2
you	2
caesar	2
was	2
ambitious	2

$\Longrightarrow$

term	docID
ambitious	2
be	2
brutus	1
brutus	2
capitol	1
caesar	1
caesar	2
caesar	2
did	1
enact	1
hath	1
I	1
I	1
i'	1
it	2
julius	1
killed	1
killed	1
let	2
me	1
noble	2
so	2
the	1
the	2
told	2
you	2
was	1
was	2
with	2

O Google usa o modelo Booleano?

- No Google, a interpretação default de uma consulta $[w_1\ w_2 \ldots w_n]$ é w_1 AND w_2 AND ... AND w_n
- Casos onde há "hits" mas não contêm um dos w_i:
  - Texto âncora
  - Página contém variante de w_i (morfologia, correção ortográfica, sinônimo)
  - Consultas longas (n grande)
  - Expressão booleana gera poucos "hits"
- Recuperação Booleana simples vs. Rankeamento dos resultados
  - Recuperação Booleana simples retorna os documentos sem um ordenamento significativo.
  - O Google (e a maioria das máquinas booleanas bem projetadas) rankeiam os resultados – os melhores "hits" hits (de acordo com alguma estatísticade relevância) aparecem mais altos do que os "hits" piores.

Distinção entre *Tipos* e *Tokens*

- Token – Instância de palavra ou termo ocorrendo em um documento
- Tipo – Uma classe de equivalência de tokens
- *In June, the dog likes to chase the cat in the barn.*
- 12 tokens, 9 tipos

Problemas com Tokenização

- Quais são os delimitadores? espaço? apóstrofe? hífen?
- Para cada um destes: às vezes eles delimitam, às vezes não.
- Muitas líguas não possuem espaços! (por ex., Chinês)
- Não há espaços e palavras compostas em Holandês, Alemão e Sueco (*Lebensversicherungsgesellschaftsangestellter*)

Problemas com Classes de Equivalência

- Um termo é uma classe de equivalencia de tokens.
- Como definimos classes de equivalência?
- Numeros (3/20/91 vs. 20/3/91)
- Capitalização
- "Stemming", Porter stemmer
- Análise morfológica : inflecional vs. derivacional
- Classes de equivalência para múltiplas línguas?
  - Morfologias mais complexas:
  - Finlandês: um único verbo pode ter 12,000 formas diferentes!!
  - Acentos, etc.

Índices Posicionais

- Listas de "postings" em um índice não posicional: cada "posting" é apenas um docID
- Listas de "postings" em um índice posicional: cada "posting" é um docID e uma lista de posições
- Exemplo: *"to$_1$ be$_2$ or$_3$ not$_4$ to$_5$ be$_6$"*

TO, 993427:
⟨ 1: ⟨7, 18, 33, 72, 86, 231⟩;
2: ⟨1, 17, 74, 222, 255⟩;
4: ⟨8, 16, 190, 429, 433⟩;
5: ⟨363, 367⟩;
7: ⟨13, 23, 191⟩; . . . ⟩

BE, 178239:
⟨ 1: ⟨17, 25⟩;
4: ⟨17, 191, 291, 430, 434⟩;
5: ⟨14, 19, 101⟩; . . . ⟩

Documento 4 não é um resultado!

Índices Posicionais

- Com um índice posicional, podemos responder
  - Consultas por frases
  - Consultas por proximidade

Moral da estória de hoje

Moral da estória de hoje

- Rankeamento dos resultados da busca: por que é importante (em comparação com apresentar um conjunto desordenado de resultados Booleanos)

Moral da estória de hoje

- Rankeamento dos resultados da busca: por que é importante (em comparação com apresentar um conjunto desordenado de resultados Booleanos)
- Frequência do termo: Ingrediente chave do ranqueamento

Moral da estória de hoje

- Rankeamento dos resultados da busca: por que é importante (em comparação com apresentar um conjunto desordenado de resultados Booleanos)
- Frequência do termo: Ingrediente chave do ranqueamento
- Rankeamento por Tf-idf: Melhor rankeamento tradicional.

Moral da estória de hoje

- Rankeamento dos resultados da busca: por que é importante (em comparação com apresentar um conjunto desordenado de resultados Booleanos)
- Frequência do termo: Ingrediente chave do ranqueamento
- Rankeamento por Tf-idf: Melhor rankeamento tradicional.
- Modelo de espaço vetorial: Um dos mais importantes modelos para recuperação de informação, juntamente com o modelo Booleano e o probabiliístico

Recuperação Rankeada

Recuperação Rankeada

- Até agora, nossas consultas têm sido Booleanas.

Recuperação Rankeada

- Até agora, nossas consultas têm sido Booleanas.
  - Documentos possuem ou não os termos.

Recuperação Rankeada

- Até agora, nossas consultas têm sido Booleanas.
  - Documentos possuem ou não os termos.
- Bom para usuários especialistas com um conhecimento preciso de suas necessidades e da coleção.

Recuperação Rankeada

- Até agora, nossas consultas têm sido Booleanas.
  - Documentos possuem ou não os termos.
- Bom para usuários especialistas com um conhecimento preciso de suas necessidades e da coleção.
- Também é bom para aplicações: Aplicações podem consumir milhares de resultados.

Recuperação Rankeada

- Até agora, nossas consultas têm sido Booleanas.
  - Documentos possuem ou não os termos.
- Bom para usuários especialistas com um conhecimento preciso de suas necessidades e da coleção.
- Também é bom para aplicações: Aplicações podem consumir milhares de resultados.
- Não é bom para a maioria dos usuários

Recuperação Rankeada

- Até agora, nossas consultas têm sido Booleanas.
  - Documentos possuem ou não os termos.
- Bom para usuários especialistas com um conhecimento preciso de suas necessidades e da coleção.
- Também é bom para aplicações: Aplicações podem consumir milhares de resultados.
- Não é bom para a maioria dos usuários
- A maioria dos usuários não é capaz de escrever consultas booleanas. . .

Recuperação Rankeada

- Até agora, nossas consultas têm sido Booleanas.
  - Documentos possuem ou não os termos.
- Bom para usuários especialistas com um conhecimento preciso de suas necessidades e da coleção.
- Também é bom para aplicações: Aplicações podem consumir milhares de resultados.
- Não é bom para a maioria dos usuários
- A maioria dos usuários não é capaz de escrever consultas booleanas. . .
  - . . . ou até são, mas acham que dá muito trabalho.

Recuperação Rankeada

- Até agora, nossas consultas têm sido Booleanas.
  - Documentos possuem ou não os termos.
- Bom para usuários especialistas com um conhecimento preciso de suas necessidades e da coleção.
- Também é bom para aplicações: Aplicações podem consumir milhares de resultados.
- Não é bom para a maioria dos usuários
- A maioria dos usuários não é capaz de escrever consultas booleanas. . .
  - . . . ou até são, mas acham que dá muito trabalho.
- A maioria dos usuários não quer inspecionar milhares de resultados.

Recuperação Rankeada

- Até agora, nossas consultas têm sido Booleanas.
  - Documentos possuem ou não os termos.
- Bom para usuários especialistas com um conhecimento preciso de suas necessidades e da coleção.
- Também é bom para aplicações: Aplicações podem consumir milhares de resultados.
- Não é bom para a maioria dos usuários
- A maioria dos usuários não é capaz de escrever consultas booleanas. . .
  - . . . ou até são, mas acham que dá muito trabalho.
- A maioria dos usuários não quer inspecionar milhares de resultados.
- Isto é particularmente verdadeiro para buscas na Web.

Problema com a busca Booleana: Banquete ou fome

Problema com a busca Booleana: Banquete ou fome

- Consultas Booleanas frequentemente resultam em demasiados ou muito poucos resultados.

Problema com a busca Booleana: Banquete ou fome

- Consultas Booleanas frequentemente resultam em demasiados ou muito poucos resultados.
- Com Consultas Booleanas, é preciso muita habilidade para produzir uma consulta que retorne um número razoável de "hits".

Banquete ou Fome: Não é Problema para Busca Rankeada

Banquete ou Fome: Não é Problema para Busca Rankeada

- Com ranqueamento, grandes conjuntos de resultados não são um problema.

Banquete ou Fome: Não é Problema para Busca Rankeada

- Com ranqueamento, grandes conjuntos de resultados não são um problema.
- Basta mostrar os 10 melhores

Banquete ou Fome: Não é Problema para Busca Rankeada

- Com ranqueamento, grandes conjuntos de resultados não são um problema.
- Basta mostrar os 10 melhores
- Não sobrecarrega os usuários

Banquete ou Fome: Não é Problema para Busca Rankeada

- Com ranqueamento, grandes conjuntos de resultados não são um problema.
- Basta mostrar os 10 melhores
- Não sobrecarrega os usuários
- Premissa: O rankeamento funciona: Resultados mais relevantes são posicionados acima de menos relevantes.

Escores são a base da Busca Rankeada

Escores são a base da Busca Rankeada

- Queremos rankear documentos mais relevamtes acima de documentos menos relevantes.

Escores são a base da Busca Rankeada

- Queremos rankear documentos mais relevamtes acima de documentos menos relevantes.
- Como fazer isso para uma dada consulta?

Escores são a base da Busca Rankeada

- Queremos rankear documentos mais relevamtes acima de documentos menos relevantes.
- Como fazer isso para uma dada consulta?
- Dê um escore a cada para consulta-documento, no intervalo $[0, 1]$.

Escores são a base da Busca Rankeada

- Queremos rankear documentos mais relevamtes acima de documentos menos relevantes.
- Como fazer isso para uma dada consulta?
- Dê um escore a cada para consulta-documento, no intervalo $[0, 1]$.
- Este escore mede a qualidade da correspondência consulta-documento.

Escores de correspondência consulta-documento

- Como computar o escore de um par consulta-documento?

Escores de correspondência consulta-documento

- Como computar o escore de um par consulta-documento?
- Vamos começar com uma consulta de um termo.

Escores de correspondência consulta-documento

- Como computar o escore de um par consulta-documento?
- Vamos começar com uma consulta de um termo.
- Se o termo da consulta não ocorre no documento: Escore deve ser 0.

Escores de correspondência consulta-documento

- Como computar o escore de um par consulta-documento?
- Vamos começar com uma consulta de um termo.
- Se o termo da consulta não ocorre no documento: Escore deve ser 0.
- Quanto mais frequente o termo de consulta ocorre no documento, maior deve ser o escore.

Escores de correspondência consulta-documento

- Como computar o escore de um par consulta-documento?
- Vamos começar com uma consulta de um termo.
- Se o termo da consulta não ocorre no documento: Escore deve ser 0.
- Quanto mais frequente o termo de consulta ocorre no documento, maior deve ser o escore.
- Como fazer isso?

Tentativa 1: coeficiente Jaccard

Tentativa 1: coeficiente Jaccard

- Uma medida comumente utilizada para medir interseção de conjuntos

Tentativa 1: coeficiente Jaccard

- Uma medida comumente utilizada para medir interseção de conjuntos
- Sejam A e B dois conjuntos

Tentativa 1: coeficiente Jaccard

- Uma medida comumente utilizada para medir interseção de conjuntos
- Sejam A e B dois conjuntos
- Coeficiente de Jaccard:

$$\text{JACCARD}(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

$(A \neq \emptyset \text{ or } B \neq \emptyset)$

Tentativa 1: coeficiente Jaccard

- Uma medida comumente utilizada para medir interseção de conjuntos
- Sejam A e B dois conjuntos
- Coeficiente de Jaccard:

$$\textsc{jaccard}(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

$(A \neq \emptyset \text{ or } B \neq \emptyset)$

- $\textsc{jaccard}(A, A) = 1$

Tentativa 1: coeficiente Jaccard

- Uma medida comumente utilizada para medir interseção de conjuntos
- Sejam A e B dois conjuntos
- Coeficiente de Jaccard:

$$\textsc{jaccard}(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

$(A \neq \emptyset \text{ or } B \neq \emptyset)$

- $\textsc{jaccard}(A, A) = 1$
- $\textsc{jaccard}(A, B) = 0$ se $A \cap B = 0$

Tentativa 1: coeficiente Jaccard

- Uma medida comumente utilizada para medir interseção de conjuntos
- Sejam A e B dois conjuntos
- Coeficiente de Jaccard:

$$\textsc{jaccard}(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

  $(A \neq \emptyset \text{ or } B \neq \emptyset)$
- $\textsc{jaccard}(A, A) = 1$
- $\textsc{jaccard}(A, B) = 0$ se $A \cap B = 0$
- A e B não têm que ter o mesmo tamanho.

Tentativa 1: coeficiente Jaccard

- Uma medida comumente utilizada para medir interseção de conjuntos
- Sejam A e B dois conjuntos
- Coeficiente de Jaccard:

$$\textsc{jaccard}(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

  $(A \neq \emptyset \text{ or } B \neq \emptyset)$
- $\textsc{jaccard}(A, A) = 1$
- $\textsc{jaccard}(A, B) = 0$ se $A \cap B = 0$
- A e B não têm que ter o mesmo tamanho.
- Sempre assume valores entre 0 e 1.

Coeficiente de Jaccard: Exemplo

Coeficiente de Jaccard: Exemplo

- Qual o escorede correspondência consulta-documento dado pelo coef. de Jaccard para:

Coeficiente de Jaccard: Exemplo

- Qual o escorede correspondência consulta-documento dado pelo coef. de Jaccard para:
  - Consulta: “ides of March”

Coeficiente de Jaccard: Exemplo

- Qual o escorede correspondência consulta-documento dado pelo coef. de Jaccard para:
  - Consulta: "ides of March"
  - Documento "Caesar died in March"

Coeficiente de Jaccard: Exemplo

- Qual o escorede correspondência consulta-documento dado pelo coef. de Jaccard para:
  - Consulta: "ides of March"
  - Documento "Caesar died in March"
  - $\textsc{jaccard}(q, d) = 1/6$

O que há de errado com Jaccard?

O que há de errado com Jaccard?

- Não considera a frequência do termo

O que há de errado com Jaccard?

- Não considera a frequência do termo
- Termos raros geralmente são mais informativos que termos frequentes. Jaccard não considera esta informação.

O que há de errado com Jaccard?

- Não considera a frequência do termo
- Termos raros geralmente são mais informativos que termos frequentes. Jaccard não considera esta informação.
- Precisamos de uma maneira mais sofisticada de normalizar pelo comprimento do documento.

O que há de errado com Jaccard?

- Não considera a frequência do termo
- Termos raros geralmente são mais informativos que termos frequentes. Jaccard não considera esta informação.
- Precisamos de uma maneira mais sofisticada de normalizar pelo comprimento do documento.
- Mais tarde usaremos $|A \cap B|/\sqrt{|A \cup B|}$ (cosseno) ...

O que há de errado com Jaccard?

- Não considera a frequência do termo
- Termos raros geralmente são mais informativos que termos frequentes. Jaccard não considera esta informação.
- Precisamos de uma maneira mais sofisticada de normalizar pelo comprimento do documento.
- Mais tarde usaremos $|A \cap B|/\sqrt{|A \cup B|}$ (cosseno) ...
- ... ao invés de $|A \cap B|/|A \cup B|$ (Jaccard) para normalização de comprimento.

Matriz de Incidência Binária

	Anthony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth	...
Anthony	1	1	0	0	0	1	
Brutus	1	1	0	1	0	0	
Caesar	1	1	0	1	1	1	
Calpurnia	0	1	0	0	0	0	
Cleopatra	1	0	0	0	0	0	
mercy	1	0	1	1	1	1	
worser	1	0	1	1	1	0	
...							

Cada documento é representado como um vetor binário $\in \{0,1\}^{|V|}$.

Matriz de Incidência Binária

	Anthony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth	...
Anthony	1	1	0	0	0	1	
Brutus	1	1	0	1	0	0	
Caesar	1	1	0	1	1	1	
Calpurnia	0	1	0	0	0	0	
Cleopatra	1	0	0	0	0	0	
mercy	1	0	1	1	1	1	
worser	1	0	1	1	1	0	
...							

Cada documento é representado como um vetor binário $\in \{0,1\}^{|V|}$.

Matriz de Contagem

	Anthony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth	...
Anthony	157	73	0	0	0	1	
Brutus	4	157	0	2	0	0	
Caesar	232	227	0	2	1	0	
Calpurnia	0	10	0	0	0	0	
Cleopatra	57	0	0	0	0	0	
mercy	2	0	3	8	5	8	
worser	2	0	1	1	1	5	
...							

Cada Documento é agora representado como um vetor de contagem $\in \mathbb{N}^{|V|}$.

Matriz de Contagem

	Anthony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth	...
ANTHONY	157	73	0	0	0	1	
BRUTUS	4	157	0	2	0	0	
CAESAR	232	227	0	2	1	0	
CALPURNIA	0	10	0	0	0	0	
CLEOPATRA	57	0	0	0	0	0	
MERCY	2	0	3	8	5	8	
WORSER	2	0	1	1	1	5	
...							

Cada Documento é agora representado como um vetor de contagem $\in \mathbb{N}^{|V|}$.

Modelo do Saco de Palavras

Modelo do Saco de Palavras

- Não consideramos a ordem das palavras em um documento.

Modelo do Saco de Palavras

- Não consideramos a ordem das palavras em um documento.
- *John is quicker than Mary* e *Mary is quicker than John* são representados da mesma forma.

Modelo do Saco de Palavras

- Não consideramos a ordem das palavras em um documento.
- *John is quicker than Mary* e *Mary is quicker than John* são representados da mesma forma.
- Isto se chama um modelo de saco de palavras.

Modelo do Saco de Palavras

- Não consideramos a ordem das palavras em um documento.
- *John is quicker than Mary* e *Mary is quicker than John* são representados da mesma forma.
- Isto se chama um modelo de saco de palavras.
- De certa maneira é um passo atrás: O índice positional era capaz de distinguir estes dois documentos.

Modelo do Saco de Palavras

- Não consideramos a ordem das palavras em um documento.
- *John is quicker than Mary* e *Mary is quicker than John* são representados da mesma forma.
- Isto se chama um modelo de saco de palavras.
- De certa maneira é um passo atrás: O índice positional era capaz de distinguir estes dois documentos.
- Vamos ver como "recuperar" a informação positional mais tarde.

Modelo do Saco de Palavras

- Não consideramos a ordem das palavras em um documento.
- *John is quicker than Mary* e *Mary is quicker than John* são representados da mesma forma.
- Isto se chama um modelo de saco de palavras.
- De certa maneira é um passo atrás: O índice positional era capaz de distinguir estes dois documentos.
- Vamos ver como "recuperar" a informação positional mais tarde.
- Por agora: Modelo do saco de palavras

Frequência do termo **tf**

- TA frequência $\text{tf}_{t,d}$ do termo t no documento d é definida como o número de vezes que t ocorre em d.

Frequência do termo **tf**

- TA frequência $\text{tf}_{t,d}$ do termo t no documento d é definida como o número de vezes que t ocorre em d.
- Queremos usar tf ao comput escores de correspondência consulta-documento.

Frequência do termo **tf**

- TA frequência $\text{tf}_{t,d}$ do termo t no documento d é definida como o número de vezes que t ocorre em d.
- Queremos usar tf ao comput escores de correspondência consulta-documento.
- Mas como?

Frequência do termo **tf**

- TA frequência $\text{tf}_{t,d}$ do termo t no documento d é definida como o número de vezes que t ocorre em d.
- Queremos usar tf ao comput escores de correspondência consulta-documento.
- Mas como?
- Frequência absoluta dos tesmos não é o que queremos pois:

Frequência do termo **tf**

- TA frequência $\text{tf}_{t,d}$ do termo t no documento d é definida como o número de vezes que t ocorre em d.
- Queremos usar tf ao comput escores de correspondência consulta-documento.
- Mas como?
- Frequência absoluta dos tesmos não é o que queremos pois:
- Um documento com $\text{tf} = 10$ ocorrências do termo é mais relevante do que um documento com $\text{tf} = 1$ ocorrência do termo.

Frequência do termo **tf**

- TA frequência $\text{tf}_{t,d}$ do termo t no documento d é definida como o número de vezes que t ocorre em d.
- Queremos usar tf ao comput escores de correspondência consulta-documento.
- Mas como?
- Frequência absoluta dos tesmos não é o que queremos pois:
- Um documento com $\text{tf} = 10$ ocorrências do termo é mais relevante do que um documento com $\text{tf} = 1$ ocorrência do termo.
- Mas não 10 vezes mais relevante.

Frequência do termo **tf**

- TA frequência $\text{tf}_{t,d}$ do termo t no documento d é definida como o número de vezes que t ocorre em d.
- Queremos usar tf ao comput escores de correspondência consulta-documento.
- Mas como?
- Frequência absoluta dos tesmos não é o que queremos pois:
- Um documento com $\text{tf} = 10$ ocorrências do termo é mais relevante do que um documento com $\text{tf} = 1$ ocorrência do termo.
- Mas não 10 vezes mais relevante.
- A Relevância não aumenta em proporção com a frequência do termo.

Frequência do termo **tf**

- TA frequência $\text{tf}_{t,d}$ do termo t no documento d é definida como o número de vezes que t ocorre em d.
- Queremos usar tf ao comput escores de correspondência consulta-documento.
- Mas como?
- Frequência absoluta dos tesmos não é o que queremos pois:
- Um documento com $\text{tf} = 10$ ocorrências do termo é mais relevante do que um documento com $\text{tf} = 1$ ocorrência do termo.
- Mas não 10 vezes mais relevante.
- A Relevância não aumenta em proporção com a frequência do termo.

Frequência do termo **tf**

- TA frequência $\text{tf}_{t,d}$ do termo t no documento d é definida como o número de vezes que t ocorre em d.
- Queremos usar tf ao comput escores de correspondência consulta-documento.
- Mas como?
- Frequência absoluta dos tesmos não é o que queremos pois:
- Um documento com $\text{tf} = 10$ ocorrências do termo é mais relevante do que um documento com $\text{tf} = 1$ ocorrência do termo.
- Mas não 10 vezes mais relevante.
- A Relevância não aumenta em proporção com a frequência do termo.

Ao invés da frequência absoluta: ponderação pelo *Log* da frequência

Ao invés da frequência absoluta: ponderação pelo *Log* da frequência

- O peso *log* da frequência do termo t em d é definido como

$$w_{t,d} = \begin{cases} 1 + \log_{10} \mathrm{tf}_{t,d} & \text{se } \mathrm{tf}_{t,d} > 0 \\ 0 & \text{caso contrário} \end{cases}$$

Ao invés da frequência absoluta: ponderação pelo *Log* da frequência

- O peso *log* da frequência do termo t em d é definido como

$$w_{t,d} = \begin{cases} 1 + \log_{10} \mathrm{tf}_{t,d} & \text{se } \mathrm{tf}_{t,d} > 0 \\ 0 & \text{caso contrário} \end{cases}$$

- $\mathrm{tf}_{t,d} \rightarrow w_{t,d}$:
  $0 \rightarrow 0$, $1 \rightarrow 1$, $2 \rightarrow 1.3$, $10 \rightarrow 2$, $1000 \rightarrow 4$, etc.

Ao invés da frequência absoluta: ponderação pelo *Log* da frequência

- O peso *log* da frequência do termo t em d é definido como

$$w_{t,d} = \begin{cases} 1 + \log_{10} \mathrm{tf}_{t,d} & \text{se } \mathrm{tf}_{t,d} > 0 \\ 0 & \text{caso contrário} \end{cases}$$

- $\mathrm{tf}_{t,d} \rightarrow w_{t,d}$:
  $0 \rightarrow 0$, $1 \rightarrow 1$, $2 \rightarrow 1.3$, $10 \rightarrow 2$, $1000 \rightarrow 4$, etc.
- Escore de um par consulta-documento: Soma sobre os termos t em q e d:
  escore $\mathrm{tf}(q,d) = \sum_{t \in q \cap d} (1 + \log \mathrm{tf}_{t,d})$

Ao invés da frequência absoluta: ponderação pelo *Log* da frequência

- O peso *log* da frequência do termo t em d é definido como

$$w_{t,d} = \begin{cases} 1 + \log_{10} \mathrm{tf}_{t,d} & \text{se } \mathrm{tf}_{t,d} > 0 \\ 0 & \text{caso contrário} \end{cases}$$

- $\mathrm{tf}_{t,d} \rightarrow w_{t,d}$:
  $0 \rightarrow 0$, $1 \rightarrow 1$, $2 \rightarrow 1.3$, $10 \rightarrow 2$, $1000 \rightarrow 4$, etc.
- Escore de um par consulta-documento: Soma sobre os termos t em q e d:
  escore $\mathrm{tf}(q,d) = \sum_{t \in q \cap d} (1 + \log \mathrm{tf}_{t,d})$
- O escore é 0 se nenhum dos termos de consulta está presente no documento.

Exercício

- Compute o escore de correspondência de Jaccard e o peso de correspondência tf para os seguintes pares consulta-documento.
- q: [informação sobre carros] d: "Tudo que você sempre quis saber sobre carros"
- q: [informação sobre carros] d: "Informação sobre caminhões, informação sobre aviões, informação sobre trens"
- q: [carros vermelhos e caminhões vermelhos] d: "Policiais param carros vermelhos com mais frequência"

Frequência no documento vs. frequência na coleção

Frequência no documento vs. frequência na coleção

- Além da frequência do termo (sua frequência no documento)

  ...

Frequência no documento vs. frequência na coleção

- Além da frequência do termo (sua frequência no documento) ...
- ... também queremos usar a frequência do termo na coleção
  Para rankear e ponderar.

Peso Desejado para Termos Raros

Peso Desejado para Termos Raros

- Termos raros são mais informativos do que termos frequentes.

Peso Desejado para Termos Raros

- Termos raros são mais informativos do que termos frequentes.
- Considere um termo em uma consulta que é raro na coleção (p.ex., ARACNOCÊNTRICO).

Peso Desejado para Termos Raros

- Termos raros são mais informativos do que termos frequentes.
- Considere um termo em uma consulta que é raro na coleção (p.ex., ARACNOCÊNTRICO).
- Um documento contendo este termo tem grandes chances de ser relevante.

Peso Desejado para Termos Raros

- Termos raros são mais informativos do que termos frequentes.
- Considere um termo em uma consulta que é raro na coleção (p.ex., ARACNOCÊNTRICO).
- Um documento contendo este termo tem grandes chances de ser relevante.
- $\rightarrow$ Queremos Pesos altos para termos raros como ARACNOCÊNTRICO.

Peso desejado para termos frequentes

Peso desejado para termos frequentes

- Termos frequentes são menos informativos do que termos raros.

Peso desejado para termos frequentes

- Termos frequentes são menos informativos do que termos raros.
- Considere um termo na consulta, que é frequente na coleção (p.ex., BOM, AUMENTA, LINHA).

Peso desejado para termos frequentes

- Termos frequentes são menos informativos do que termos raros.
- Considere um termo na consulta, que é frequente na coleção (p.ex., BOM, AUMENTA, LINHA).
- Um documento contendo estes termos tem mais chances de ser relevante do que um documento que não os contenha...

Peso desejado para termos frequentes

- Termos frequentes são menos informativos do que termos raros.
- Considere um termo na consulta, que é frequente na coleção (p.ex., BOM, AUMENTA, LINHA).
- Um documento contendo estes termos tem mais chances de ser relevante do que um documento que não os contenha...
- ...mas palavras como BOM, AUMENTA e LINHA Não são indicadores garantidos de relevância.

Peso desejado para termos frequentes

- Termos frequentes são menos informativos do que termos raros.
- Considere um termo na consulta, que é frequente na coleção (p.ex., BOM, AUMENTA, LINHA).
- Um documento contendo estes termos tem mais chances de ser relevante do que um documento que não os contenha...
- ...mas palavras como BOM, AUMENTA e LINHA Não são indicadores garantidos de relevância.
- $\rightarrow$ Para termos frequentes como BOM, AUMENTA, and LINHA, queremos pesos positivos ...

Peso desejado para termos frequentes

- Termos frequentes são menos informativos do que termos raros.
- Considere um termo na consulta, que é frequente na coleção (p.ex., BOM, AUMENTA, LINHA).
- Um documento contendo estes termos tem mais chances de ser relevante do que um documento que não os contenha...
- ...mas palavras como BOM, AUMENTA e LINHA Não são indicadores garantidos de relevância.
- → Para termos frequentes como BOM, AUMENTA, and LINHA, queremos pesos positivos ...
- ...mas pesos mais baixos que os de termos raros.

Frequência de Documento

- Queremos pesos altos para termos raros como ARACNOCÊNTRICO.
- Queremos baixos pesos positivos para termos frequentes como BOM, AUMENTA, and LINHA.

Frequência de Documento

- Queremos pesos altos para termos raros como ARACNOCÊNTRICO.
- Queremos baixos pesos positivos para termos frequentes como BOM, AUMENTA, and LINHA.
- Usaremos a frequência de documento para incluir isto no cálculo do escore de correspondência.

Frequência de Documento

- Queremos pesos altos para termos raros como ARACNOCÊNTRICO.
- Queremos baixos pesos positivos para termos frequentes como BOM, AUMENTA, and LINHA.
- Usaremos a frequência de documento para incluir isto no cálculo do escore de correspondência.
- TA frequência de documento é o número de documentos na coleção em que o termo ocorre.

idf

idf

- df_t é a frequência de documento, o número de documentos em que t ocorre.

idf

- df_t é a frequência de documento, o número de documentos em que t ocorre.
- df_t é uma medida inversa da informatividade do termo t.

idf

- df_t é a frequência de documento, o número de documentos em que t ocorre.
- df_t é uma medida inversa da informatividade do termo t.
- Definimos idf de um termo t da seguinte maneira:

$$\text{idf}_t = \log_{10} \frac{N}{\text{df}_t}$$

(N é o número de documentos na coleção.)

idf

- df_t é a frequência de documento, o número de documentos em que t ocorre.
- df_t é uma medida inversa da informatividade do termo t.
- Definimos idf de um termo t da seguinte maneira:

$$\text{idf}_t = \log_{10} \frac{N}{\text{df}_t}$$

(N é o número de documentos na coleção.)

- idf_t é uma medida da informatividade do termo.

idf

- df_t é a frequência de documento, o número de documentos em que t ocorre.
- df_t é uma medida inversa da informatividade do termo t.
- Definimos idf de um termo t da seguinte maneira:

$$\text{idf}_t = \log_{10} \frac{N}{\text{df}_t}$$

(N é o número de documentos na coleção.)

- idf_t é uma medida da informatividade do termo.
- $[\log N/\text{df}_t]$ ao invés de $[N/\text{df}_t]$ para "atenuar" o efeito de idf

idf

- df_t é a frequência de documento, o número de documentos em que t ocorre.
- df_t é uma medida inversa da informatividade do termo t.
- Definimos idf de um termo t da seguinte maneira:

$$\text{idf}_t = \log_{10} \frac{N}{\text{df}_t}$$

(N é o número de documentos na coleção.)

- idf_t é uma medida da informatividade do termo.
- $[\log N/\text{df}_t]$ ao invés de $[N/\text{df}_t]$ para "atenuar" o efeito de idf
- Note que usamos a transformação log para ambas as frequências: termo e documento.

Exemplos de idf

Exemplos de idf

Calcule idf_t usando a fórmula the formula: $\text{idf}_t = \log_{10} \frac{1{,}000{,}000}{\text{df}_t}$

term	df_t	idf_t
calpurnia	1	
animal	100	
sunday	1000	
fly	10,000	
under	100,000	
the	1,000,000	

Exemplos de idf

Calcule idf_t usando a fórmula the formula: $\text{idf}_t = \log_{10} \frac{1{,}000{,}000}{\text{df}_t}$

term	df_t	idf_t
calpurnia	1	
animal	100	
sunday	1000	
fly	10,000	
under	100,000	
the	1,000,000	

Exemplos de idf

Calcule idf_t usando a fórmula the formula: $\text{idf}_t = \log_{10} \frac{1{,}000{,}000}{\text{df}_t}$

term	df_t	idf_t
calpurnia	1	6
animal	100	4
sunday	1000	3
fly	10,000	2
under	100,000	1
the	1,000,000	0

Efeito do idf no rankeamento

Efeito do idf no rankeamento

- idf afeta o rankeamento de documentos para consultas com pelo menos dois termos.

Efeito do idf no rankeamento

- idf afeta o rankeamento de documentos para consultas com pelo menos dois termos.
- Por exemplo, na consulta "aracnocêntric linha", a ponderação por idf aumenta o peso relativo de ARACNOCÊNTRICO e diminui o peso relativo de LINHA.

Efeito do idf no rankeamento

- idf afeta o rankeamento de documentos para consultas com pelo menos dois termos.
- Por exemplo, na consulta "aracnocêntric linha", a ponderação por idf aumenta o peso relativo de ARACNOCÊNTRICO e diminui o peso relativo de LINHA.
- idf tem pouco efeito no rankeamento consultas de um único termo.

Frequência na Coleção vs. frequência de Documento

palavra	frequência na coleção	frequência de documentos
SEGURO	10440	3997
TENTAR	10422	8760

- Frequência na coleção de t: número de tokens t na coleção
- frequência de documentos de t: números de documentos em que t ocorre

Frequência na Coleção vs. frequência de Documento

palavra	frequência na coleção	frequência de documentos
SEGURO	10440	3997
TENTAR	10422	8760

- Frequência na coleção de t: número de tokens t na coleção
- frequência de documentos de t: números de documentos em que t ocorre
- Porque estes números?

Frequência na Coleção vs. frequência de Documento

palavra	frequência na coleção	frequência de documentos
SEGURO	10440	3997
TENTAR	10422	8760

- Frequência na coleção de t: número de tokens t na coleção
- frequência de documentos de t: números de documentos em que t ocorre
- Porque estes números?
- Qual palavra é um melhor termo de busca (e deveria receber o maior peso)?

Frequência na Coleção vs. frequência de Documento

palavra	frequência na coleção	frequência de documentos
SEGURO	10440	3997
TENTAR	10422	8760

- Frequência na coleção de t: número de tokens t na coleção
- frequência de documentos de t: números de documentos em que t ocorre
- Porque estes números?
- Qual palavra é um melhor termo de busca (e deveria receber o maior peso)?
- Este exemplo sugere que df (e idf) são melhores para ponderação do que cf (e "icf").

Ponderação por tf-idf

Ponderação por tf-idf

- A ponderação por tf-idf de um termo é o produto de seu tf e seu idf.

Ponderação por tf-idf

- A ponderação por tf-idf de um termo é o produto de seu tf e seu idf.
- $$w_{t,d} = (1 + \log \mathrm{tf}_{t,d}) \cdot \log \frac{N}{\mathrm{df}_t}$$

Ponderação por tf-idf

- A ponderação por tf-idf de um termo é o produto de seu tf e seu idf.
- $$w_{t,d} = (1 + \log \mathrm{tf}_{t,d}) \cdot \log \frac{N}{\mathrm{df}_t}$$
- tf

Ponderação por tf-idf

- A ponderação por tf-idf de um termo é o produto de seu tf e seu idf.
- $$w_{t,d} = (1 + \log \mathrm{tf}_{t,d}) \cdot \log \frac{N}{\mathrm{df}_t}$$
- idf

Ponderação por tf-idf

- A ponderação por tf-idf de um termo é o produto de seu tf e seu idf.
- $$w_{t,d} = (1 + \log \mathrm{tf}_{t,d}) \cdot \log \frac{N}{\mathrm{df}_t}$$
- Melhor esquema conhecido de ponderação em recuperação de informação

Ponderação por tf-idf

- A ponderação por tf-idf de um termo é o produto de seu tf e seu idf.
- $$w_{t,d} = (1 + \log \mathrm{tf}_{t,d}) \cdot \log \frac{N}{\mathrm{df}_t}$$

- Melhor esquema conhecido de ponderação em recuperação de informação
- Note: O "-" em tf-idf is a hyphen, não um sinal de subtração!

Ponderação por tf-idf

- A ponderação por tf-idf de um termo é o produto de seu tf e seu idf.
- $$w_{t,d} = (1 + \log \mathrm{tf}_{t,d}) \cdot \log \frac{N}{\mathrm{df}_t}$$

- Melhor esquema conhecido de ponderação em recuperação de informação
- Note: O "-" em tf-idf is a hyphen, não um sinal de subtração!
- Outros nomes: tf.idf, tf x idf

Resumo: tf-idf

Resumo: tf-idf

- Atribuir um peso tf-idf para cada termo t em cada documento d: $w_{t,d} = (1 + \log \mathrm{tf}_{t,d}) \cdot \log \frac{N}{\mathrm{df}_t}$

Resumo: tf-idf

- Atribuir um peso tf-idf para cada termo t em cada documento d: $w_{t,d} = (1 + \log \mathrm{tf}_{t,d}) \cdot \log \frac{N}{\mathrm{df}_t}$
- O peso tf-idf ...

Resumo: tf-idf

- Atribuir um peso tf-idf para cada termo t em cada documento d: $w_{t,d} = (1 + \log \mathrm{tf}_{t,d}) \cdot \log \frac{N}{\mathrm{df}_t}$
- O peso tf-idf ...
  - ...Aumenta com o número de ocorrências dentro de um documento. (frequência do termo)

Resumo: tf-idf

- Atribuir um peso tf-idf para cada termo t em cada documento d: $w_{t,d} = (1 + \log \mathrm{tf}_{t,d}) \cdot \log \frac{N}{\mathrm{df}_t}$
- O peso tf-idf ...
  - ...Aumenta com o número de ocorrências dentro de um documento. (frequência do termo)
  - ...Aumenta com a raridade do termo na coleção. (frequência inversa de documentos)

Exercício: Frequência de Termo, Coleção and Documento

Grandeza	Símbolo	Definição
frequência do termo	$\text{tf}_{t,d}$	número de ocorrências de t in d
frequência de documentos	df_t	número de documentos na coleção em que t ocorre
frequência na coleção	cf_t	número total de ocorrências de t na coleção

- Relação entre df e cf?
- Relação entre tf e cf?
- Relação entre tf e df?

Matriz de Incidência Binária

	Anthony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth	...
Anthony	1	1	0	0	0	1	
Brutus	1	1	0	1	0	0	
Caesar	1	1	0	1	1	1	
Calpurnia	0	1	0	0	0	0	
Cleopatra	1	0	0	0	0	0	
mercy	1	0	1	1	1	1	
worser	1	0	1	1	1	0	
...							

Cada documento é representado como um vetor binário $\in \{0,1\}^{|V|}$.

Matriz de Contagem

	Anthony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth	...
Anthony	157	73	0	0	0	1	
Brutus	4	157	0	2	0	0	
Caesar	232	227	0	2	1	0	
Calpurnia	0	10	0	0	0	0	
Cleopatra	57	0	0	0	0	0	
mercy	2	0	3	8	5	8	
worser	2	0	1	1	1	5	
...							

Cada Documento é agora representado como um vetor de contagem $\in \mathbb{N}^{|V|}$.

Matriz Binária → Contagem → Pesos

	Anthony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth	...
ANTHONY	5.25	3.18	0.0	0.0	0.0	0.35	
BRUTUS	1.21	6.10	0.0	1.0	0.0	0.0	
CAESAR	8.59	2.54	0.0	1.51	0.25	0.0	
CALPURNIA	0.0	1.54	0.0	0.0	0.0	0.0	
CLEOPATRA	2.85	0.0	0.0	0.0	0.0	0.0	
MERCY	1.51	0.0	1.90	0.12	5.25	0.88	
WORSER	1.37	0.0	0.11	4.15	0.25	1.95	
...							

Cada documento agora é representado como um vetor real de pesos tf-idf $\in \mathbb{R}^{|V|}$.

Matriz Binária → Contagem → Pesos

	Anthony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth	...
ANTHONY	5.25	3.18	0.0	0.0	0.0	0.35	
BRUTUS	1.21	6.10	0.0	1.0	0.0	0.0	
CAESAR	8.59	2.54	0.0	1.51	0.25	0.0	
CALPURNIA	0.0	1.54	0.0	0.0	0.0	0.0	
CLEOPATRA	2.85	0.0	0.0	0.0	0.0	0.0	
MERCY	1.51	0.0	1.90	0.12	5.25	0.88	
WORSER	1.37	0.0	0.11	4.15	0.25	1.95	
...							

Cada documento agora é representado como um vetor real de pesos tf-idf $\in \mathbb{R}^{|V|}$.

Documentos como vetores

Documentos como vetores

- Cada documento é agora representado como um vetor real de pesos tf-idf $\in \mathbb{R}^{|V|}$.

Documentos como vetores

- Cada documento é agora representado como um vetor real de pesos tf-idf $\in \mathbb{R}^{|V|}$.
- Então agora temos um espaço vetorial $|V|$-dimensional.

Documentos como vetores

- Cada documento é agora representado como um vetor real de pesos tf-idf $\in \mathbb{R}^{|V|}$.
- Então agora temos um espaço vetorial $|V|$-dimensional.
- Os termos são os eixos do espaço.

Documentos como vetores

- Cada documento é agora representado como um vetor real de pesos tf-idf $\in \mathbb{R}^{|V|}$.
- Então agora temos um espaço vetorial $|V|$-dimensional.
- Os termos são os eixos do espaço.
- os documentos são pontos ou vetores neste espaço.

Documentos como vetores

- Cada documento é agora representado como um vetor real de pesos tf-idf $\in \mathbb{R}^{|V|}$.
- Então agora temos um espaço vetorial $|V|$-dimensional.
- Os termos são os eixos do espaço.
- os documentos são pontos ou vetores neste espaço.
- Dimensionalidade muito alta: dezenas de milhões de dimensões quando aplica-se isto a máquinas de busca para a web

Documentos como vetores

- Cada documento é agora representado como um vetor real de pesos tf-idf $\in \mathbb{R}^{|V|}$.
- Então agora temos um espaço vetorial $|V|$-dimensional.
- Os termos são os eixos do espaço.
- os documentos são pontos ou vetores neste espaço.
- Dimensionalidade muito alta: dezenas de milhões de dimensões quando aplica-se isto a máquinas de busca para a web
- Cada vetor é extremamente esparso - a maioria dos valores são zero.

Consultas como vetores

Consultas como vetores

- Idéia chave 1: faça os mesmo para as consultas: represente-as como vetores neste espaço multi-dimensional

Consultas como vetores

- Idéia chave 1: faça os mesmo para as consultas: represente-as como vetores neste espaço multi-dimensional
- Idéia chave 2: Rankeie os documentos de acordo com sua proximidade à consulta.

Consultas como vetores

- Idéia chave 1: faça os mesmo para as consultas: represente-as como vetores neste espaço multi-dimensional
- Idéia chave 2: Rankeie os documentos de acordo com sua proximidade à consulta.
- proximidade= similaridade

Consultas como vetores

- Idéia chave 1: faça os mesmo para as consultas: represente-as como vetores neste espaço multi-dimensional
- Idéia chave 2: Rankeie os documentos de acordo com sua proximidade à consulta.
- proximidade= similaridade
- proximidade $\approx$ distância negativa

Consultas como vetores

- Idéia chave 1: faça os mesmo para as consultas: represente-as como vetores neste espaço multi-dimensional
- Idéia chave 2: Rankeie os documentos de acordo com sua proximidade à consulta.
- proximidade= similaridade
- proximidade $\approx$ distância negativa
- Lembre-se: fazemos isso para nos afastar do tudo-ou-nada do modelo Booleano.

Como formalizamos similaridade em um espaço vetorial?

Como formalizamos similaridade em um espaço vetorial?

- Primeira tentativa: distância (negativa) entre dois pontos

Como formalizamos similaridade em um espaço vetorial?

- Primeira tentativa: distância (negativa) entre dois pontos
- (= distância entre as extremidades de dois vetores)

Como formalizamos similaridade em um espaço vetorial?

- Primeira tentativa: distância (negativa) entre dois pontos
- (= distância entre as extremidades de dois vetores)
- Distância Euclidiana?

Como formalizamos similaridade em um espaço vetorial?

- Primeira tentativa: distância (negativa) entre dois pontos
- (= distância entre as extremidades de dois vetores)
- Distância Euclidiana?
- Distância Euclidiana é uma má ideia . . .

Como formalizamos similaridade em um espaço vetorial?

- Primeira tentativa: distância (negativa) entre dois pontos
- (= distância entre as extremidades de dois vetores)
- Distância Euclidiana?
- Distância Euclidiana é uma má ideia . . .
- . . . pois a distância Euclidiana é grande for vetores de comprimentos diferentes.

Porquê distância é uma má idéia

Porquê distância é uma má idéia

POOR

d_1: *Ranks of starving poets swell*

d_2: Rich poor gap grows

q: *[rich poor]*

d_3: *Record baseball salaries in 2010*

RICH

A distância Euclidiana de $\vec{q}$ e $\vec{d}_2$ é grande ainda que a distribuição de termos na consulta q e a distribuição de termos no documento d_2 sejam muito similares.

Usar ângulos ao invés de distâncias

Usar ângulos ao invés de distâncias

- Rankear documentos de acordo com o ângulo formado com a consulta

Usar ângulos ao invés de distâncias

- Rankear documentos de acordo com o ângulo formado com a consulta
- Imagine: pegue um documento d e adicione-o ao final dele mesmo. chame este novo documento d'. d' é duas vezes mais longo que d.

Usar ângulos ao invés de distâncias

- Rankear documentos de acordo com o ângulo formado com a consulta
- Imagine: pegue um documento d e adicione-o ao final dele mesmo. chame este novo documento d'. d' é duas vezes mais longo que d.
- "Semanticamente" d e d' têm o mesmo conteúdo.

Usar ângulos ao invés de distâncias

- Rankear documentos de acordo com o ângulo formado com a consulta
- Imagine: pegue um documento d e adicione-o ao final dele mesmo. chame este novo documento d'. d' é duas vezes mais longo que d.
- "Semanticamente" d e d' têm o mesmo conteúdo.
- O ângulo entre os dois documentos é 0, correspondendo a máxima similaridade...

Usar ângulos ao invés de distâncias

- Rankear documentos de acordo com o ângulo formado com a consulta
- Imagine: pegue um documento d e adicione-o ao final dele mesmo. chame este novo documento d'. d' é duas vezes mais longo que d.
- "Semanticamente" d e d' têm o mesmo conteúdo.
- O ângulo entre os dois documentos é 0, correspondendo a máxima similaridade. . .
- . . . Mas sua distância Euclidiana é bem grande.

de Ângulos a Cossenos

de Ângulos a Cossenos

- O dois conceitos a seguir são equivalentes.

de Ângulos a Cossenos

- O dois conceitos a seguir são equivalentes.
  - Rankear documentos de acordo com o ângulo entre consulta e documento em ordem decrescente

de Ângulos a Cossenos

- O dois conceitos a seguir são equivalentes.
  - Rankear documentos de acordo com o ângulo entre consulta e documento em ordem decrescente
  - Rankear documentos de acordo com o cosseno(query,document) em ordem crescente

de Ângulos a Cossenos

- O dois conceitos a seguir são equivalentes.
  - Rankear documentos de acordo com o ângulo entre consulta e documento em ordem decrescente
  - Rankear documentos de acordo com o cosseno(query,document) em ordem crescente
- O cosseno é uma função monotonicamente decrescente do ângulo no intervalo $[0°, 180°]$

Normalização do comprimento

Normalização do comprimento

- Como calculamos o cosseno?

Normalização do comprimento

- Como calculamos o cosseno?
- Um vetor poder ser normalizado com respeito ao seu comprimento, por meio da divisão de cada um de seus componentes por seu comprimento – aqui usamos a norma L_2 norm: $||x||_2 = \sqrt{\sum_i x_i^2}$

Normalização do comprimento

- Como calculamos o cosseno?
- Um vetor poder ser normalizado com respeito ao seu comprimento, por meio da divisão de cada um de seus componentes por seu comprimento – aqui usamos a norma L_2 norm: $||x||_2 = \sqrt{\sum_i x_i^2}$
- Isto mapeia os vetores na esfera unitária ...

Normalização do comprimento

- Como calculamos o cosseno?
- Um vetor poder ser normalizado com respeito ao seu comprimento, por meio da divisão de cada um de seus componentes por seu comprimento – aqui usamos a norma L_2 norm: $||x||_2 = \sqrt{\sum_i x_i^2}$
- Isto mapeia os vetores na esfera unitária ...
- ... uma vez que após a normalização: $||x||_2 = \sqrt{\sum_i x_i^2} = 1.0$

Normalização do comprimento

- Como calculamos o cosseno?
- Um vetor poder ser normalizado com respeito ao seu comprimento, por meio da divisão de cada um de seus componentes por seu comprimento – aqui usamos a norma L_2 norm: $||x||_2 = \sqrt{\sum_i x_i^2}$
- Isto mapeia os vetores na esfera unitária ...
- ... uma vez que após a normalização: $||x||_2 = \sqrt{\sum_i x_i^2} = 1.0$
- Por conseguinte, documentos longos e curto têm pesos com a mesma ordem de magnitude.

Normalização do comprimento

- Como calculamos o cosseno?
- Um vetor poder ser normalizado com respeito ao seu comprimento, por meio da divisão de cada um de seus componentes por seu comprimento – aqui usamos a norma L_2 norm: $||x||_2 = \sqrt{\sum_i x_i^2}$
- Isto mapeia os vetores na esfera unitária ...
- ... uma vez que após a normalização: $||x||_2 = \sqrt{\sum_i x_i^2} = 1.0$
- Por conseguinte, documentos longos e curto têm pesos com a mesma ordem de magnitude.
- Efeito nos documentos d and d' (d somado a si mesmo) do slide anterior: eles têm vetores idênticos após normalização do comprimento.

Similaridade por Cosseno entre consulta e documento

Similaridade por Cosseno entre consulta e documento

$$\cos(\vec{q},\vec{d}) = \text{SIM}(\vec{q},\vec{d}) = \frac{\vec{q}\cdot\vec{d}}{|\vec{q}||\vec{d}|} = \frac{\sum_{i=1}^{|V|} q_i d_i}{\sqrt{\sum_{i=1}^{|V|} q_i^2}\sqrt{\sum_{i=1}^{|V|} d_i^2}}$$

- q_i é o peso tf-idf do termo i na consulta.

Similaridade por Cosseno entre consulta e documento

$$\cos(\vec{q},\vec{d}) = \text{SIM}(\vec{q},\vec{d}) = \frac{\vec{q}\cdot\vec{d}}{|\vec{q}||\vec{d}|} = \frac{\sum_{i=1}^{|V|} q_i d_i}{\sqrt{\sum_{i=1}^{|V|} q_i^2}\sqrt{\sum_{i=1}^{|V|} d_i^2}}$$

- q_i é o peso tf-idf do termo i na consulta.
- d_i é o peso tf-idf do termo i no documento.

Similaridade por Cosseno entre consulta e documento

$$\cos(\vec{q},\vec{d}) = \text{SIM}(\vec{q},\vec{d}) = \frac{\vec{q}\cdot\vec{d}}{|\vec{q}||\vec{d}|} = \frac{\sum_{i=1}^{|V|} q_i d_i}{\sqrt{\sum_{i=1}^{|V|} q_i^2}\sqrt{\sum_{i=1}^{|V|} d_i^2}}$$

- q_i é o peso tf-idf do termo i na consulta.
- d_i é o peso tf-idf do termo i no documento.
- $|\vec{q}|$ e $|\vec{d}|$ são os comprimentos de $\vec{q}$ and $\vec{d}$.

Similaridade por Cosseno entre consulta e documento

$$\cos(\vec{q},\vec{d}) = \text{SIM}(\vec{q},\vec{d}) = \frac{\vec{q}\cdot\vec{d}}{|\vec{q}||\vec{d}|} = \frac{\sum_{i=1}^{|V|} q_i d_i}{\sqrt{\sum_{i=1}^{|V|} q_i^2}\sqrt{\sum_{i=1}^{|V|} d_i^2}}$$

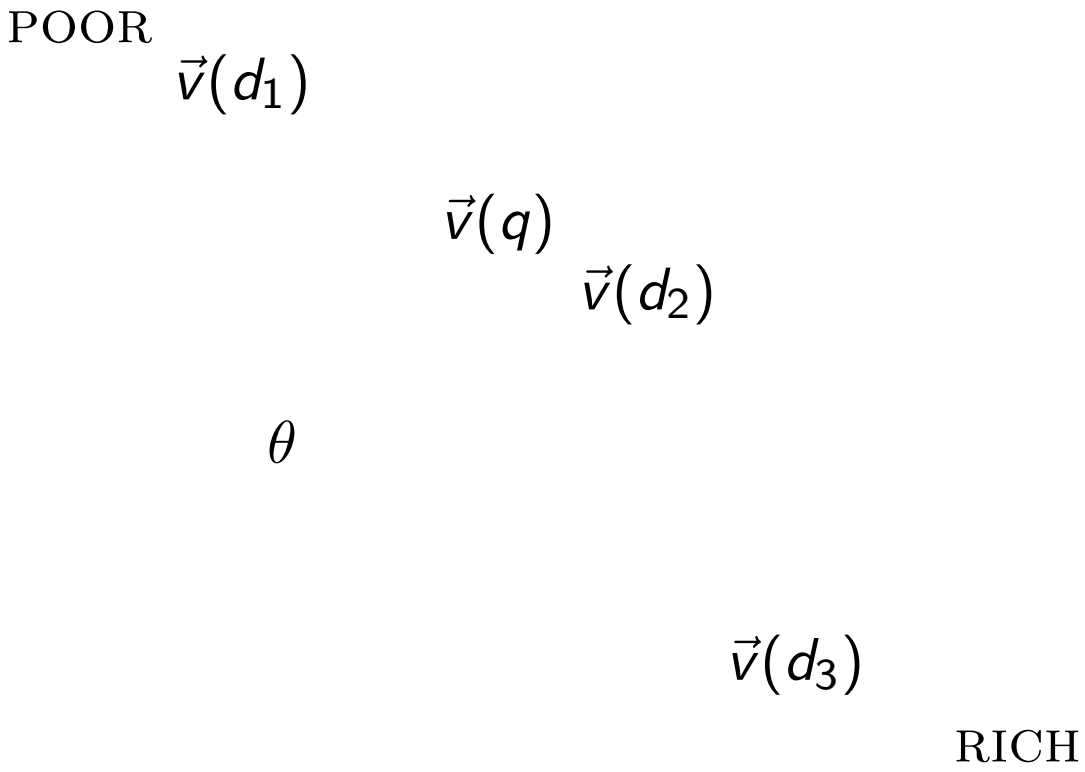
- q_i é o peso tf-idf do termo i na consulta.
- d_i é o peso tf-idf do termo i no documento.
- $|\vec{q}|$ e $|\vec{d}|$ são os comprimentos de $\vec{q}$ and $\vec{d}$.
- Isto é a similaridade por cosseno entre $\vec{q}$ e $\vec{d}$ . . . . . ou, o cosseno do ângulo entre $\vec{q}$ e $\vec{d}$.

Cosseno para vetores normalizados

- Para vetores normalizados, o cosseno é equivalente ao produto interno ou produto escalar .
- $\cos(\vec{q}, \vec{d}) = \vec{q} \cdot \vec{d} = \sum_i q_i \cdot d_i$
  - (se $\vec{q}$ e $\vec{d}$ normalizados no comprimento).

Similaridade por Cosseno Ilustrada

Similaridade por Cosseno Ilustrada



Cosseno: Exemplo

Quão similares são estas novelas?

SaS: Sense and Sensibility

PaP: Pride and Prejudice

WH: Wuthering Heights

Cosseno: Exemplo

Quão similares são estas novelas?

SaS: Sense and Sensibility

PaP: Pride and Prejudice

WH: Wuthering Heights

frequências de termos (contagens)

termo	SaS	PaP	WH
AFFECTION	115	58	20
JEALOUS	10	7	11
GOSSIP	2	0	6
WUTHERING	0	0	38

Cosseno: Exemplo

frequências de termos (contagens)

term	SaS	PaP	WH
AFFECTION	115	58	20
JEALOUS	10	7	11
GOSSIP	2	0	6
WUTHERING	0	0	38

Cosseno: Exemplo

frequências de termos (contagens)

term	SaS	PaP	WH
AFFECTION	115	58	20
JEALOUS	10	7	11
GOSSIP	2	0	6
WUTHERING	0	0	38

log da frequência

term	SaS	PaP	WH
AFFECTION	3.06	2.76	2.30
JEALOUS	2.0	1.85	2.04
GOSSIP	1.30	0	1.78
WUTHERING	0	0	2.58

Cosseno: Exemplo

frequências de termos (contagens)

term	SaS	PaP	WH
AFFECTION	115	58	20
JEALOUS	10	7	11
GOSSIP	2	0	6
WUTHERING	0	0	38

log da frequência

term	SaS	PaP	WH
AFFECTION	3.06	2.76	2.30
JEALOUS	2.0	1.85	2.04
GOSSIP	1.30	0	1.78
WUTHERING	0	0	2.58

(Para simplificar este exemplo, não faremos ponderação por idf .)

Cosseno: Exemplo

log da frequência

term	SaS	PaP	WH
AFFECTION	3.06	2.76	2.30
JEALOUS	2.0	1.85	2.04
GOSSIP	1.30	0	1.78
WUTHERING	0	0	2.58

Cosseno: Exemplo

log da frequência

term	SaS	PaP	WH
AFFECTION	3.06	2.76	2.30
JEALOUS	2.0	1.85	2.04
GOSSIP	1.30	0	1.78
WUTHERING	0	0	2.58

log da frequência
& normalização por cosseno

term	SaS	PaP	WH
AFFECTION	0.789	0.832	0.524
JEALOUS	0.515	0.555	0.465
GOSSIP	0.335	0.0	0.405
WUTHERING	0.0	0.0	0.588

Cosseno: Exemplo

log da frequência

term	SaS	PaP	WH
AFFECTION	3.06	2.76	2.30
JEALOUS	2.0	1.85	2.04
GOSSIP	1.30	0	1.78
WUTHERING	0	0	2.58

log da frequência & normalização por cosseno

term	SaS	PaP	WH
AFFECTION	0.789	0.832	0.524
JEALOUS	0.515	0.555	0.465
GOSSIP	0.335	0.0	0.405
WUTHERING	0.0	0.0	0.588

- cos(SaS,PaP) $\approx$
  $0.789 * 0.832 + 0.515 * 0.555 + 0.335 * 0.0 + 0.0 * 0.0 \approx 0.94.$

Cosseno: Exemplo

log da frequência

term	SaS	PaP	WH
AFFECTION	3.06	2.76	2.30
JEALOUS	2.0	1.85	2.04
GOSSIP	1.30	0	1.78
WUTHERING	0	0	2.58

log da frequência & normalização por cosseno

term	SaS	PaP	WH
AFFECTION	0.789	0.832	0.524
JEALOUS	0.515	0.555	0.465
GOSSIP	0.335	0.0	0.405
WUTHERING	0.0	0.0	0.588

- cos(SaS,PaP) $\approx$
  $0.789 * 0.832 + 0.515 * 0.555 + 0.335 * 0.0 + 0.0 * 0.0 \approx 0.94.$
- cos(SaS,WH) ≈ 0.79

Cosseno: Exemplo

log da frequência

term	SaS	PaP	WH
AFFECTION	3.06	2.76	2.30
JEALOUS	2.0	1.85	2.04
GOSSIP	1.30	0	1.78
WUTHERING	0	0	2.58

log da frequência & normalização por cosseno

term	SaS	PaP	WH
AFFECTION	0.789	0.832	0.524
JEALOUS	0.515	0.555	0.465
GOSSIP	0.335	0.0	0.405
WUTHERING	0.0	0.0	0.588

- cos(SaS,PaP) $\approx$
  $0.789 * 0.832 + 0.515 * 0.555 + 0.335 * 0.0 + 0.0 * 0.0 \approx 0.94.$
- cos(SaS,WH) ≈ 0.79
- cos(PaP,WH) ≈ 0.69

Cosseno: Exemplo

log da frequência

term	SaS	PaP	WH
AFFECTION	3.06	2.76	2.30
JEALOUS	2.0	1.85	2.04
GOSSIP	1.30	0	1.78
WUTHERING	0	0	2.58

log da frequência & normalização por cosseno

term	SaS	PaP	WH
AFFECTION	0.789	0.832	0.524
JEALOUS	0.515	0.555	0.465
GOSSIP	0.335	0.0	0.405
WUTHERING	0.0	0.0	0.588

- cos(SaS,PaP) $\approx$
  $0.789 * 0.832 + 0.515 * 0.555 + 0.335 * 0.0 + 0.0 * 0.0 \approx 0.94.$
- cos(SaS,WH) ≈ 0.79
- cos(PaP,WH) ≈ 0.69
- Porquê temos cos(SaS,PaP) $>$ cos(SAS,WH)?

Calculando o escore cosseno

Calculando o escore cosseno

EscoreCosseno(q)

1 *float* $Escores[N] = 0$
2 *float* $Comprimento[N]$
3 **for each** termo de busca t
4 **do** calcule $w_{t,q}$ e recupere a lista de postings para t
5 **for each** par$(d, tf_{t,d})$ na lista de
6 **do** $Escores[d] += w_{t,d} \times w_{t,q}$
7 Leia o *Comprimento*da matriz
8 **for each** d
9 **do** $Escores[d] = Escores[d]/Comprimento[d]$
10 **return** Top K componentes de $Escores[]$

Componentes da ponderação tf-idf

Frequência do termo		Frequência de documentos		Normalização	
n (natural)	$\mathrm{tf}_{t,d}$	n (no)	1	n (none)	1
l (logarithm)	$1 + \log(\mathrm{tf}_{t,d})$	t (idf)	$\log \frac{N}{\mathrm{df}_t}$	c (cosine)	$\frac{1}{\sqrt{w_1^2+w_2^2+\ldots+w_M^2}}$
a (augmented)	$0.5 + \frac{0.5 \times \mathrm{tf}_{t,d}}{\max_t(\mathrm{tf}_{t,d})}$	p (prob idf)	$\max\{0, \log \frac{N-t}{t}\}$	u (pivoted unique)	$1/u$
b (boolean)	$\begin{cases} 1 & \text{if } {}_{t,d} > 0 \\ 0 & \text{otherwise} \end{cases}$			b (byte size)	$1/CharLength^{\alpha}$, $\alpha < 1$
L (log ave)	$\frac{1+\log({}_{t,d})}{1+\log(_{t \in d}({}_{t,d}))}$				

Componentes da ponderação tf-idf

Frequência do termo		Frequência de documentos		Normalização	
n (natural)	$\mathrm{tf}_{t,d}$	n (no)	1	n (none)	1
l (logarithm)	$1 + \log(\mathrm{tf}_{t,d})$	t (idf)	$\log \frac{N}{\mathrm{df}_t}$	c (cosine)	$\frac{1}{\sqrt{w_1^2+w_2^2+...+w_M^2}}$
a (augmented)	$0.5 + \frac{0.5 \times \mathrm{tf}_{t,d}}{\max_t(\mathrm{tf}_{t,d})}$	p (prob idf)	$\max\{0, \log \frac{N-t}{t}\}$	u (pivoted unique)	$1/u$
b (boolean)	$\begin{cases} 1 & \text{if } {}_{t,d} > 0 \\ 0 & \text{otherwise} \end{cases}$			b (byte size)	$1/CharLength^{\alpha}$, $\alpha < 1$
L (log ave)	$\frac{1+\log({}_{t,d})}{1+\log(_{t \in d}({}_{t,d}))}$				

Melhor combinação conhecida de opções de ponderação

Componentes da ponderação tf-idf

Frequência do termo		Frequência de documentos		Normalização	
n (natural)	$\mathrm{tf}_{t,d}$	n (no)	1	n (none)	1
l (logarithm)	$1 + \log(\mathrm{tf}_{t,d})$	t (idf)	$\log \frac{N}{\mathrm{df}_t}$	c (cosine)	$\frac{1}{\sqrt{w_1^2+w_2^2+\ldots+w_M^2}}$
a (augmented)	$0.5 + \frac{0.5 \times \mathrm{tf}_{t,d}}{\max_t(\mathrm{tf}_{t,d})}$	p (prob idf)	$\max\{0, \log \frac{N-t}{t}\}$	u (pivoted unique)	$1/u$
b (boolean)	$\begin{cases} 1 & \text{if } {}_{t,d} > 0 \\ 0 & \text{otherwise} \end{cases}$			b (byte size)	$1/CharLength^{\alpha}$, $\alpha < 1$
L (log ave)	$\frac{1+\log({}_{t,d})}{1+\log({}_{t \in d}({}_{t,d}))}$				

Default: sem ponderação

Exemplo: tf-idf

Exemplo: tf-idf

- Frequentemente usamos ponderações diferentes para consultas e documentos.

Exemplo: tf-idf

- Frequentemente usamos ponderações diferentes para consultas e documentos.
- Notação: ddd.qqq

Exemplo: tf-idf

- Frequentemente usamos ponderações diferentes para consultas e documentos.
- Notação: ddd.qqq
- Exemplo: lnc.ltn

Exemplo: tf-idf

- Frequentemente usamos ponderações diferentes para consultas e documentos.
- Notação: ddd.qqq
- Exemplo: lnc.ltn
- documento: tf logarítmica, sem ponderação por df, normalização por cosseno

Exemplo: tf-idf

- Frequentemente usamos ponderações diferentes para consultas e documentos.
- Notação: ddd.qqq
- Exemplo: lnc.ltn
- documento: tf logarítmica, sem ponderação por df, normalização por cosseno
- consulta: tf logarítmica, idf, sem normalização

Exemplo: tf-idf

- Frequentemente usamos ponderações diferentes para consultas e documentos.
- Notação: ddd.qqq
- Exemplo: lnc.ltn
- documento: tf logarítmica, sem ponderação por df, normalização por cosseno
- consulta: tf logarítmica, idf, sem normalização
- É ruim não ponderar por idf o documento?

Exemplo: tf-idf

- Frequentemente usamos ponderações diferentes para consultas e documentos.
- Notação: ddd.qqq
- Exemplo: lnc.ltn
- documento: tf logarítmica, sem ponderação por df, normalização por cosseno
- consulta: tf logarítmica, idf, sem normalização
- É ruim não ponderar por idf o documento?
- Consulta: "best car insurance"

Exemplo: tf-idf

- Frequentemente usamos ponderações diferentes para consultas e documentos.
- Notação: ddd.qqq
- Exemplo: lnc.ltn
- documento: tf logarítmica, sem ponderação por df, normalização por cosseno
- consulta: tf logarítmica, idf, sem normalização
- É ruim não ponderar por idf o documento?
- Consulta: "best car insurance"
- Documento: "car insurance auto insurance"

Exemplo de tf-idf: lnc.ltn

Query: "best car insurance". Document: "car insurance auto insurance".

palavra	consulta					documento				produto
	tf-raw	tf-wght	df	idf	weight	tf-raw	tf-wght	weight	n'lized	
auto										
best										
car										
insurance										

Chave para as colunas: tf-raw: frequência absoluta do termo, sem ponderação, tf-wght: tf ponderada logaritmicamente, df: frequência de documentos, idf: frequência inversa de documentos, weight: O peso final do termo na consulta ou documento, n'lized: Peso de documentos após normalização por cosseno, product: O produto da ponderação final de consulta e documento

Exemplo de tf-idf: lnc.ltn

Query: "best car insurance". Document: "car insurance auto insurance".

palavra	consulta					documento				produto
	tf-raw	tf-wght	df	idf	weight	tf-raw	tf-wght	weight	n'lized	
auto	0									
best	1									
car	1									
insurance	1									

Chave para as colunas: tf-raw: frequência absoluta do termo, sem ponderação, tf-wght: tf ponderada logaritmicamente, df: frequência de documentos, idf: frequência inversa de documentos, weight: O peso final do termo na consulta ou documento, n'lized: Peso de documentos após normalização por cosseno, product: O produto da ponderação final de consulta e documento

Exemplo de tf-idf: lnc.ltn

Query: "best car insurance". Document: "car insurance auto insurance".

palavra	consulta					documento				produto
	tf-raw	tf-wght	df	idf	weight	tf-raw	tf-wght	weight	n'lized	
auto	0					1				
best	1					0				
car	1					1				
insurance	1					2				

Chave para as colunas: tf-raw: frequência absoluta do termo, sem ponderação, tf-wght: tf ponderada logaritmicamente, df: frequência de documentos, idf: frequência inversa de documentos, weight: O peso final do termo na consulta ou documento, n'lized: Peso de documentos após normalização por cosseno, product: O produto da ponderação final de consulta e documento

Exemplo de tf-idf: lnc.ltn

Query: "best car insurance". Document: "car insurance auto insurance".

palavra	consulta					documento				produto
	tf-raw	tf-wght	df	idf	weight	tf-raw	tf-wght	weight	n'lized	
auto	0	0				1				
best	1	1				0				
car	1	1				1				
insurance	1	1				2				

Chave para as colunas: tf-raw: frequência absoluta do termo, sem ponderação, tf-wght: tf ponderada logaritmicamente, df: frequência de documentos, idf: frequência inversa de documentos, weight: O peso final do termo na consulta ou documento, n'lized: Peso de documentos após normalização por cosseno, product: O produto da ponderação final de consulta e documento

Exemplo de tf-idf: lnc.ltn

Query: "best car insurance". Document: "car insurance auto insurance".

palavra	consulta					documento				produto
	tf-raw	tf-wght	df	idf	weight	tf-raw	tf-wght	weight	n'lized	
auto	0	0				1	1			
best	1	1				0	0			
car	1	1				1	1			
insurance	1	1				2	1.3			

Chave para as colunas: tf-raw: frequência absoluta do termo, sem ponderação, tf-wght: tf ponderada logaritmicamente, df: frequência de documentos, idf: frequência inversa de documentos, weight: O peso final do termo na consulta ou documento, n'lized: Peso de documentos após normalização por cosseno, product: O produto da ponderação final de consulta e documento

Exemplo de tf-idf: lnc.ltn

Query: "best car insurance". Document: "car insurance auto insurance".

palavra	consulta					documento				produto
	tf-raw	tf-wght	df	idf	weight	tf-raw	tf-wght	weight	n'lized	
auto	0	0	5000			1	1			
best	1	1	50000			0	0			
car	1	1	10000			1	1			
insurance	1	1	1000			2	1.3			

Chave para as colunas: tf-raw: frequência absoluta do termo, sem ponderação, tf-wght: tf ponderada logaritmicamente, df: frequência de documentos, idf: frequência inversa de documentos, weight: O peso final do termo na consulta ou documento, n'lized: Peso de documentos após normalização por cosseno, product: O produto da ponderação final de consulta e documento

Exemplo de tf-idf: lnc.ltn

Query: "best car insurance". Document: "car insurance auto insurance".

palavra	consulta					documento				produto
	tf-raw	tf-wght	df	idf	weight	tf-raw	tf-wght	weight	n'lized	
auto	0	0	5000	2.3		1	1			
best	1	1	50000	1.3		0	0			
car	1	1	10000	2.0		1	1			
insurance	1	1	1000	3.0		2	1.3			

Chave para as colunas: tf-raw: frequência absoluta do termo, sem ponderação, tf-wght: tf ponderada logaritmicamente, df: frequência de documentos, idf: frequência inversa de documentos, weight: O peso final do termo na consulta ou documento, n'lized: Peso de documentos após normalização por cosseno, product: O produto da ponderação final de consulta e documento

Exemplo de tf-idf: lnc.ltn

Query: "best car insurance". Document: "car insurance auto insurance".

palavra	consulta					documento				produto
	tf-raw	tf-wght	df	idf	weight	tf-raw	tf-wght	weight	n'lized	
auto	0	0	5000	2.3	0	1	1			
best	1	1	50000	1.3	1.3	0	0			
car	1	1	10000	2.0	2.0	1	1			
insurance	1	1	1000	3.0	3.0	2	1.3			

Chave para as colunas: tf-raw: frequência absoluta do termo, sem ponderação, tf-wght: tf ponderada logaritmicamente, df: frequência de documentos, idf: frequência inversa de documentos, weight: O peso final do termo na consulta ou documento, n'lized: Peso de documentos após normalização por cosseno, product: O produto da ponderação final de consulta e documento

Exemplo de tf-idf: lnc.ltn

Query: "best car insurance". Document: "car insurance auto insurance".

palavra	consulta					documento				produto
	tf-raw	tf-wght	df	idf	weight	tf-raw	tf-wght	weight	n'lized	
auto	0	0	5000	2.3	0	1	1			
best	1	1	50000	1.3	1.3	0	0			
car	1	1	10000	2.0	2.0	1	1			
insurance	1	1	1000	3.0	3.0	2	1.3			

Chave para as colunas: tf-raw: frequência absoluta do termo, sem ponderação, tf-wght: tf ponderada logaritmicamente, df: frequência de documentos, idf: frequência inversa de documentos, weight: O peso final do termo na consulta ou documento, n'lized: Peso de documentos após normalização por cosseno, product: O produto da ponderação final de consulta e documento

Exemplo de tf-idf: lnc.ltn

Query: "best car insurance". Document: "car insurance auto insurance".

palavra	consulta					documento				produto
	tf-raw	tf-wght	df	idf	weight	tf-raw	tf-wght	weight	n'lized	
auto	0	0	5000	2.3	0	1	1	1		
best	1	1	50000	1.3	1.3	0	0	0		
car	1	1	10000	2.0	2.0	1	1	1		
insurance	1	1	1000	3.0	3.0	2	1.3	1.3		

Chave para as colunas: tf-raw: frequência absoluta do termo, sem ponderação, tf-wght: tf ponderada logaritmicamente, df: frequência de documentos, idf: frequência inversa de documentos, weight: O peso final do termo na consulta ou documento, n'lized: Peso de documentos após normalização por cosseno, product: O produto da ponderação final de consulta e documento

Exemplo de tf-idf: lnc.ltn

Query: "best car insurance". Document: "car insurance auto insurance".

palavra	consulta					documento				produto
	tf-raw	tf-wght	df	idf	weight	tf-raw	tf-wght	weight	n'lized	
auto	0	0	5000	2.3	0	1	1	1	0.52	
best	1	1	50000	1.3	1.3	0	0	0	0	
car	1	1	10000	2.0	2.0	1	1	1	0.52	
insurance	1	1	1000	3.0	3.0	2	1.3	1.3	0.68	

Chave para as colunas: tf-raw: frequência absoluta do termo, sem ponderação, tf-wght: tf ponderada logaritmicamente, df: frequência de documentos, idf: frequência inversa de documentos, weight: O peso final do termo na consulta ou documento, n'lized: Peso de documentos após normalização por cosseno, product: O produto da ponderação final de consulta e documento

$\sqrt{1^2 + 0^2 + 1^2 + 1.3^2} \approx 1.92$
$1/1.92 \approx 0.52$
$1.3/1.92 \approx 0.68$

Exemplo de tf-idf: lnc.ltn

Query: "best car insurance". Document: "car insurance auto insurance".

palavra	consulta					documento				produto
	tf-raw	tf-wght	df	idf	weight	tf-raw	tf-wght	weight	n'lized	
auto	0	0	5000	2.3	0	1	1	1	0.52	0
best	1	1	50000	1.3	1.3	0	0	0	0	0
car	1	1	10000	2.0	2.0	1	1	1	0.52	1.04
insurance	1	1	1000	3.0	3.0	2	1.3	1.3	0.68	2.04

Chave para as colunas: tf-raw: frequência absoluta do termo, sem ponderação, tf-wght: tf ponderada logaritmicamente, df: frequência de documentos, idf: frequência inversa de documentos, weight: O peso final do termo na consulta ou documento, n'lized: Peso de documentos após normalização por cosseno, product: O produto da ponderação final de consulta e documento

Exemplo de tf-idf: lnc.ltn

Query: "best car insurance". Document: "car insurance auto insurance".

palavra	consulta					documento				produto
	tf-raw	tf-wght	df	idf	weight	tf-raw	tf-wght	weight	n'lized	
auto	0	0	5000	2.3	0	1	1	1	0.52	0
best	1	1	50000	1.3	1.3	0	0	0	0	0
car	1	1	10000	2.0	2.0	1	1	1	0.52	1.04
insurance	1	1	1000	3.0	3.0	2	1.3	1.3	0.68	2.04

Chave para as colunas: tf-raw: frequência absoluta do termo, sem ponderação, tf-wght: tf ponderada logaritmicamente, df: frequência de documentos, idf: frequência inversa de documentos, weight: O peso final do termo na consulta ou documento, n'lized: Peso de documentos após normalização por cosseno, product: O produto da ponderação final de consulta e documento

Escore final de similaridade entre consulta e documento:
$\sum_i w_{qi} \cdot w_{di} = 0 + 0 + 1.04 + 2.04 = 3.08$

Exemplo de tf-idf: lnc.ltn

Query: "best car insurance". Document: "car insurance auto insurance".

palavra	consulta					documento				produto
	tf-raw	tf-wght	df	idf	weight	tf-raw	tf-wght	weight	n'lized	
auto	0	0	5000	2.3	0	1	1	1	0.52	0
best	1	1	50000	1.3	1.3	0	0	0	0	0
car	1	1	10000	2.0	2.0	1	1	1	0.52	1.04
insurance	1	1	1000	3.0	3.0	2	1.3	1.3	0.68	2.04

Chave para as colunas: tf-raw: frequência absoluta do termo, sem ponderação, tf-wght: tf ponderada logaritmicamente, df: frequência de documentos, idf: frequência inversa de documentos, weight: O peso final do termo na consulta ou documento, n'lized: Peso de documentos após normalização por cosseno, product: O produto da ponderação final de consulta e documento

Escore final de similaridade entre consulta e documento:
$\sum_i w_{qi} \cdot w_{di} = 0 + 0 + 1.04 + 2.04 = 3.08$

Perguntas?

Resumo: Recuparação rankeada no modelo de espaço vetorial

Resumo: Recuparação rankeada no modelo de espaço vetorial

- Represente a consulta como um vetor ponderado de tf-idf

Resumo: Recuparação rankeada no modelo de espaço vetorial

- Represente a consulta como um vetor ponderado de tf-idf
- Represente cada documento como um vetor ponderado de tf-idf

Resumo: Recuparação rankeada no modelo de espaço vetorial

- Represente a consulta como um vetor ponderado de tf-idf
- Represente cada documento como um vetor ponderado de tf-idf
- Calcule a similaridade por cosseno entre o vetor da consulta e os vetores de cada documento

Resumo: Recuparação rankeada no modelo de espaço vetorial

- Represente a consulta como um vetor ponderado de tf-idf
- Represente cada documento como um vetor ponderado de tf-idf
- Calcule a similaridade por cosseno entre o vetor da consulta e os vetores de cada documento
- Rankeie os documentos com respeito à consulta

Resumo: Recuparação rankeada no modelo de espaço vetorial

- Represente a consulta como um vetor ponderado de tf-idf
- Represente cada documento como um vetor ponderado de tf-idf
- Calcule a similaridade por cosseno entre o vetor da consulta e os vetores de cada documento
- Rankeie os documentos com respeito à consulta
- Retorne os top K (p.ex., $K = 10$) ao usuário

Moral da estória de hoje

- Rankeamento dos resultados da busca: por que é importante (em comparação com apresentar um conjunto desordenado de resultados Booleanos)
- Frequência do termo: Ingrediente chave do ranqueamento
- Rankeamento por Tf-idf: Melhor rankeamento tradicional.
- Modelo de espaço vetorial: Um dos mais importantes modelos para recuperação de informação, juntamente com o modelo Booleano e o probabiliístico

Material

- Capítulos 6 and 7 do livro
- Mais materiais em `http://ifnlp.org/ir`
  - Vector space for dummies
  - Exploring the similarity space (Moffat and Zobel, 2005)
  - Okapi BM25 (a state-of-the-art weighting method, 11.4.3 of IIR)