Recuperação de Informações - Parte 2

Júlio César Batista

FURB - Universidade de Blumenau

Fevereiro, 2019

Agenda

- Motores de Busca
  - Índices
  - Consultas *booleanas*
  - Correção Ortográfica
  - Resultados Ordenados
- Imagens
  - Representação
  - *Template Matching*
  - *OCR*

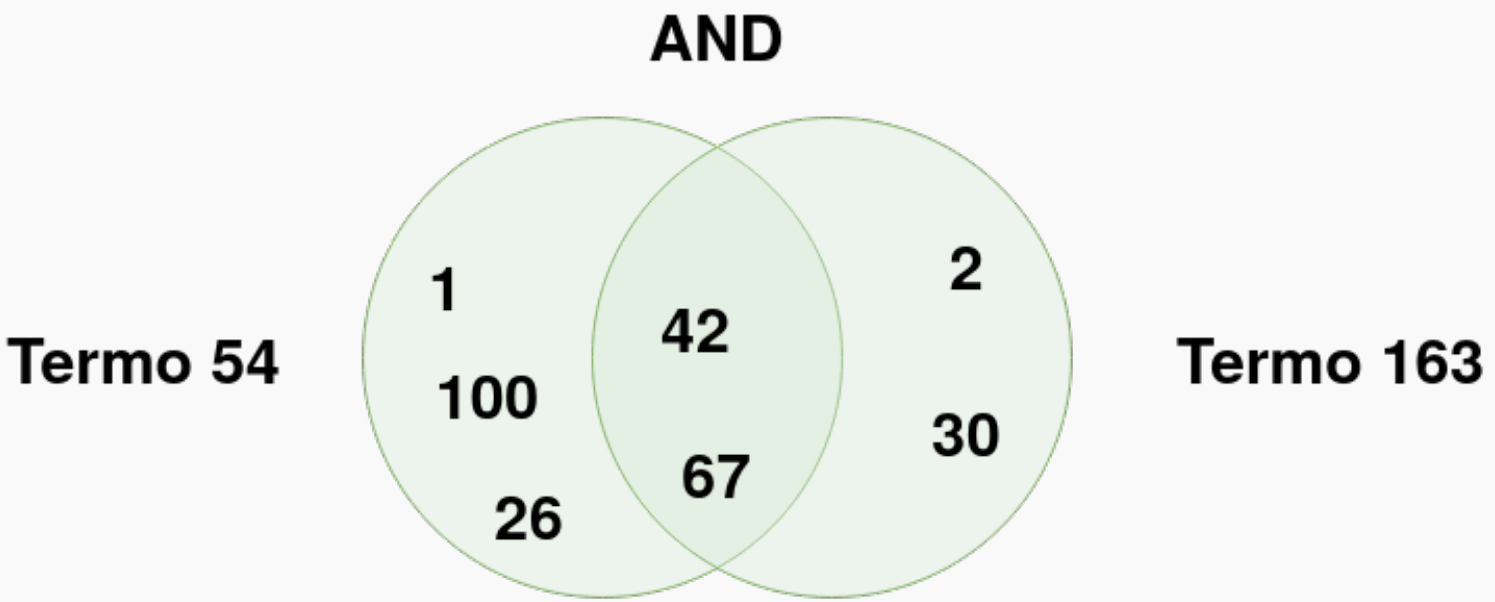
Índice invertido

- O problema de usar uma matriz termo-documento é que usaremos muita memória para armazenar muitos *0*s
- O índice invertido é um dicionário de termos para listas ordenadas de documentos

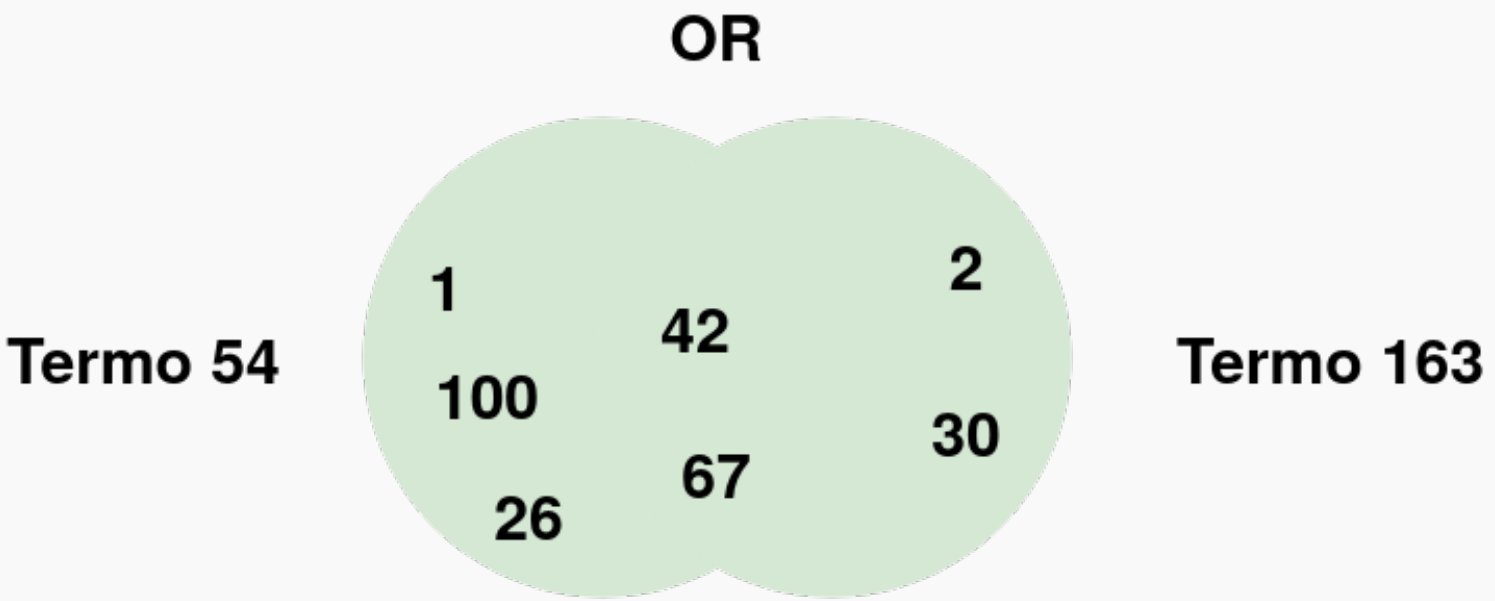
Índice invertido

```
documentos = [
    "The Who is a rock band", # 1
    "Only in the darkness can you see the stars." # 2
]

I = {
    'the': [1, 2],
    'who': [1],
    'is': [1],
    # ...
    # ...
    'can': [2],
    'see': [2],
    'stars': [2]
}

```

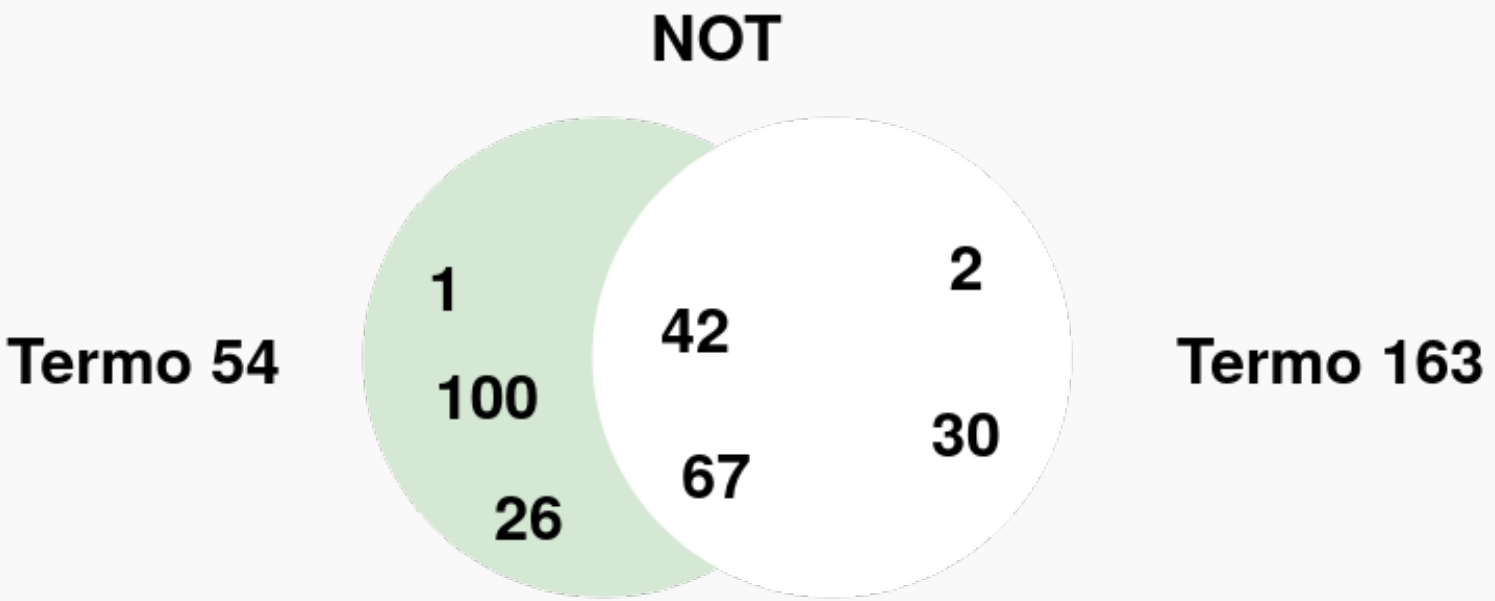
Operação AND - Intersecção de conjuntos



Operação OR - União de conjuntos



Operação NOT - Diferença de conjuntos



Operação AND - Algoritmo

```
INTERSECT(p1, p2)
 1  answer ← ⟨ ⟩
 2  while p1 ≠ NIL and p2 ≠ NIL
 3  do if docID(p1) = docID(p2)
 4        then ADD(answer, docID(p1))
 5             p1 ← next(p1)
 6             p2 ← next(p2)
 7        else if docID(p1) < docID(p2)
 8                then p1 ← next(p1)
 9                else p2 ← next(p2)
10  return answer
```

Figure: Intersecção de listas. Fonte: Introduction to Information Retrieval

Correção ortográfica

- Dado um *termo* inexistente no vocabulário
- Encontre o *termo* mais próximo no vocabulário

Correção ortográfica

- De forma geral, não é toda a palavra que está incorreta
- Exemplo: *comesso*, troca de *ç* por *ss*

Correção ortográfica

- De forma geral, não é toda a palavra que está incorreta
- Exemplo: *comesso*, troca de *ç* por *ss*
- Que tal comparar *substrings* (*k-grams*) de *comesso* com *começo*?

Correção ortográfica - k-grams

- 2-grams de *comesso*
- *$c*, *co*, *om*, *me*, *es*, *ss*, *so*, *o$*

Correção ortográfica - k-grams

- 3-grams de *comesso*
- *$$c*, *$co*, *com*, *ome*, *mes*, *ess*, *sso*, *so$*, *o$$*

Correção ortográfica - Comparação de k-grams

- 3-grams de *comesso*
- *$$c*, *$co*, *com*, *ome*, *mes*, *ess*, *sso*, *so$*, *o$$*
- 3-grams de *começo*
- *$$c*, *$co*, *com*, *ome*, *meç*, *eço*, *ço$*, *o$$*

Correção ortográfica - Comparação de k-grams

- Similaridade: Razão da Quantidade de *k-grams* em comum pela Quantidade total de *k-grams*
- Comum (5): *$$c*, *$co*, *com*, *ome*, *o$$*
- Total (12): *$$c*, *$co*, *com*, *ome*, *meç*, *eço*, *ço$*, *o$$*, *mes*, *ess*, *sso*, *so$*
- Similaridade = $\frac{5}{12}$

Correção ortográfica - Comparação de k-grams

- Coeficiente de Jaccard (Intersecção sobre Unisão)
- A = *$$c*, *$co*, *com*, *ome*, *mes*, *ess*, *sso*, *so$*, *o$$*
- B = *$$c*, *$co*, *com*, *ome*, *meç*, *eço*, *ço$*, *o$$*
- $\frac{|A \cap B|}{|A \cup B|} = \frac{5}{12}$

Correção ortográfica - Otimização

- Comparar contra todos os *termos* no vocabulário custa muito
- Ideal é comparar apenas com os termos que compartilham ao menos um *k-gram* em comum

Correção ortográfica - Otimização

- Comparar contra todos os *termos* no vocabulário custa muito
- Ideal é comparar apenas com os termos que compartilham ao menos um *k-gram* em comum
- Solução: Construir um índice de *k-grams* que mapeia *k-grams* para *termos*

Correção ortográfica - Índice k-grams

- $$c: começo, capaz, comer, correr, ...
- com: comigo, comando, começo, ...
- ome: homem, começo, fome, ...
- o$$: moço, carro, pescoço, começo, ...

Correção ortográfica - Algoritmo

- Se o *termo* existe no vocabulário, retorna
- Senão, computa os *k-grams* do *termo*
- Encontra os *termos* que compartilham ao menos um *k-gram* utilizando o índice de *k-grams*
- Calcula o Coeficiente de Jaccard para todos os *termos* candidatos
- Retorna o *termo* com maior Coeficiente de Jaccard

Correção ortográfica - Distância de Levenshtein

- Distância entre strings
- Distância: Custo de adicionar, remover ou trocar caractéres

Correção ortográfica - Distância de Levenshtein

	ϵ	c	o	m	e	s	s	o
ϵ	0							
c								
o								
m								
e								
ç								
o								

Correção ortográfica - Distância de Levenshtein

	ϵ	c	o	m	e	s	s	o
ϵ	0	1	2	3	4	5	6	7
c	1							
o	2							
m	3							
e	4							
ç	5							
o	6							

Correção ortográfica - Distância de Levenshtein

	ϵ	c	o	m	e	s	s	o
ϵ	0	1	2	3	4	5	6	7
c	1	$D_{i,j}$						
o	2							
m	3							
e	4							
ç	5							
o	6							

$$D_{i,j} = \min \begin{cases} D_{i-1,j} + 1 \\ D_{i,j-1} + 1 \\ D_{i-1,j-1} + 1_{a_i \neq b_j} \end{cases}$$

Correção ortográfica - Distância de Levenshtein

	ϵ	c	o	m	e	s	s	o
ϵ	0	1	2	3	4	5	6	7
c	1	$D_{i,j}$						
o	2							
m	3							
e	4							
ç	5							
o	6							

$$D_{i,j} = \min \begin{cases} D_{i-1,j} + 1 \\ D_{i,j-1} + 1 \\ D_{i-1,j-1} + 1_{a_i \neq b_j} \end{cases} = \min \begin{cases} 1 + 1 \\ 1 + 1 \\ 0 + 1_{c \neq c} \end{cases}$$

Correção ortográfica - Distância de Levenshtein

	ϵ	c	o	m	e	s	s	o
ϵ	0	1	2	3	4	5	6	7
c	1	0						
o	2							
m	3							
e	4							
ç	5							
o	6							

$$D_{i,j} = \min \begin{cases} D_{i-1,j} + 1 \\ D_{i,j-1} + 1 \\ D_{i-1,j-1} + 1_{a_i \neq b_j} \end{cases} = \min \begin{cases} 1 + 1 \\ 1 + 1 \\ 0 + 1_{c \neq c} \end{cases} = \min \begin{cases} 2 \\ 2 \\ 0 \end{cases}$$

Correção ortográfica - Distância de Levenshtein

	ϵ	c	o	m	e	s	s	o
ϵ	**0**	1	2	3	4	5	6	7
c	1	0	$D_{i,j}$					
o	2							
m	3							
e	4							
ç	5							
o	6							

$$D_{i,j} = \min \begin{cases} D_{i-1,j} + 1 \\ D_{i,j-1} + 1 \\ D_{i-1,j-1} + 1_{a_i \neq b_j} \end{cases} = \min \begin{cases} 2 + 1 \\ 0 + 1 \\ 1 + 1_{c \neq o} \end{cases} = \min \begin{cases} 3 \\ 1 \\ 2 \end{cases}$$

Correção ortográfica - Distância de Levenshtein

	ϵ	c	o	m	e	s	s	o
ϵ	**0**	1	2	3	4	5	6	7
c	1	0	1					
o	2							
m	3							
e	4							
ç	5							
o	6							

$$D_{i,j} = \min \begin{cases} D_{i-1,j} + 1 \\ D_{i,j-1} + 1 \\ D_{i-1,j-1} + 1_{a_i \neq b_j} \end{cases} = \min \begin{cases} 2 + 1 \\ 0 + 1 \\ 1 + 1_{c \neq o} \end{cases} = \min \begin{cases} 3 \\ 1 \\ 2 \end{cases}$$

Correção ortográfica - Distância de Levenshtein

	ϵ	c	o	m	e	s	s	o
ϵ	**0**	1	2	3	4	5	6	7
c	1	0	1					
o	2	$D_{i,j}$						
m	3							
e	4							
ç	5							
o	6							

$$D_{i,j} = \min \begin{cases} D_{i-1,j} + 1 \\ D_{i,j-1} + 1 \\ D_{i-1,j-1} + 1_{a_i \neq b_j} \end{cases} = \min \begin{cases} 0 + 1 \\ 2 + 1 \\ 1 + 1_{o \neq c} \end{cases} = \min \begin{cases} 1 \\ 3 \\ 2 \end{cases}$$

Correção ortográfica - Distância de Levenshtein

	ϵ	c	o	m	e	s	s	o
ϵ	**0**	1	2	3	4	5	6	7
c	1	0	1					
o	2	1						
m	3							
e	4							
ç	5							
o	6							

$$D_{i,j} = \min \begin{cases} D_{i-1,j} + 1 \\ D_{i,j-1} + 1 \\ D_{i-1,j-1} + 1_{a_i \neq b_j} \end{cases} = \min \begin{cases} 0 + 1 \\ 2 + 1 \\ 1 + 1_{o \neq c} \end{cases} = \min \begin{cases} 1 \\ 3 \\ 2 \end{cases}$$

Correção ortográfica - Distância de Levenshtein

	ϵ	c	o	m	e	s	s	o
ϵ	**0**	1	2	3	4	5	6	7
c	1	**0**	1	2	3	4	5	6
o	2	1	**0**	1	2	3	4	5
m	3	2	1	**0**	1	2	3	4
e	4	3	2	1	**0**	1	2	3
ç	5	4	3	2	1	**1**	**2**	3
o	6	5	4	3	2	2	2	**2**

Correção ortográfica - Algoritmo

- Se o *termo* existe no vocabulário, retorna
- Senão, computa os *k-grams* do *termo*
- Encontra os *termos* que compartilham ao menos um *k-gram* utilizando o índice de *k-grams*
- Calcula a Distância de Levenshtein para todos os *termos* candidatos
- Retorna o *termo* com menor Distância de Levenshtein

Correção ortográfica - Extras

- Como resolver *ph* no lugar de *f*?
- Como resolver *kuin* no lugar de *queen*?

Resultados Ordenados

- Consultas *booleanas* retornam todos os documentos que contém os termos
- Não existe ordem de importância no resultado

Resultados Ordenados

- Consultas *booleanas* retornam todos os documentos que contém os termos
- Não existe ordem de importância no resultado
- Alguns documentos são mais relevantes do que outros
- Usuários não querem ver os milhões de documentos disponíveis
- Idealmente, são vistos apenas os documentos nas primeiras páginas de uma consulta

Resultados Ordenados - Modelos

- $P(R = 1|D = d, Q = q)$
  - Probabilidade do documento d ser relevante para a consulta q
  - Ordena os resultados do maior para o menor
- $D(q, d)$
  - Distância da consulta a para o documento d
  - Ordena os resultados do menor para o maior

Resultados Ordenados - D(q, d)

- Não precisa de um método de aprendizado (não paramétrico)
- Distâncias "são fáceis" para interpretar
- Existem muitas funções de distância: Euclidiana, Manhattan, ...
- Pode usar uma função de similaridade [-1, 1]: Correlação, Similaridade de cosenos, ...

Resultados Ordenados - D(q, d)

- Consulta q e Documento d precisam ser vetores
- q pode ser um vetor binário (incidência)

$$q_i = \begin{cases} 1 & \text{se i-ésimo termo aparece em} q \\ 0 & \text{caso contrário} \end{cases}$$

Resultados Ordenados - D(q, d)

- Consulta $\mathbf{q}$ e Documento $\mathbf{d}$ precisam ser vetores
- d também pode ser um vetor binário (incidência)

$$\mathbf{d}_i = \begin{cases} 1 & \text{se i-ésimo termo aparece em} d \\ 0 & \text{caso contrário} \end{cases}$$

Resultados Ordenados - D(q, d)

- Consulta $\mathbf{q}$ e Documento $\mathbf{d}$ precisam ser vetores
- d também pode ser um vetor binário (incidência)

$$\mathbf{d}_i = \begin{cases} 1 & \text{se i-ésimo termo aparece em} d \\ 0 & \text{caso contrário} \end{cases}$$

- Assim é possível montar uma matriz de incidência termo-documento ($\mathbf{D}$)
- Sendo as colunas de $\mathbf{D}$ os vetores de incidência de um determinado documento

Resultados Ordenados - Matriz de incidência

- Doc 1: *The Who is a rock band*
- Doc 2: *Only in the darkness you can see the stars*

	Doc 1	Doc 2
The	1	1
Who	1	0
...	...	...
see	0	1
stars	0	1

Resultados Ordenados - Matriz com frequência dos termos

- **D** perde um poudo da informação de relevância
- A quantidade de vezes que um termo aparece em um documento é um sinal de relevância
- $tf_{i,j}$ indica a frequência do i-ésimo termo no j-ésimo documento

	Doc 1	Doc 2
The	1	2
Who	1	0
...	...	...
see	0	1
stars	0	1

Resultados Ordenados - Normalização

- Stop-words, normalmente, são as palavras mais frequentes em documentos
- Porém, são irrelevantes para as consultas
- É necessário normalizar a frequência, pela "importância"

Resultados Ordenados - Normalização

- Stop-words, normalmente, são as palavras mais frequentes em documentos
- Porém, são irrelevantes para as consultas
- É necessário normalizar a frequência, pela "importância"
- De certa forma, a importância de um termo é inversamente proporcional a quantidade de documentos em que ele aparece
  - Termos que aparecem em todos os documentos, devem ter pouco valor em definir o conteúdo (stop-words, por exemplo)
  - Termos que aparecem em poucos documentos, devem carregar uma certa informação sobre o conteúdo
  - Termos raros, aparecem em apenas um ou outro documento, são específicos do domínio ou não muito comuns no idioma

Resultados Ordenados - Normalização

- idf_i é o inverso da frequência de documentos para o i-ésimo termo

$$idf_i = log \frac{N}{df_i}$$

- N: Quantidade de documentos na coleção
- df_i: Quantidade de documentos que o i-ésimo termo aparece

Resultados Ordenados - D(q, d)

- Juntando $tf_{i,j}$ e idf_i temos a métrica $tf\text{-}idf_{i,j}$
- $tf\text{-}idf_{i,j} = tf_{i,j} \times idf_i$
- $\mathbf{D}_{i,j} = tf\text{-}idf_{i,j}$

	Doc 1	Doc 2
The	$tf\text{-}idf_{\text{The,Doc 1}}$	$tf\text{-}idf_{\text{The,Doc 2}}$
Who	$tf\text{-}idf_{\text{Who,Doc 1}}$	$tf\text{-}idf_{\text{Who,Doc 2}}$
...	...	...
see	$tf\text{-}idf_{\text{see,Doc 1}}$	$tf\text{-}idf_{\text{see,Doc 2}}$
stars	$tf\text{-}idf_{\text{stars,Doc 1}}$	$tf\text{-}idf_{\text{stars,Doc 2}}$

Resultados Ordenados - D(q, d)

- $tf\text{-}idf_{i,j} = tf_{i,j} \times idf_i$

	Doc 1	Doc 2
The	$1 \times log\frac{2}{2} = 0$	$2 \times log\frac{2}{2} = 0$
Who	$1 \times log\frac{2}{1} = 0.693$	$0 \times log\frac{2}{1} = 0$
...	...	...
see	$0 \times log\frac{2}{1} = 0$	$1 \times log\frac{2}{1} = 0.693$
stars	$0 \times log\frac{2}{1} = 0$	$1 \times log\frac{2}{1} = 0.693$

Resultados Ordenados - D(q, d)

- Distância Euclidiana (L_2-norm)
- $D(\mathbf{q}, \mathbf{d}) = ||\mathbf{q} - \mathbf{d}||_2 = \sqrt{\sum_i^n (\mathbf{q}_i - \mathbf{d}_i)^2}$
- Valores positivos
- Quanto mais próximo de 0, mais similar (menor distância)

Resultados Ordenados - D(q, d)

- Similaridade de cosenos
- $D(\mathbf{q}, \mathbf{d}) = \frac{\mathbf{q} \times \mathbf{d}}{||\mathbf{q}||_2 ||\mathbf{d}||_2}$
- Valores no intervalo [-1, 1]
- Sendo -1: Completamente opostos
- Sendo 1: Similares