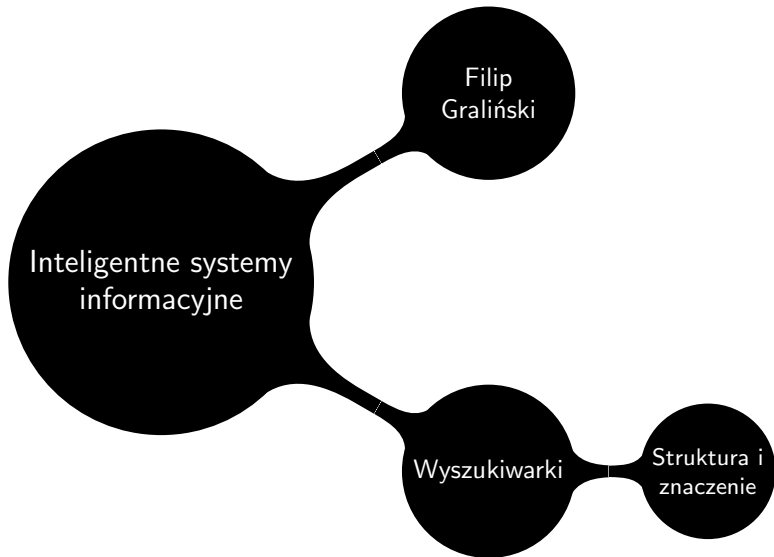


```
graph LR; A[Inteligentne systemy informacyjne] --- B[Filip Graliński]; A --- C[Wyszukiwarki]
```

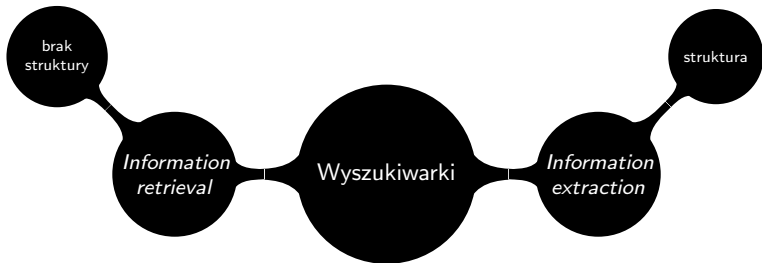
Inteligentne systemy
informacyjne

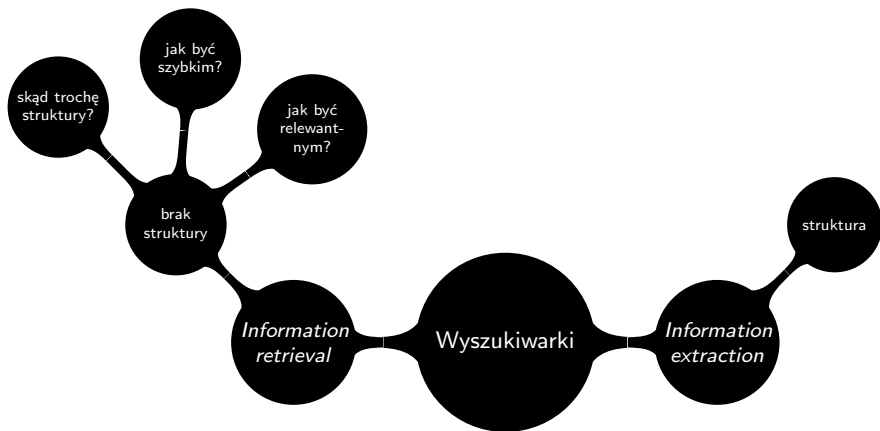
Filip
Graliński

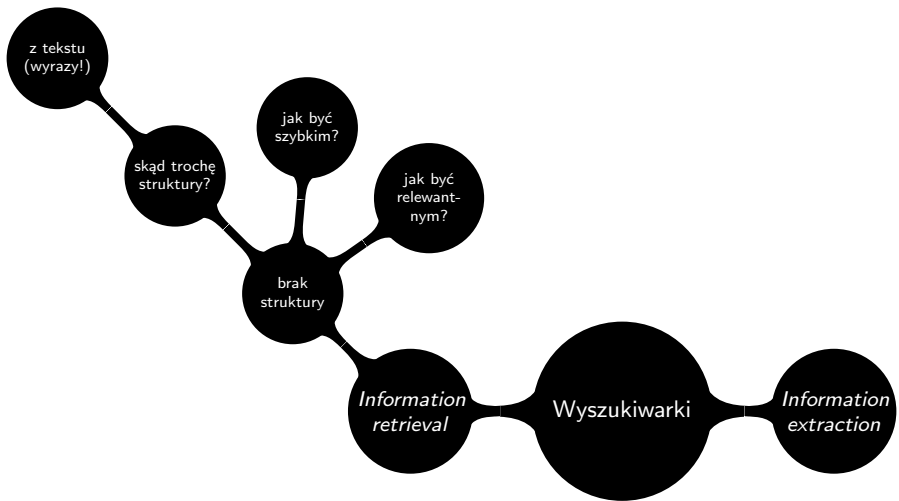
Wyszukiwarki



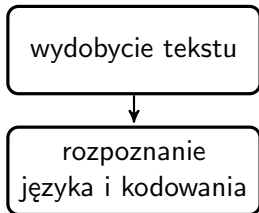




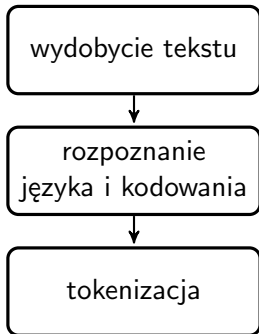




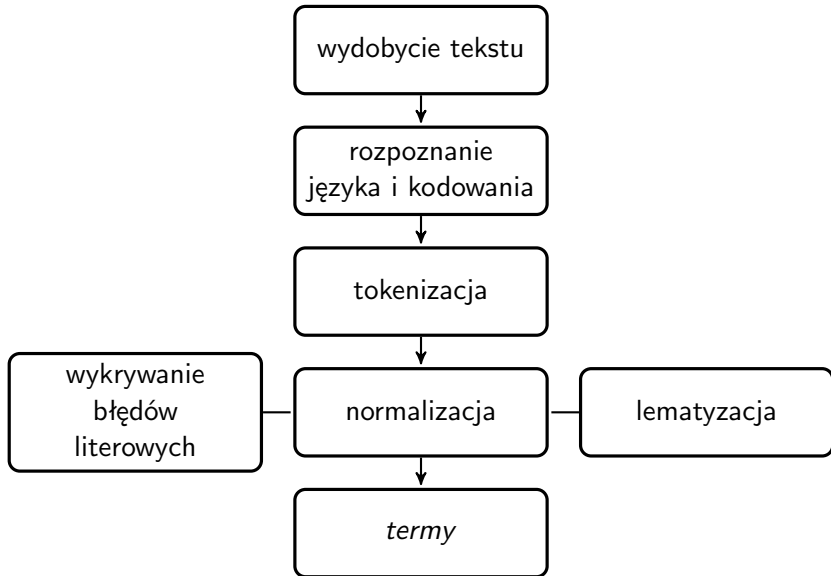
przetwarzanie dokumentu – od tekstu do *termów*



przetwarzanie dokumentu – od tekstu do *termów*



przetwarzanie dokumentu – od tekstu do *termów*



tokenizacja

Tokenizacja to podział na wyrazopodobne jednostki (**tokeny**)

- ▶ trudności dla języka angielskiego:

tokenizacja

Tokenizacja to podział na wyrazopodobne jednostki (**tokeny**)

- ▶ trudności dla języka angielskiego:
 - ▶ łącznik
 - ▶ apostrofy

tokenizacja

Tokenizacja to podział na wyrazopodobne jednostki (**tokeny**)

- ▶ trudności dla języka angielskiego:
 - ▶ łącznik
 - ▶ apostrofy
 - ▶ daty, numery telefonów, adresy e-mailowe etc.

tokenizacja

Tokenizacja to podział na wyrazopodobne jednostki (**tokeny**)

- ▶ trudności dla języka angielskiego:
 - ▶ łącznik
 - ▶ apostrofy
 - ▶ daty, numery telefonów, adresy e-mailowe etc.
 - ▶ tokeny typu *C++*, *C#*, *Yahoo!*

tokenizacja

Tokenizacja to podział na wyrazopodobne jednostki (**tokeny**)

- ▶ trudności dla języka angielskiego:
 - ▶ łącznik
 - ▶ apostrofy
 - ▶ daty, numery telefonów, adresy e-mailowe etc.
 - ▶ tokeny typu *C++*, *C#*, *Yahoo!*
- ▶ inne języki:
 - ▶ chiński – brak spacji

tokenizacja

Tokenizacja to podział na wyrazopodobne jednostki (**tokeny**)

- ▶ trudności dla języka angielskiego:
 - ▶ łącznik
 - ▶ apostrofy
 - ▶ daty, numery telefonów, adresy e-mailowe etc.
 - ▶ tokeny typu *C++*, *C#*, *Yahoo!*
- ▶ inne języki:
 - ▶ chiński – brak spacji
 - ▶ niemiecki – wyrazy złożone

tokenizacja

Tokenizacja to podział na wyrazopodobne jednostki (**tokeny**)

- ▶ trudności dla języka angielskiego:
 - ▶ łącznik
 - ▶ apostrofy
 - ▶ daty, numery telefonów, adresy e-mailowe etc.
 - ▶ tokeny typu *C++*, *C#*, *Yahoo!*
- ▶ inne języki:
 - ▶ chiński – brak spacji
 - ▶ niemiecki – wyrazy złożone
 - ▶ jęz. romańskie – zbitki typu *l'ordinateur*

Normalizacja

Czy powinny wyjść identyczne wyniki dla zapytań:

Uwaga! Spójność!

Zapytania powinny być normalizowane tak samo jak dokumenty

Normalizacja

Czy powinny wyjść identyczne wyniki dla zapytań:

▶ Łódź i Łódź ?

lowercasing

▶ koń i kon ?

usuwanie diakrytyków

Uwaga! Spójność!

Zapytania powinny być normalizowane tak samo jak dokumenty

Normalizacja

Czy powinny wyjść identyczne wyniki dla zapytań:

- ▶ Łódź i Łódź ? *lowercasing*
- ▶ koń i kon ? usuwanie diakrytyków
- ▶ legenda i legendzie ? lematyzacja

Uwaga! Spójność!

Zapytania powinny być normalizowane tak samo jak dokumenty

Normalizacja

Czy powinny wyjść identyczne wyniki dla zapytań:

- ▶ Łódź i Łódź ? *lowercasing*
- ▶ koń i kon ? usuwanie diakrytyków
- ▶ legenda i legendzie ? lematyzacja
- ▶ urządzić i urządzenie ?

Uwaga! Spójność!

Zapytania powinny być normalizowane tak samo jak dokumenty

Normalizacja

Czy powinny wyjść identyczne wyniki dla zapytań:

- ▶ Łódź i Łódź ? *lowercasing*
- ▶ koń i kon ? usuwanie diakrytyków
- ▶ legenda i legendzie ? lematyzacja
- ▶ urządzić i urządzenie ?
- ▶ legenda i legendowy ? rdzeniowanie

Uwaga! Spójność!

Zapytania powinny być normalizowane tak samo jak dokumenty

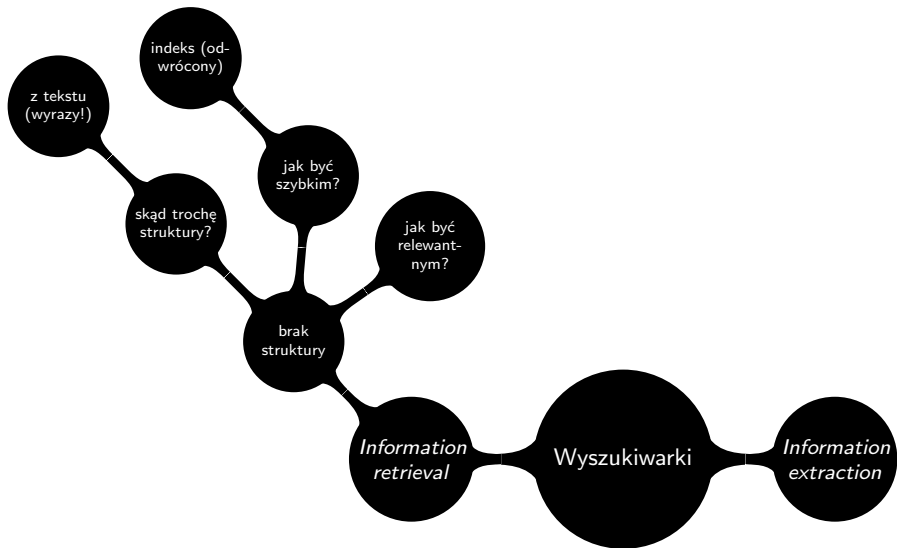
Normalizacja

Czy powinny wyjść identyczne wyniki dla zapytań:

- ▶ Łódź i Łódź ? *lowercasing*
- ▶ koń i kon ? usuwanie diakrytyków
- ▶ legenda i legendzie ? lematyzacja
- ▶ urządzić i urządzenie ?
- ▶ legenda i legendowy ? rdzeniowanie
- ▶ bałwan i bałwanek ?

Uwaga! Spójność!

Zapytania powinny być normalizowane tak samo jak dokumenty



odwrócony indeks (*inverted index*)

doc1		<i>Ala ma kota.</i>
doc2		<i>Podobno jest kot w butach.</i>
doc3		<i>Ty chyba masz kota!</i>
doc4		<i>But chyba zgubiłem.</i>

odwrócony indeks (*inverted index*)

doc1		<i>Ala ma kota.</i>
doc2		<i>Podobno jest kot w butach.</i>
doc3		<i>Ty chyba masz kota!</i>
doc4		<i>But chyba zgubiłem.</i>

stop words: jest, w

odwrócony indeks (*inverted index*)

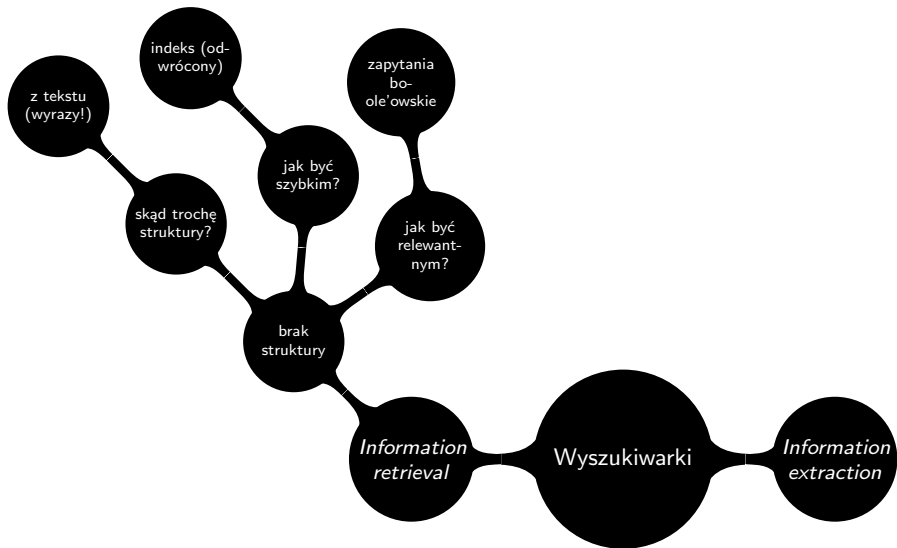
doc1		<i>Ala ma kota.</i>
doc2		<i>Podobno jest kot w butach.</i>
doc3		<i>Ty chyba masz kota!</i>
doc4		<i>But chyba zgubiłem.</i>

stop words: jest, w

<i>ala</i>	→	1	
<i>but</i>	→	2	4
<i>chyba</i>	→	3	4
<i>kot</i>	→	1	2 3
<i>mieć</i>	→	?	
<i>podobno</i>	→	2	
<i>ty</i>	→	3	
<i>zgubić</i>	→	4	

Lista wszystkich termów w kolekcji to **słownik** (*vocabulary*):

{*ala, but, chyba, kot, mieć, podobno, ty, zgubić*}



zapytania boole'owskie

- ▶ *pizzeria Poznań dowóz*

zapytania boole'owskie

- ▶ *pizzeria AND Poznań AND dowóz*

zapytania boole'owskie

- ▶ *pizzeria AND Poznań AND dowóz*
- ▶ *(pizzeria OR pizza OR tratoria) AND Poznań AND dowóz*

zapytania boole'owskie

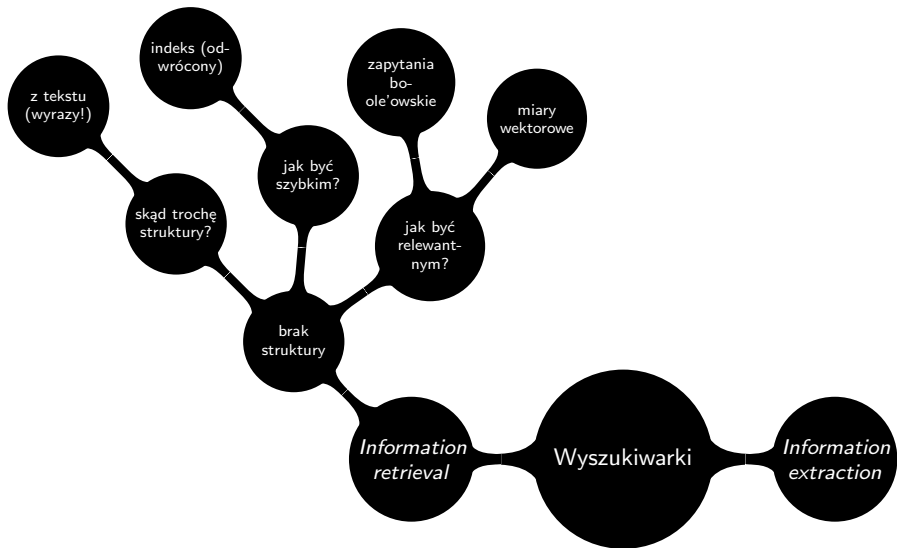
- ▶ *pizzeria AND Poznań AND dowóz*
- ▶ *(pizzeria OR pizza OR tratoria) AND Poznań AND dowóz*
- ▶ *pizzeria AND Poznań AND dowóz AND NOT golonka*

zapytania boole'owskie

- ▶ *pizzeria AND Poznań AND dowóz*
- ▶ *(pizzeria OR pizza OR tratoria) AND Poznań AND dowóz*
- ▶ *pizzeria AND Poznań AND dowóz AND NOT golonka*

Jak domyślnie interpretować zapytanie?

- ▶ jako zapytanie AND – być może za mało dokumentów
- ▶ rozwiązanie pośrednie?
- ▶ jako zapytanie OR – być może za dużo dokumentów



dokument jako wektor

doc1	<i>Ala ma kota.</i>
doc2	<i>Podobno jest kot w butach.</i>
doc3	<i>Ty chyba masz kota!</i>
doc4	<i>But chyba zgubiłem.</i>

dokument jako wektor

doc1	<i>Ala ma kota.</i>
doc2	<i>Podobno jest kot w butach.</i>
doc3	<i>Ty chyba masz kota!</i>
doc4	<i>But chyba zgubiłem.</i>

	<i>ala</i>	<i>but</i>	<i>chyba</i>	<i>kot</i>	<i>mieć</i>	<i>podobno</i>	<i>ty</i>	<i>zgubić</i>
doc1	1	0	0	1	1	0	0	0
doc2	0	1	0	1	0	1	0	0
doc3	0	0	1	1	1	0	1	0
doc4	0	1	1	0	0	0	0	1

dokument jako wektor

doc1	<i>Ala ma kota.</i>
doc2	<i>Podobno jest kot w butach.</i>
doc3	<i>Ty chyba masz kota!</i>
doc4	<i>But chyba zgubiłem.</i>
doc5	<i>Kot ma kota.</i>

	<i>ala</i>	<i>but</i>	<i>chyba</i>	<i>kot</i>	<i>mieć</i>	<i>podobno</i>	<i>ty</i>	<i>zgubić</i>
doc1	1	0	0	1	1	0	0	0
doc2	0	1	0	1	0	1	0	0
doc3	0	0	1	1	1	0	1	0
doc4	0	1	1	0	0	0	0	1

dokument jako wektor

doc1	<i>Ala ma kota.</i>
doc2	<i>Podobno jest kot w butach.</i>
doc3	<i>Ty chyba masz kota!</i>
doc4	<i>But chyba zgubiłem.</i>
doc5	<i>Kot ma kota.</i>

	<i>ala</i>	<i>but</i>	<i>chyba</i>	<i>kot</i>	<i>mieć</i>	<i>podobno</i>	<i>ty</i>	<i>zgubić</i>
doc1	1	0	0	1	1	0	0	0
doc2	0	1	0	1	0	1	0	0
doc3	0	0	1	1	1	0	1	0
doc4	0	1	1	0	0	0	0	1
doc5	0	0	0	?	1	0	0	0

jak uwzględnić frekwencję wyrazu?

$tf_{t,d}$

jak uwzględnić frekwencję wyrazu?

$$tf_{t,d}$$

$$1 + \log(tf_{t,d})$$

jak uwzględnić frekwencję wyrazu?

$$tf_{t,d}$$

$$1 + \log(tf_{t,d})$$

$$\begin{cases} 1, & \text{jeśli } tf_{t,d} > 0 \\ 0, & \text{w przeciwnym razie} \end{cases}$$

jak uwzględnić frekwencję wyrazu?

$$tf_{t,d}$$

$$1 + \log(tf_{t,d})$$

$$\begin{cases} 1, & \text{jeśli } tf_{t,d} > 0 \\ 0, & \text{w przeciwnym razie} \end{cases}$$

$$0.5 + \frac{0.5 \times tf_{t,d}}{\max_t(tf_{t,d})}$$

odwrotna częstość w dokumentach

Czy wszystkie wyrazy są tak samo ważne?

odwrotna częstość w dokumentach

Czy wszystkie wyrazy są tak samo ważne?

NIE. Wyrazy pojawiające się w wielu dokumentach są mniej ważne.

odwrotna częstość w dokumentach

Czy wszystkie wyrazy są tak samo ważne?

NIE. Wyrazy pojawiające się w wielu dokumentach są mniej ważne.

Aby to uwzględnić, przemnażamy frekwencję wyrazu przez **odwrotną częstość w dokumentach** (*inverse document frequency*):

$$\text{idf}_t = \log \frac{N}{\text{df}_t}$$

,

idf_t – odwrotna częstość wyrazu t w dokumentach

N – liczba dokumentów w kolekcji

df_f – w ilu dokumentach wystąpił wyraz t ?

odwrotna częstość w dokumentach (cd.)

wyraz wystąpił...	$idf_t = \dots$
w 1 dokumencie	

odwrotna częstość w dokumentach (cd.)

wyraz wystąpił...	$\text{idf}_t = \dots$
w 1 dokumencie	$= \log N/1 = \log N$

odwrotna częstość w dokumentach (cd.)

wyraz wystąpił...	$idf_t = \dots$
w 1 dokumencie	$= \log N/1 = \log N$
2 razy w kolekcji	$= \log N/2$ lub $\log N$

odwrotna częstość w dokumentach (cd.)

wyraz wystąpił...	$idf_t = \dots$
w 1 dokumencie	$= \log N/1 = \log N$
2 razy w kolekcji	$= \log N/2$ lub $\log N$
w połowie dokumentów	$= \log N/(N/2) = \log 2$

odwrotna częstość w dokumentach (cd.)

wyraz wystąpił...	$idf_t = \dots$
w 1 dokumencie	$= \log N/1 = \log N$
2 razy w kolekcji	$= \log N/2$ lub $\log N$
w połowie dokumentów	$= \log N/(N/2) = \log 2$
we wszystkich dokumentach	$= \log N/N = \log 1 = 0$

odwrotna częstość w dokumentach (cd.)

wyraz wystąpił...	$idf_t = \dots$
w 1 dokumencie	$= \log N/1 = \log N$
2 razy w kolekcji	$= \log N/2$ lub $\log N$
w połowie dokumentów	$= \log N/(N/2) = \log 2$
we wszystkich dokumentach	$= \log N/N = \log 1 = 0$

Zamiast $tf_{t,d}$ będziemy w wektorach rozpatrywać wartości:

$$\text{tf-idf}_{t,d} = \text{tf}_{t,d} \times \text{idf}_t$$

odwrotna częstość w dokumentach (cd.)

wyraz wystąpił...	$idf_t = \dots$
w 1 dokumencie	$= \log N/1 = \log N$
2 razy w kolekcji	$= \log N/2$ lub $\log N$
w połowie dokumentów	$= \log N/(N/2) = \log 2$
we wszystkich dokumentach	$= \log N/N = \log 1 = 0$

Zamiast $tf_{t,d}$ będziemy w wektorach rozpatrywać wartości:

$$\text{tf-idf}_{t,d} = \text{tf}_{t,d} \times \text{idf}_t$$

Overlap score measure:

$$\text{Score}(q, d) = \sum_{t \in q} \text{tf-idf}_{t,d}$$