INFORMATION RETRIEVAL

Week 9 – Champion Lists

09.05.2025

Severin Mills

Today

1

Exercise Recap

- Vector Space Model
- Questions

2

Theory

- Inexact Top-K Retrieval
- Champions Lists

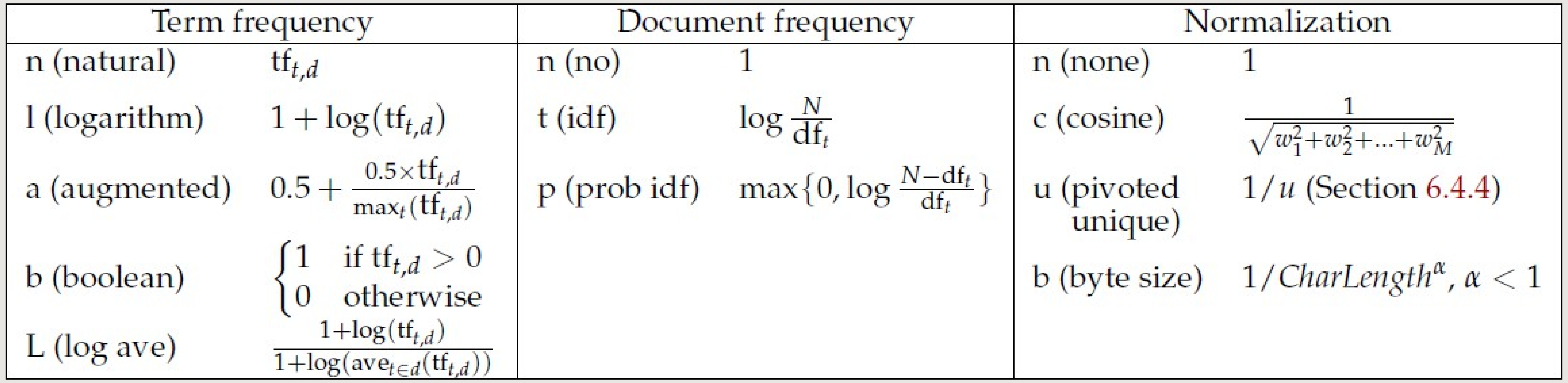
3

Kahoot

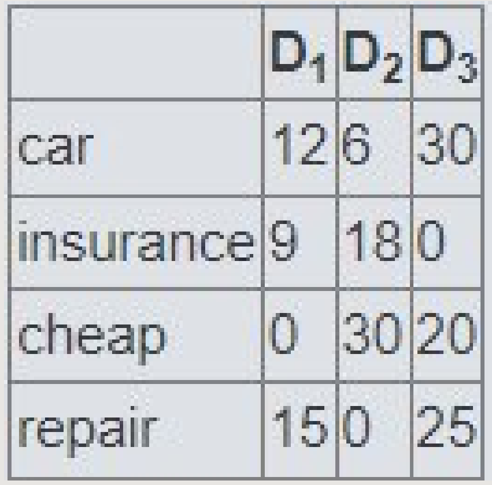
Exercise 8: Champion Lists

Exercise 7: Vector Space Model

Recap



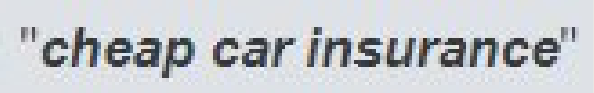
Term frequency		Document frequency		Normalization	
n (natural)	$\mathrm{tf}_{t,d}$	n (no)	1	n (none)	1
l (logarithm)	$1 + \log(\mathrm{tf}_{t,d})$	t (idf)	$\log \frac{N}{\mathrm{df}_t}$	c (cosine)	$\frac{1}{\sqrt{w_1^2+w_2^2+\ldots+w_M^2}}$
a (augmented)	$0.5 + \frac{0.5\times\mathrm{tf}_{t,d}}{\max_t(\mathrm{tf}_{t,d})}$	p (prob idf)	$\max\{0, \log \frac{N-\mathrm{df}_t}{\mathrm{df}_t}\}$	u (pivoted unique)	$1/u$ (Section 6.4.4)
b (boolean)	$\begin{cases} 1 & \text{if } \mathrm{tf}_{t,d} > 0 \\ 0 & \text{otherwise} \end{cases}$			b (byte size)	$1/CharLength^{\alpha}$, $\alpha < 1$
L (log ave)	$\frac{1+\log(\mathrm{tf}_{t,d})}{1+\log(\mathrm{ave}_{t\in d}(\mathrm{tf}_{t,d}))}$				



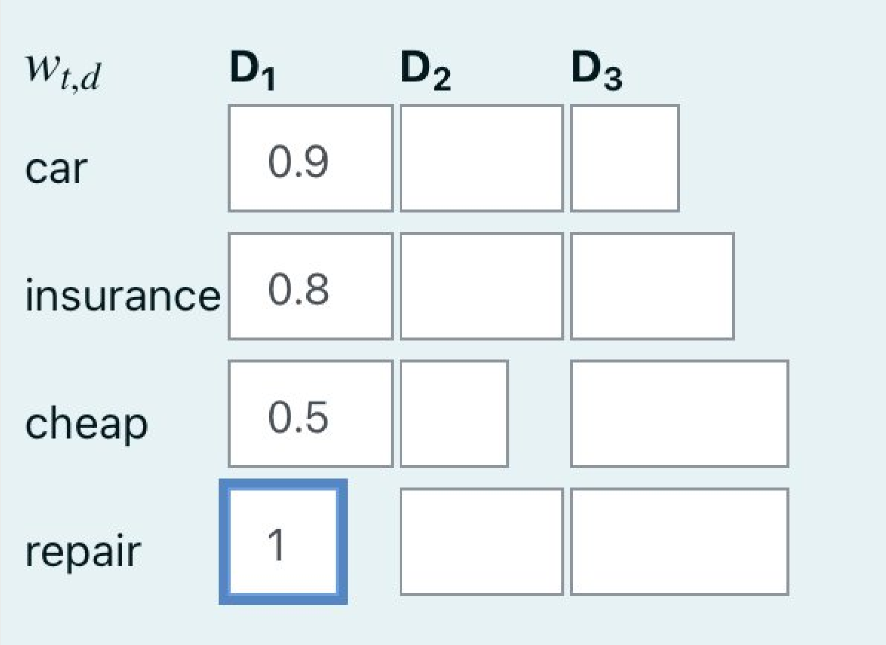
	D_1	D_2	D_3
car	12	6	30
insurance	9	18	0
cheap	0	30	20
repair	15	0	25

$\mathrm{tf}_{t,d}$

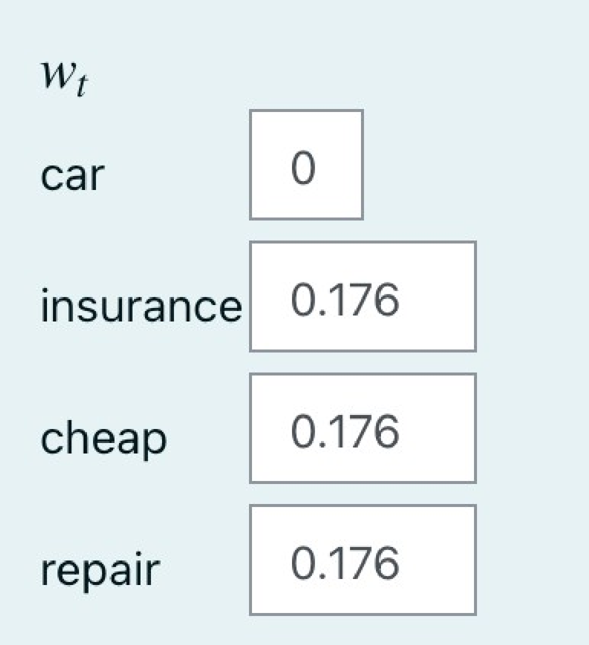
atc.nnn



"*cheap car insurance*"

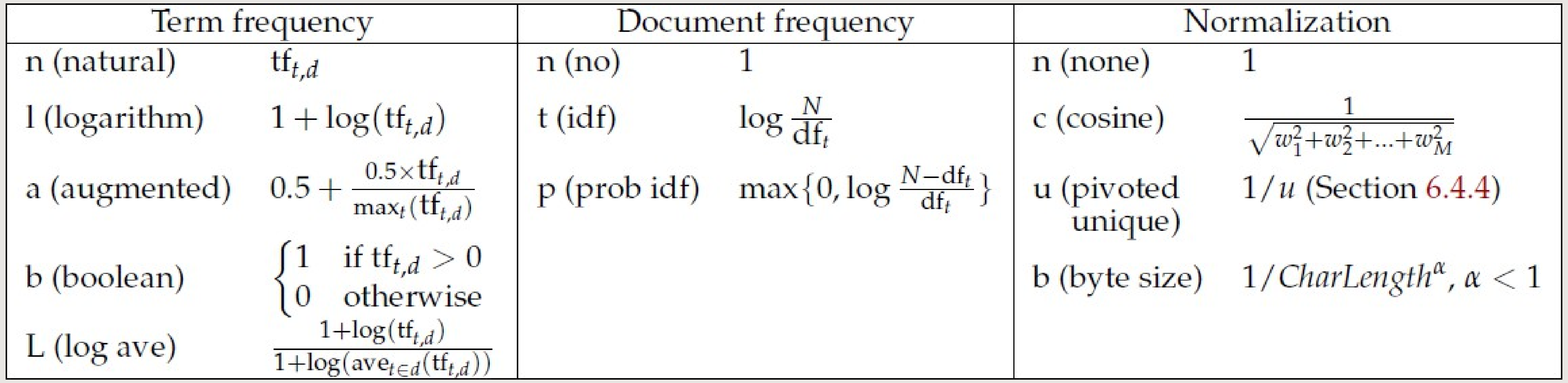


$w_{t,d}$	D_1	D_2	D_3
car	0.9		
insurance	0.8		
cheap	0.5		
repair	1		

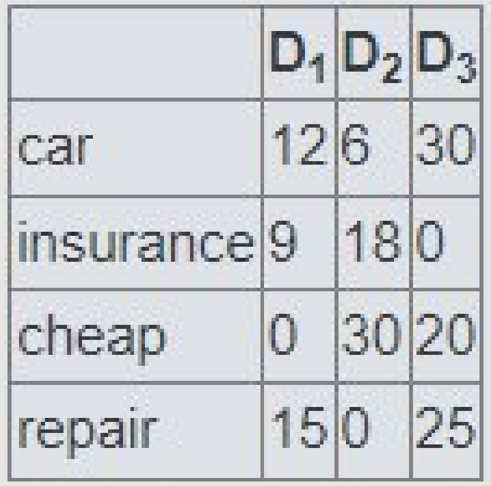


w_t	
car	0
insurance	0.176
cheap	0.176
repair	0.176

Recap

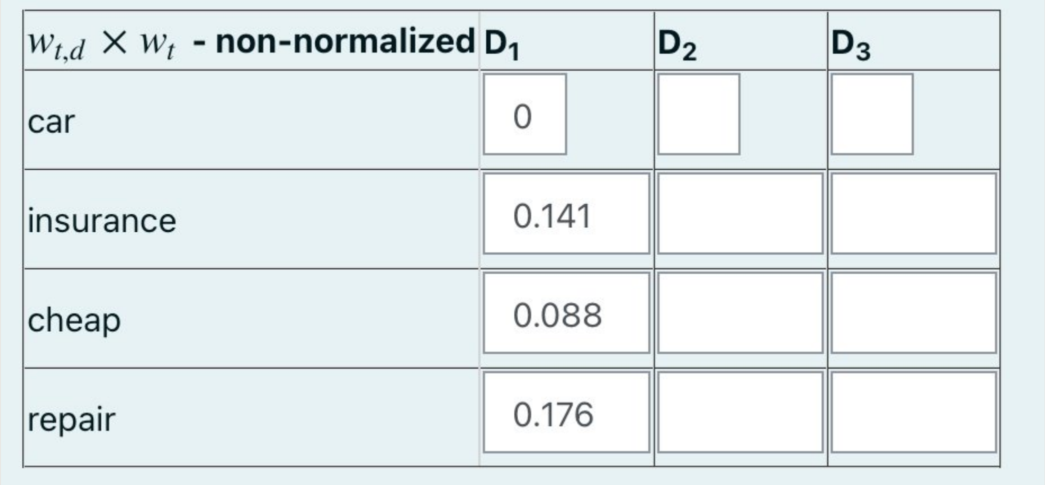


Term frequency		Document frequency		Normalization	
n (natural)	$\mathrm{tf}_{t,d}$	n (no)	1	n (none)	1
l (logarithm)	$1+\log(\mathrm{tf}_{t,d})$	t (idf)	$\log\frac{N}{\mathrm{df}_t}$	c (cosine)	$\frac{1}{\sqrt{w_1^2+w_2^2+\ldots+w_M^2}}$
a (augmented)	$0.5+\frac{0.5\times\mathrm{tf}_{t,d}}{\max_t(\mathrm{tf}_{t,d})}$	p (prob idf)	$\max\{0,\log\frac{N-\mathrm{df}_t}{\mathrm{df}_t}\}$	u (pivoted unique)	$1/u$ (Section 6.4.4)
b (boolean)	$\begin{cases}1 & \text{if } \mathrm{tf}_{t,d}>0\\ 0 & \text{otherwise}\end{cases}$			b (byte size)	$1/\mathit{CharLength}^{\alpha}$, $\alpha<1$
L (log ave)	$\frac{1+\log(\mathrm{tf}_{t,d})}{1+\log(\mathrm{ave}_{t\in d}(\mathrm{tf}_{t,d}))}$				

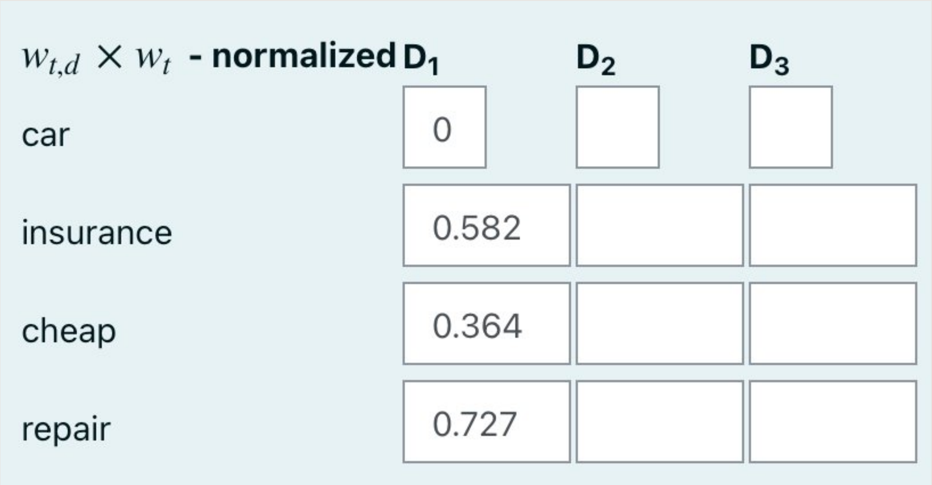


	D_1	D_2	D_3
car	12	6	30
insurance	9	18	0
cheap	0	30	20
repair	15	0	25

$\mathrm{tf}_{t,d}$

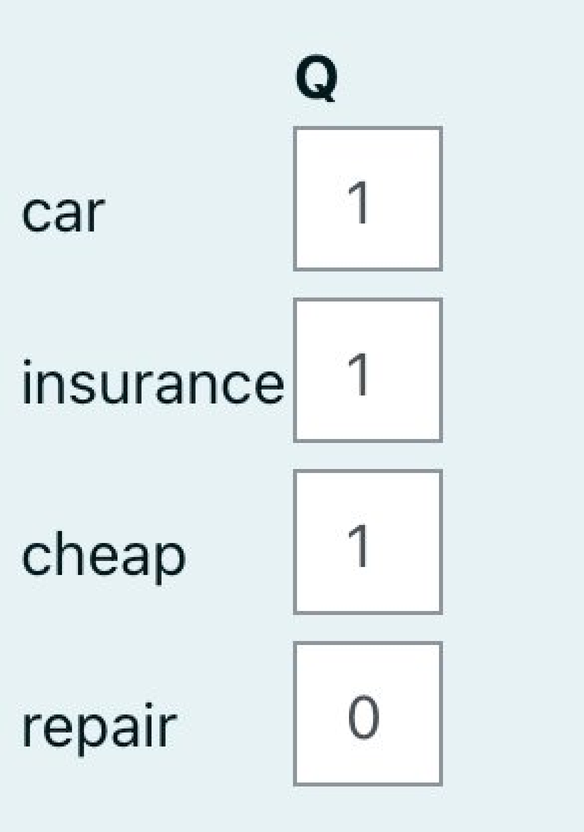


$w_{t,d}\times w_t$ **- non-normalized**	D_1	D_2	D_3
car	0		
insurance	0.141		
cheap	0.088		
repair	0.176		

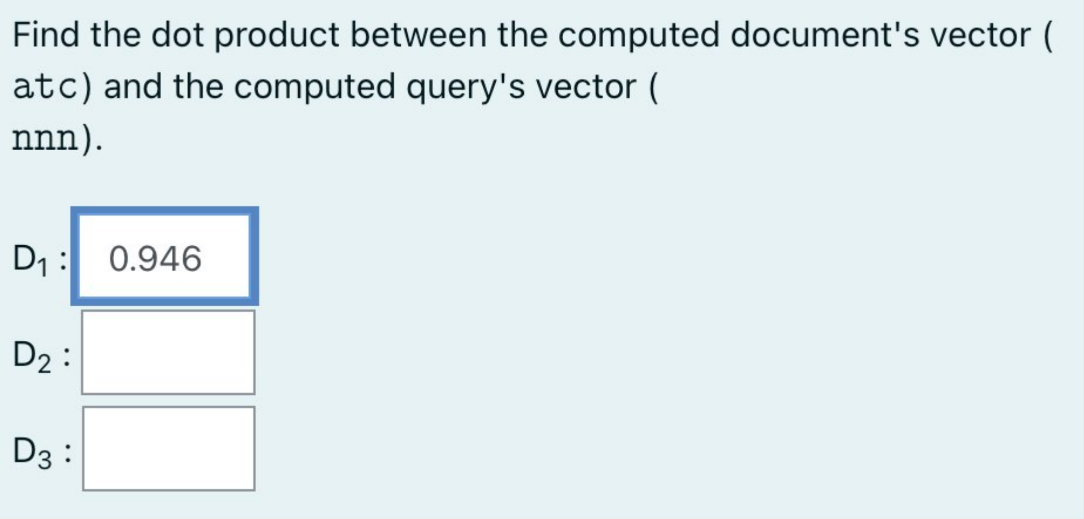


$w_{t,d}\times w_t$ **- normalized**	D_1	D_2	D_3
car	0		
insurance	0.582		
cheap	0.364		
repair	0.727		

Recap



	Q
car	1
insurance	1
cheap	1
repair	0



Find the dot product between the computed document's vector (
atc) and the computed query's vector (
nnn).

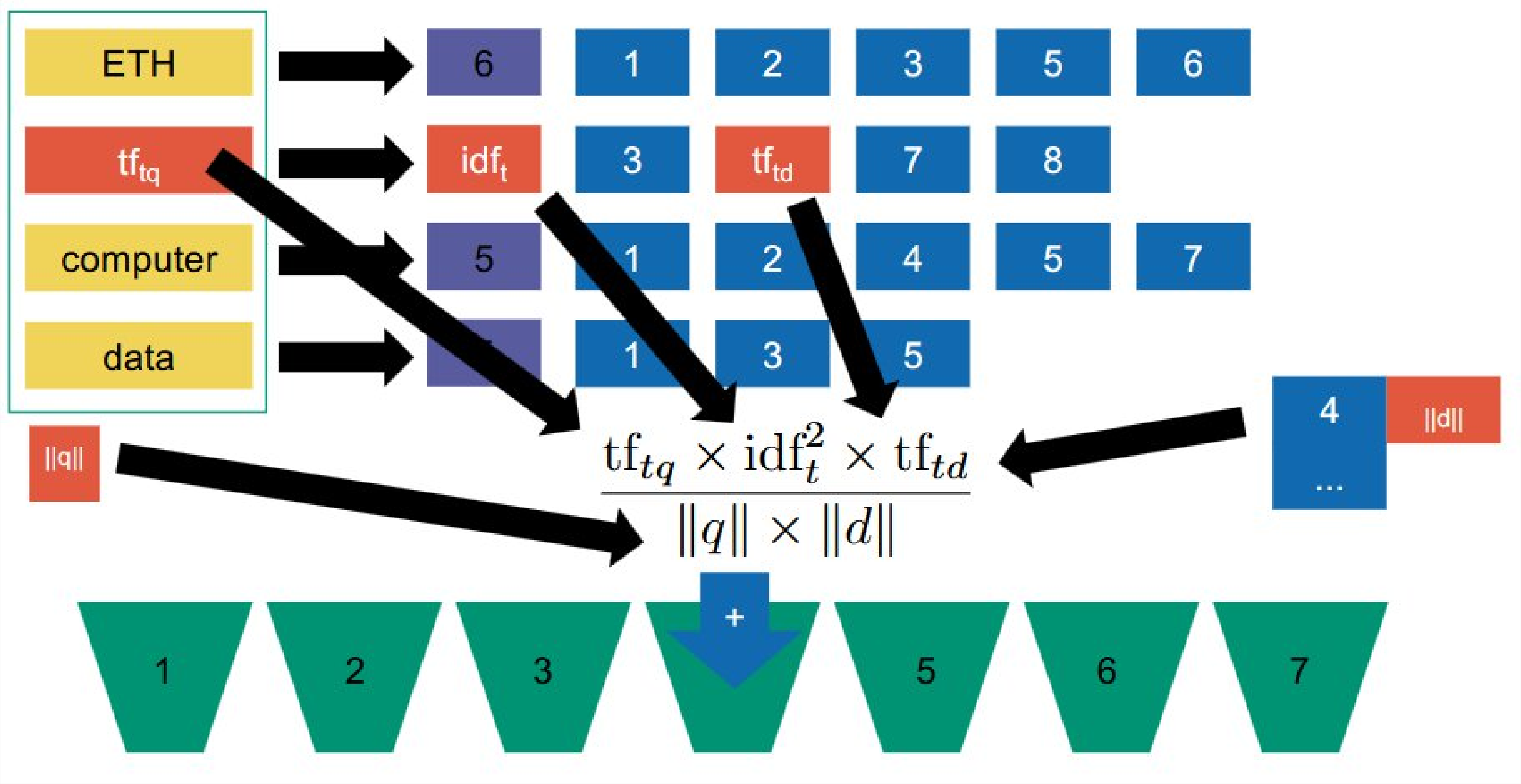
D_1 : 0.946

D_2 :

D_3 :

Recap: Vector Space Model

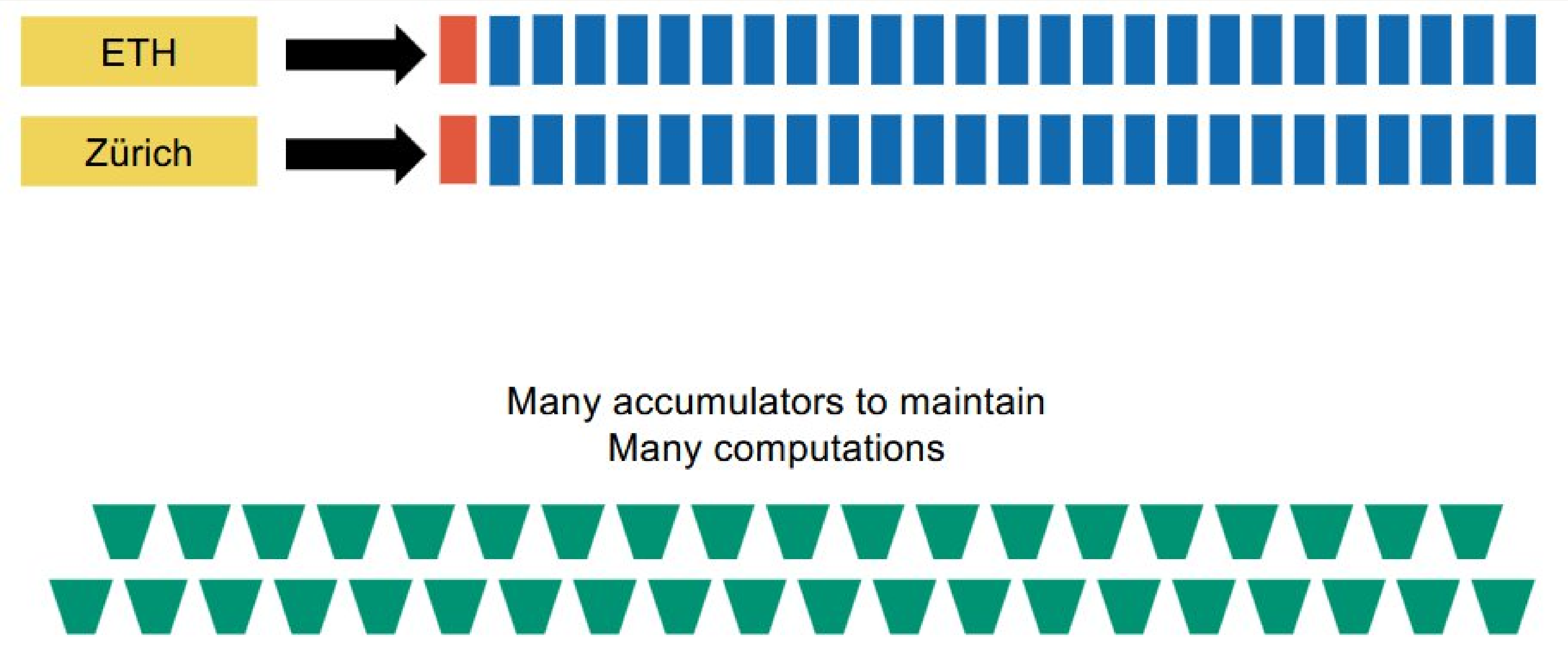
Inverted Index



$$\frac{\mathrm{tf}_{tq} \times \mathrm{idf}_t^2 \times \mathrm{tf}_{td}}{\|q\| \times \|d\|}$$

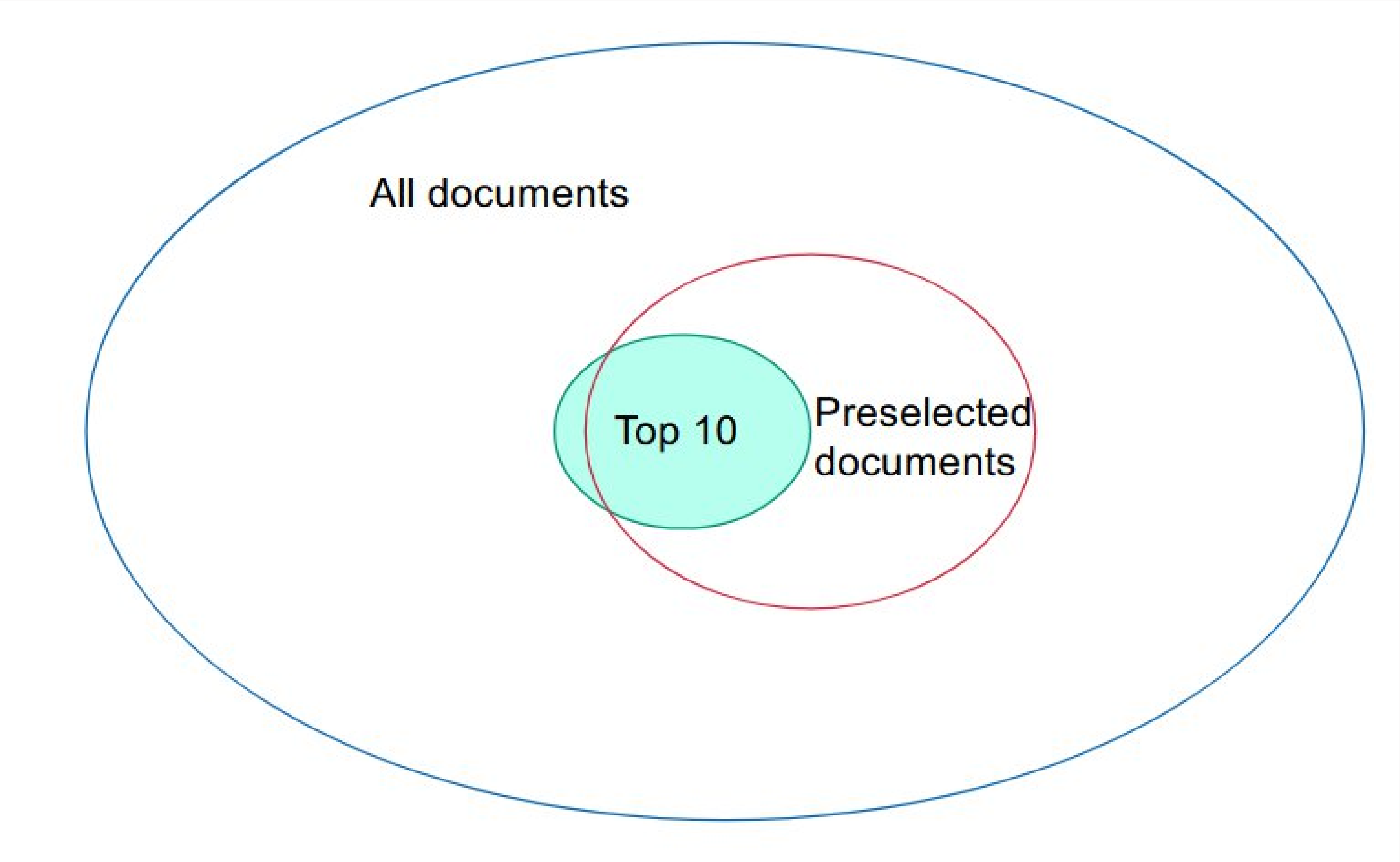
Computations

Postings lists can be very large:



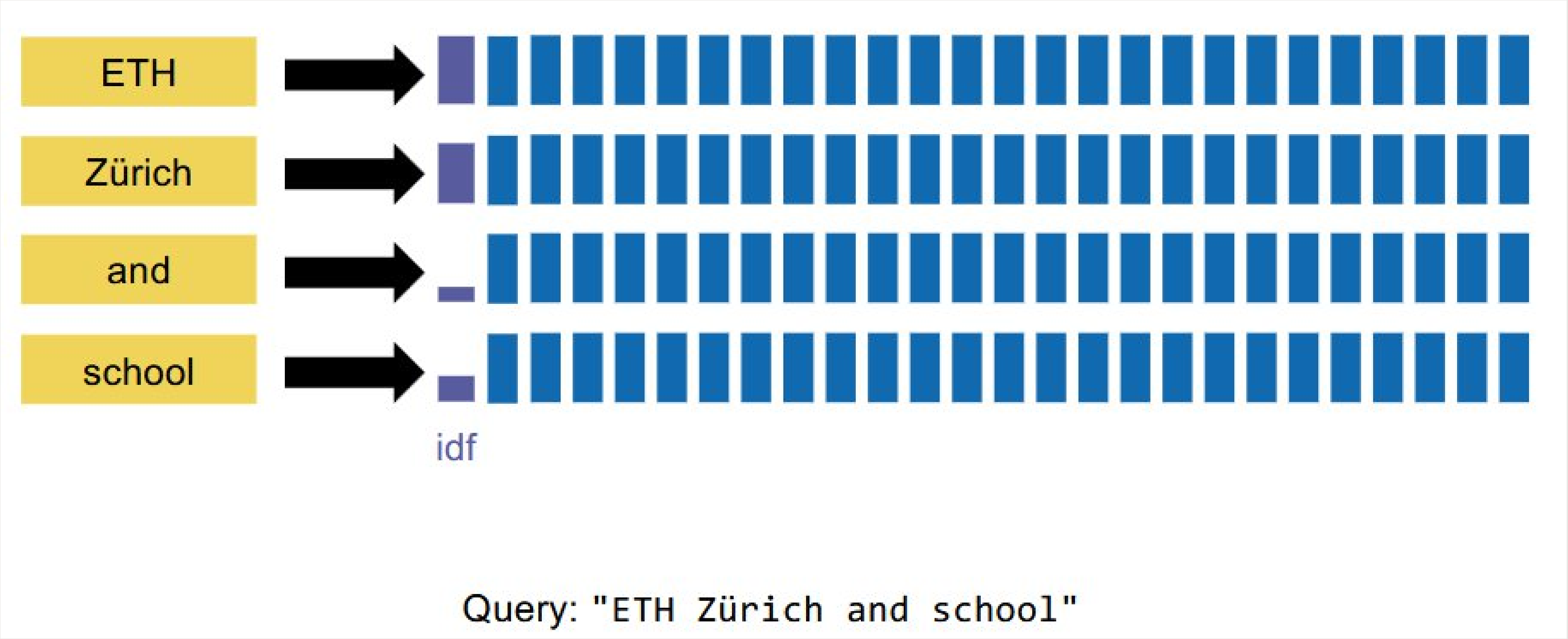
Idea

Preselect documents.
Only compute scores in smaller set.



Index Elimination

Remove terms with low idf.

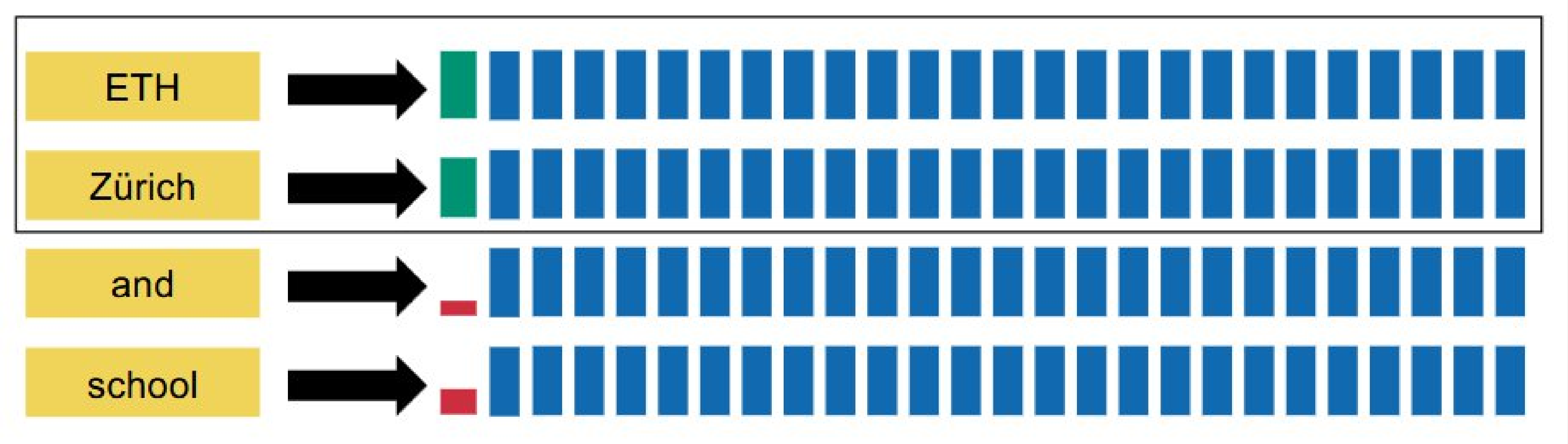


Query: "ETH Zürich and school"

Index Elimination

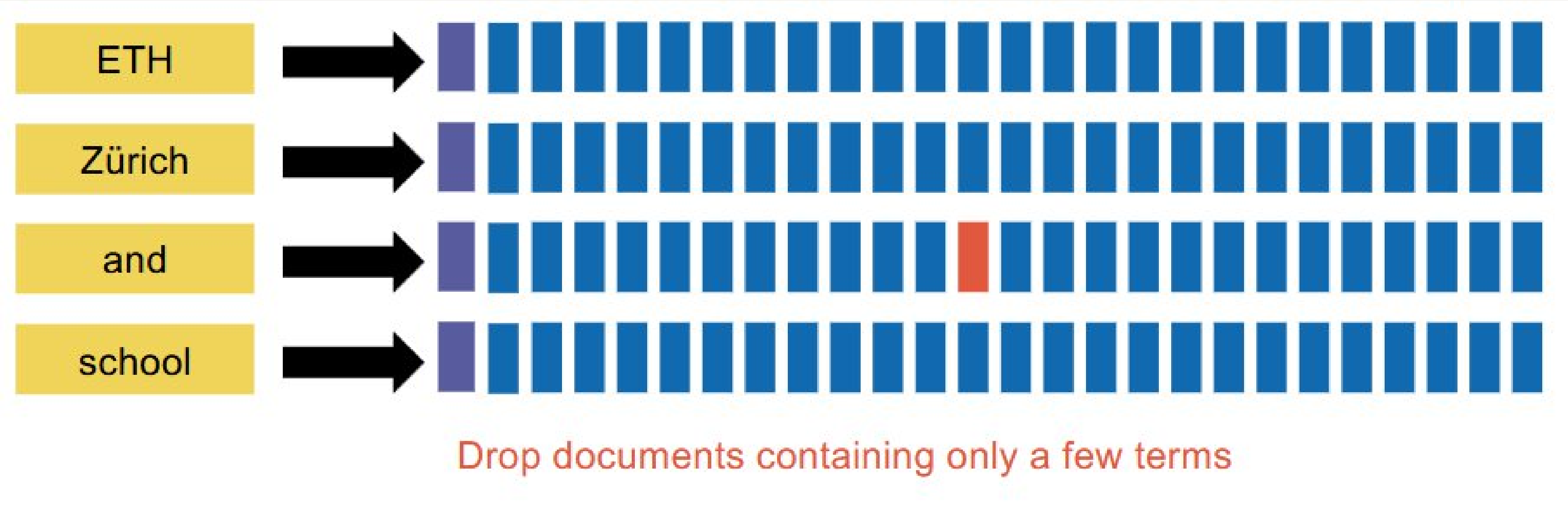
Remove terms with low idf.

Benefit: Usually low idf terms are contained in more documents.



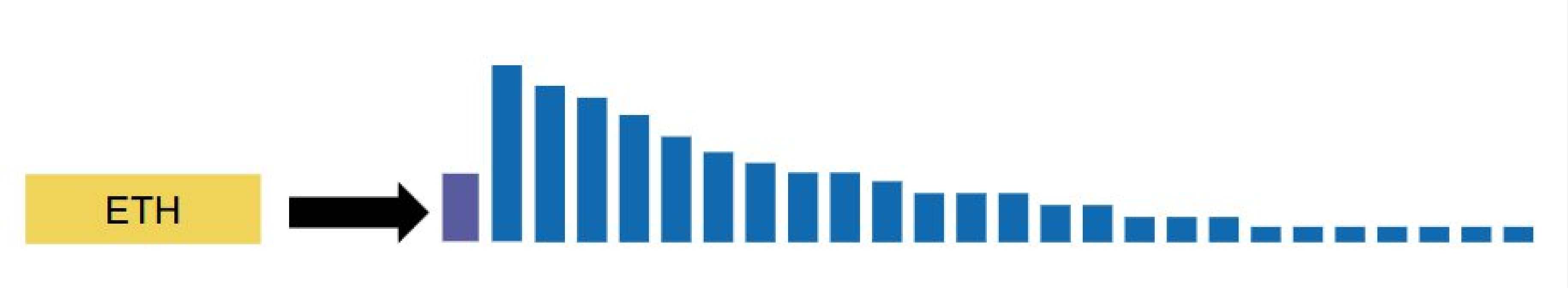
Index Elimination

Second idea: Keep only documents containing most terms.
We may drop too many documents this way though.



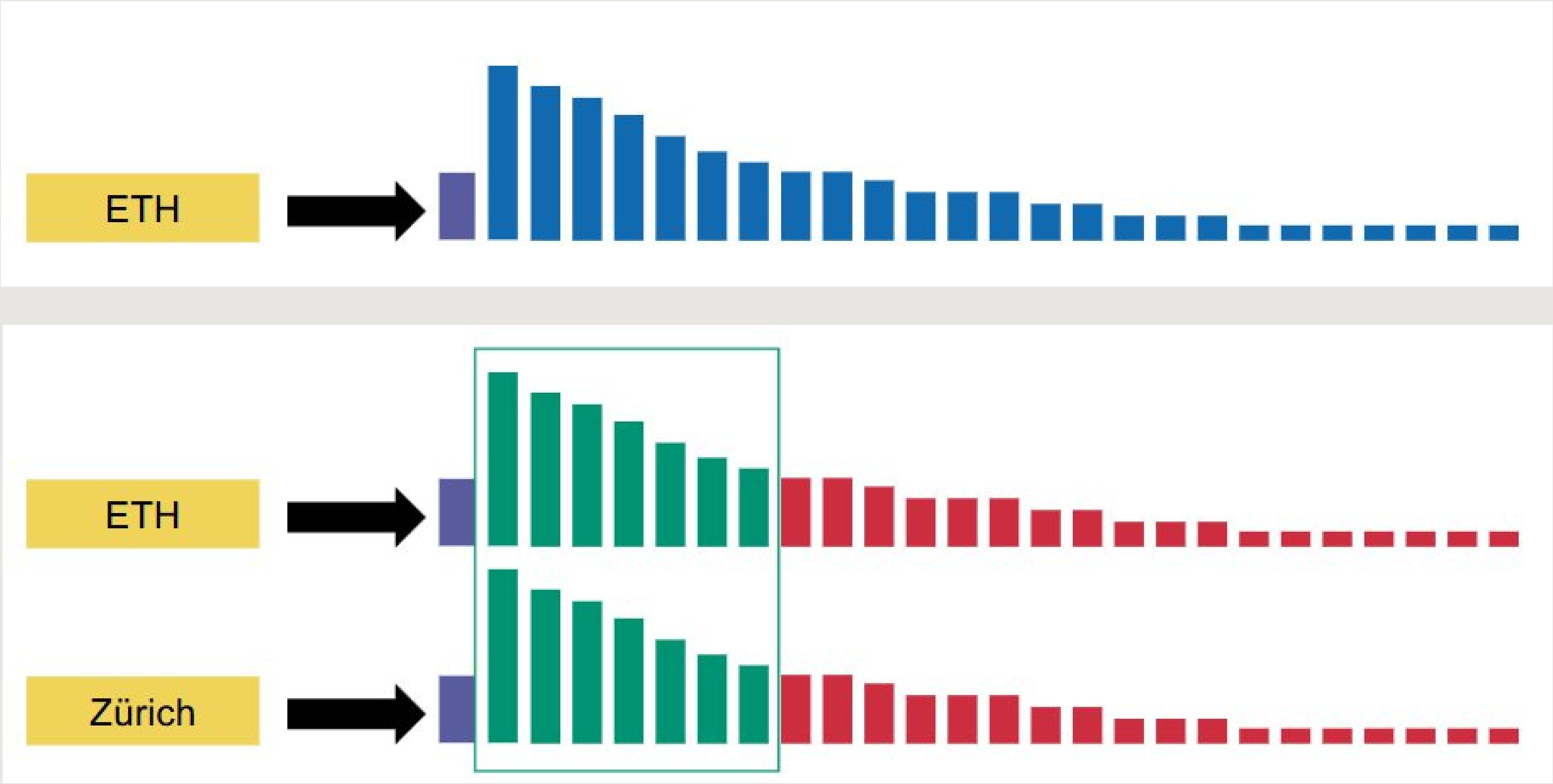
Champion Lists

1. Sort postings by decreasing term frequency



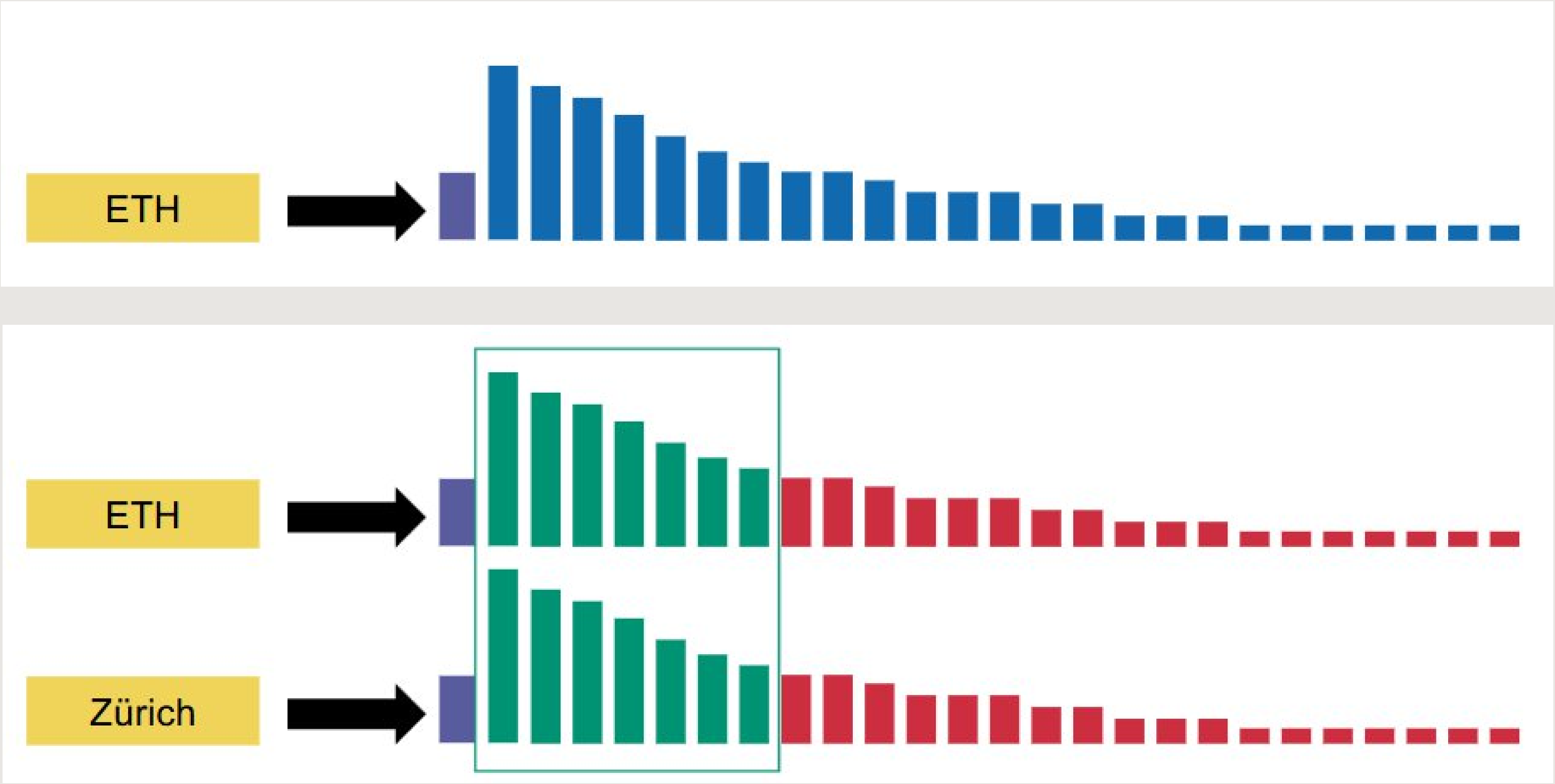
Champion Lists

1. Sort postings by decreasing term frequency
2. Only keep top K documents



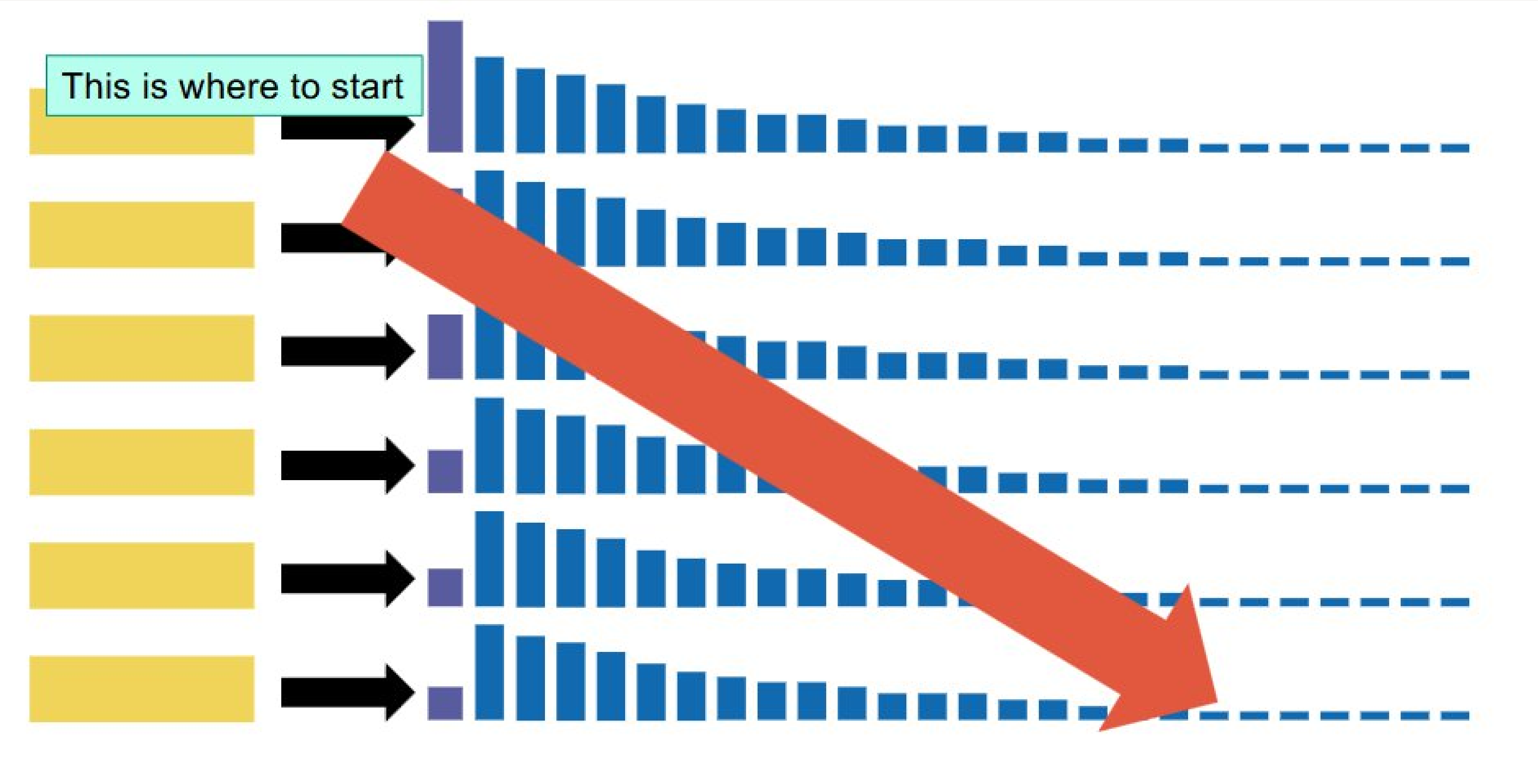
Champion Lists

1. Sort postings by decreasing term frequency
2. Only keep top K documents
3. Union those results



Impact ordering

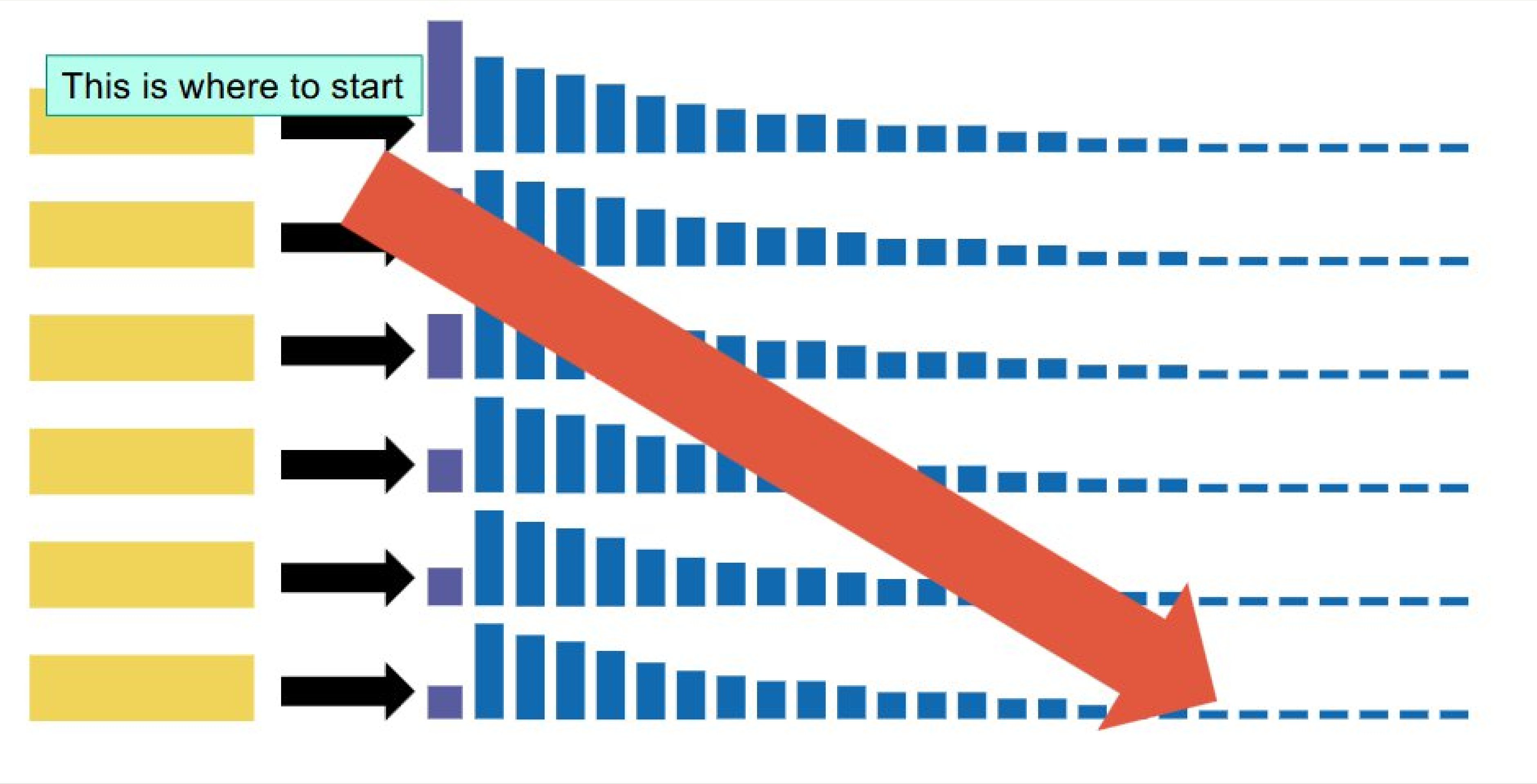
1. Create per-term Champion list
2. Sort terms by decreasing idf
3. Traverse term-at-a-time to collect top k documents



Impact ordering

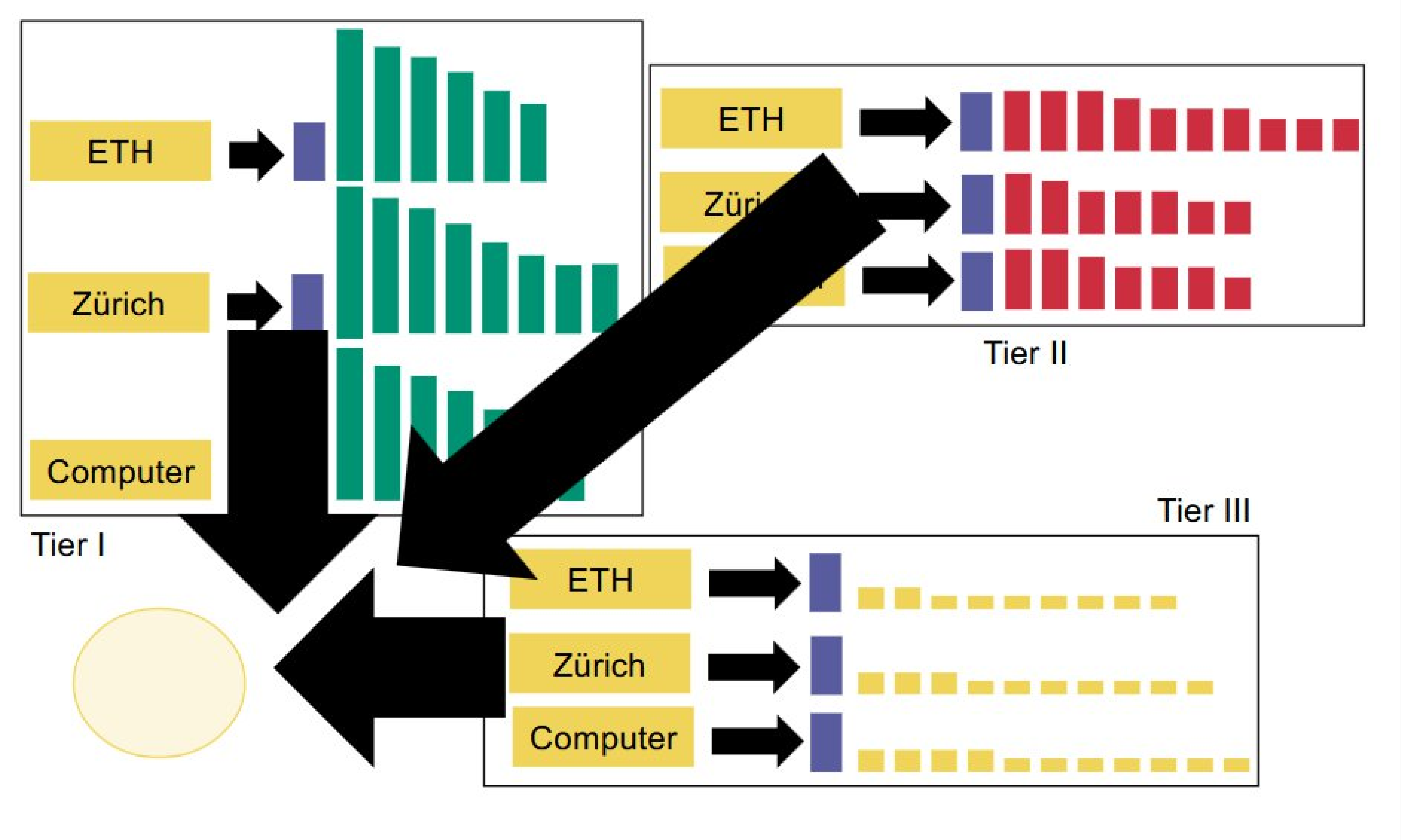
1. Create per-term Champion list
2. Sort terms by decreasing idf
3. Traverse term-at-a-time to collect top k documents

What if we don't have enough results?

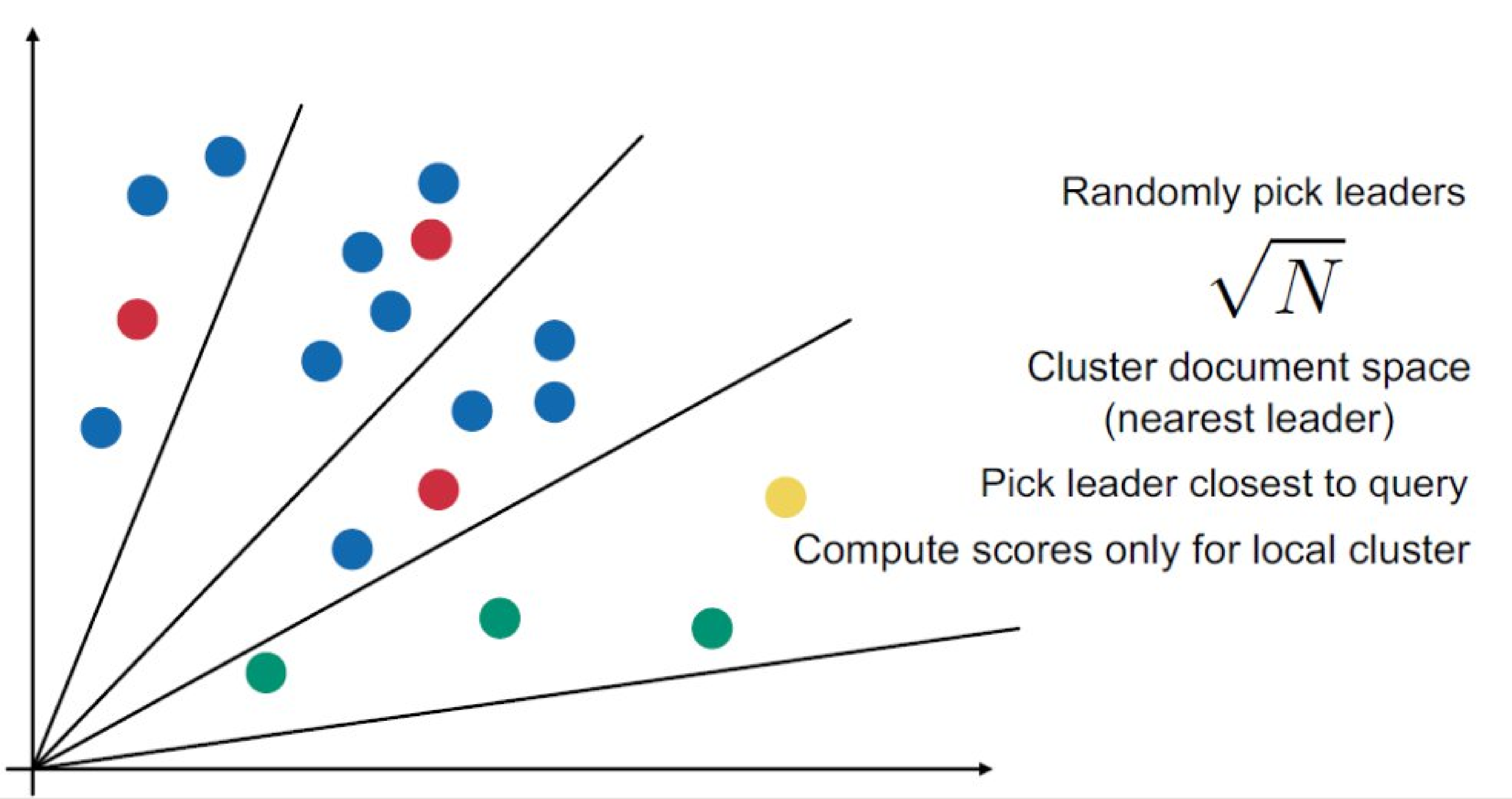


Tiered indices

1. Create impact ordering
2. Union results from Tier I
3. If not enough results, union results from Tier II
4. If still not enough results, union results from Tier III



Clustering



Randomly pick leaders

$\sqrt{N}$

Cluster document space
(nearest leader)

Pick leader closest to query

Compute scores only for local cluster

Kahoot

https://create.kahoot.it/details/duplicate-of-information-retrieval-ex-08-champion-lists-vector-space-models-mschoeb/d450d0d9-513f-4387-8aca-15e8eaae63d9