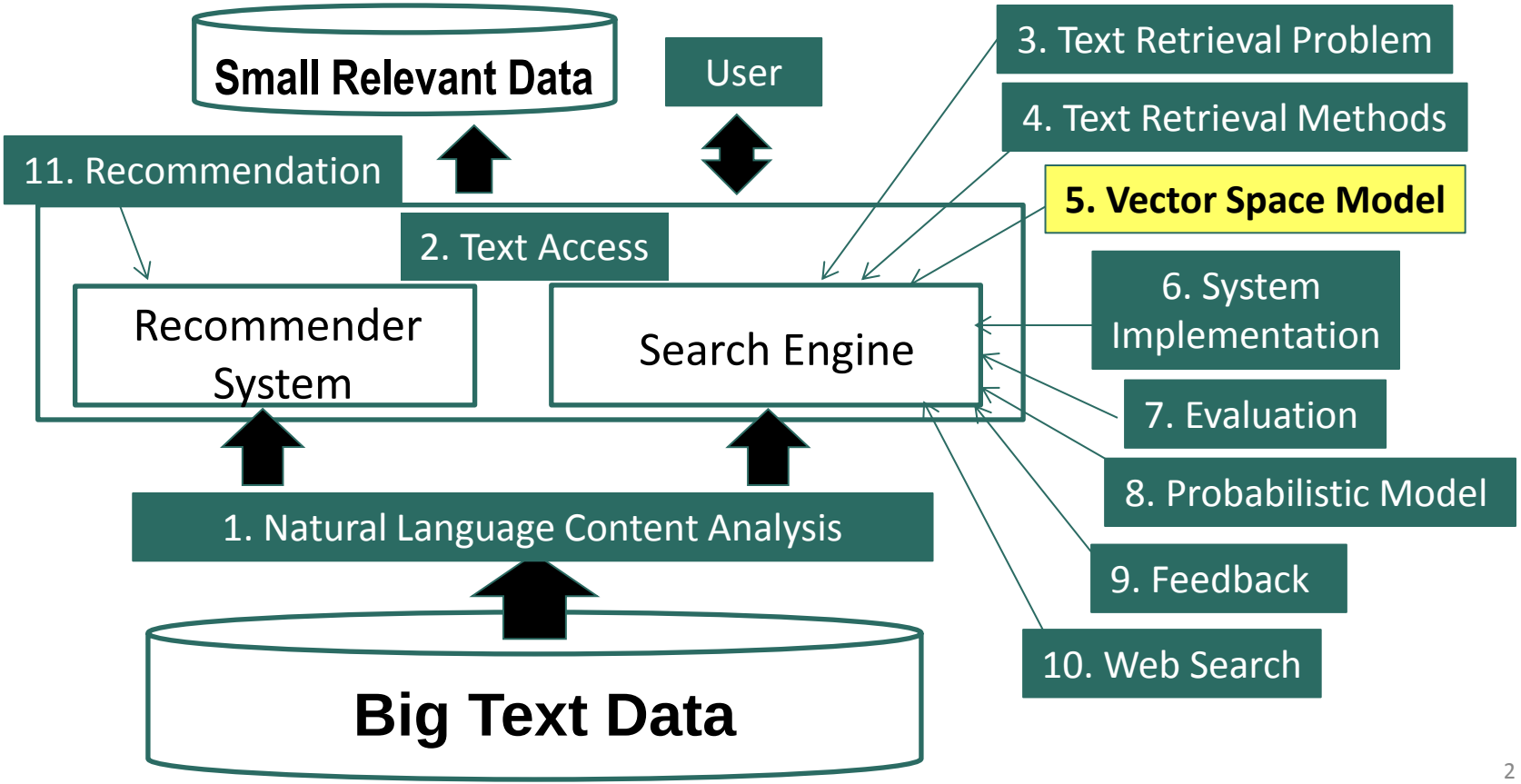
Text Retrieval and Search Engines

Vector Space Retrieval Model: TF Transformation

ChengXiang "Cheng" Zhai
Department of Computer Science
University of Illinois at Urbana-Champaign

Course Schedule



VSM with TF-IDF Weighting Still Has a Problem!

Query = "news about presidential campaign"

d1	… **news about** …	f(q,d1)=2.5
d2	… **news about** organic food **campaign**…	f(q,d2)=5.6
d3	… **news** of **presidential campaign** …	f(q,d3)=7.1
d4	… **news** of **presidential campaign** … … **presidential** candidate …	f(q,d4)=9.6
d5	… **news** of organic food **campaign**… **campaign**…**campaign**…**campaign**…	**f(q,d5)=13.9?**

Ranking Function with TF-IDF Weighting

Total # of docs in collection

$$f(q,d)=\sum_{i=1}^{N} x_i y_i = \sum_{w\in q\cap d} c(w,q)c(w,d)\log\frac{M+1}{df(w)}$$

All matched query words in d

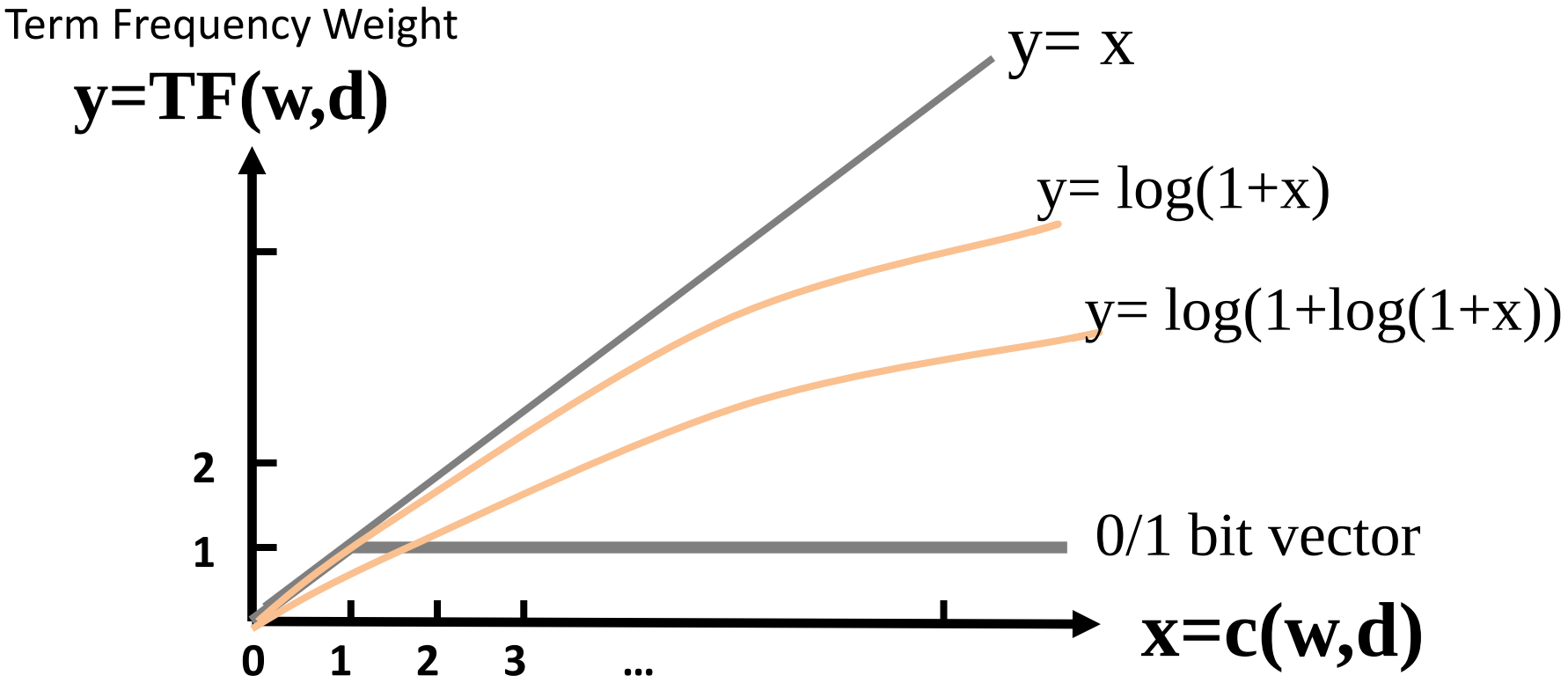
Doc Frequency

d5 … **news** of organic food **campaign**… **campaign**…**campaign**…**campaign**…

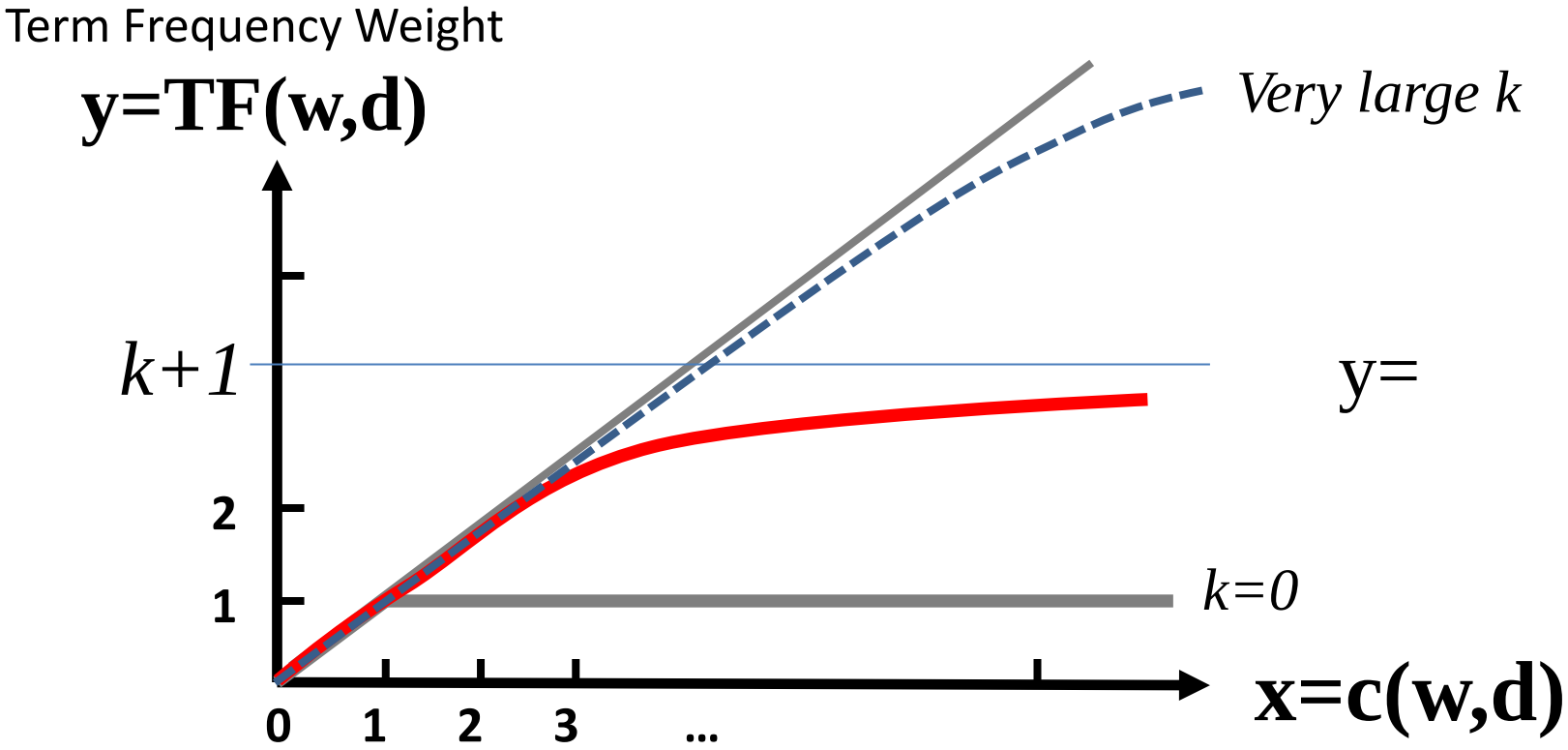
c(“campaign”,d5)=4

➔f(q,d5)=13.9?

TF Transformation: c(w,d)➔TF(w,d)



TF Transformation: BM25 Transformation



Summary

- Sublinear TF Transformation is needed to
  - capture the intuition of "diminishing return" from higher TF
  - avoid dominance by one single term over all others
- BM25 Transformation
  - has an upper bound
  - is robust and effective
- Ranking function with BM25 TF (k >=0)

$$f(q,d)=\sum_{i=1}^{N} x_i y_i = \sum_{w \in q \cap d} c(w,q)\frac{(k+1)c(w,d)}{c(w,d)+k}\log\frac{M+1}{df(w)}$$