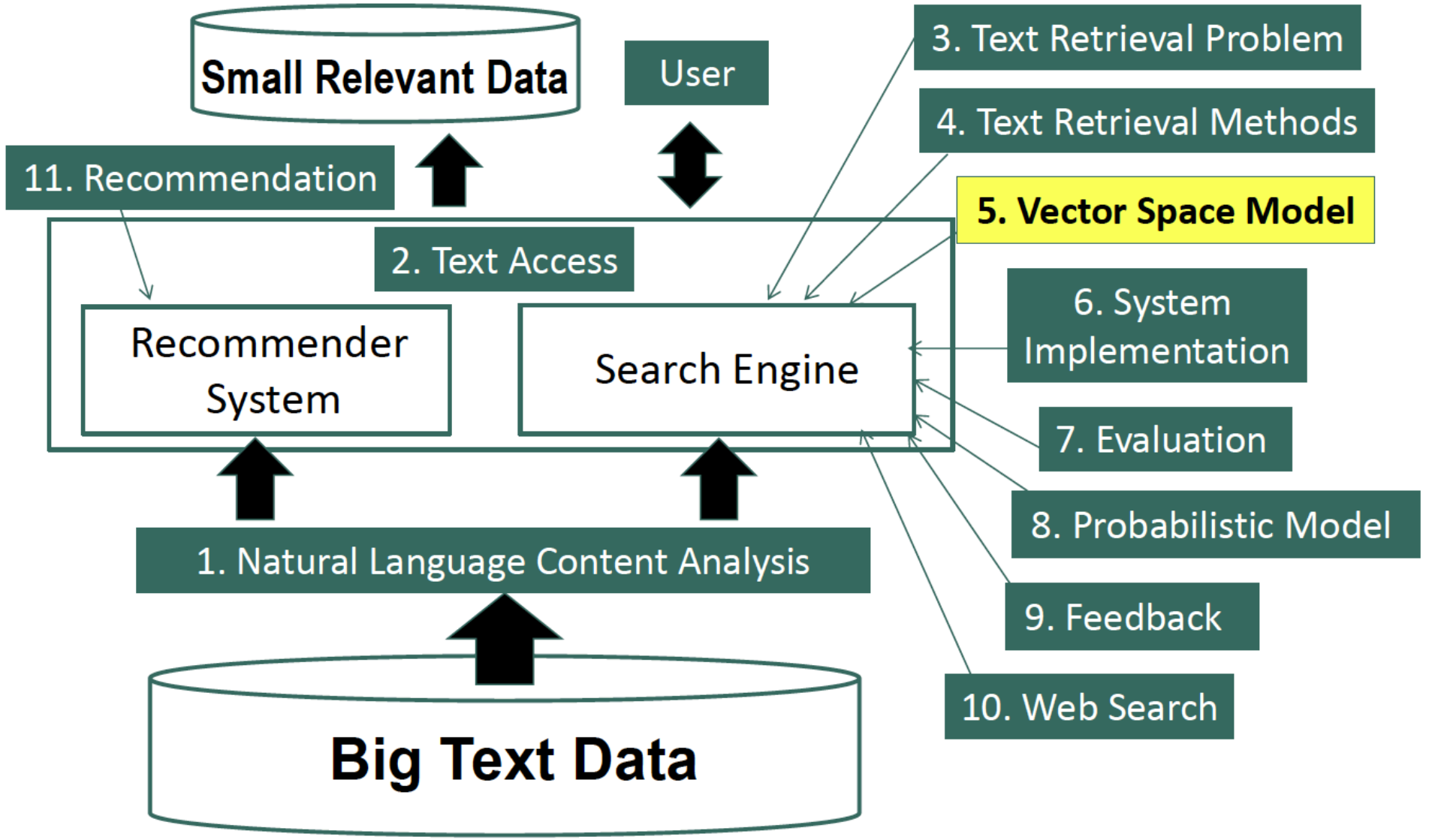
Information Retrieval & Text Mining

Vector Space Model
Doc Length Normalization

Dr. Saeed Ul Hassan
Information Technology University

Course Schedule



What about Document Length?

Query = "news about presidential campaign"

d4

… **news** of **presidential campaign** …
… **presidential** candidate …

100 words

d6 > d4?

d6

… **campaign**…..... **campaign**..... **5000 words** .....
.....
.....**news**.....
.....
..... **news**.....
.....
....
..... **presidential** ..... **presidential**.....

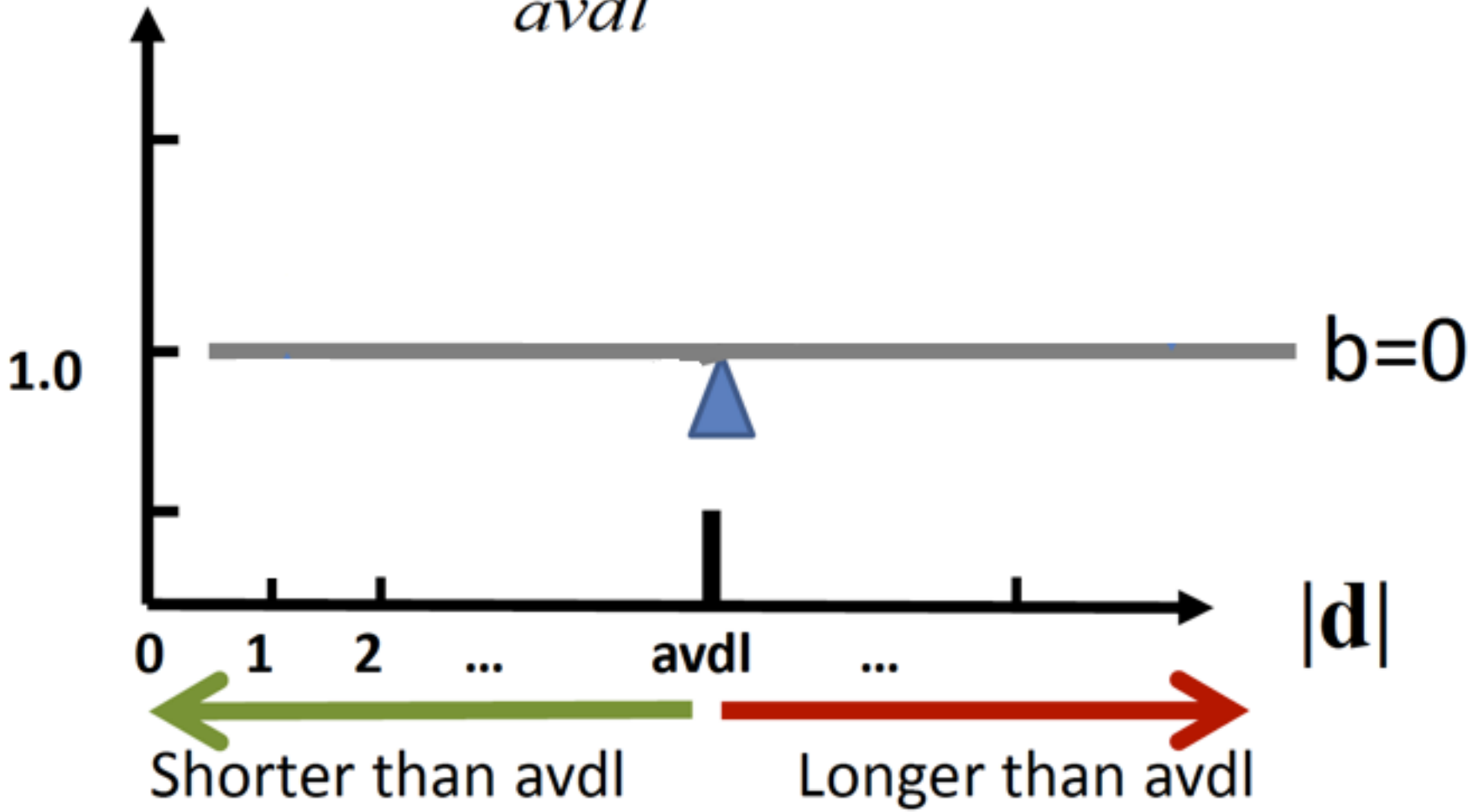
Document Length Normalization

- Penalize a long doc with a doc length normalizer
  - Long doc has a better chance to match any query
  - Need to avoid over-penalization
- A document is long because
  - it uses more words → more penalization
  - it has more contents → less penalization **Full paper vs Abstract**
- Pivoted length normalizer: average doc length as "pivot"
  - Normalizer = 1 if |d| =average doc length (avdl)

Pivoted Length Normalization

$$normalizer = 1 - b + b\frac{|d|}{avdl}$$

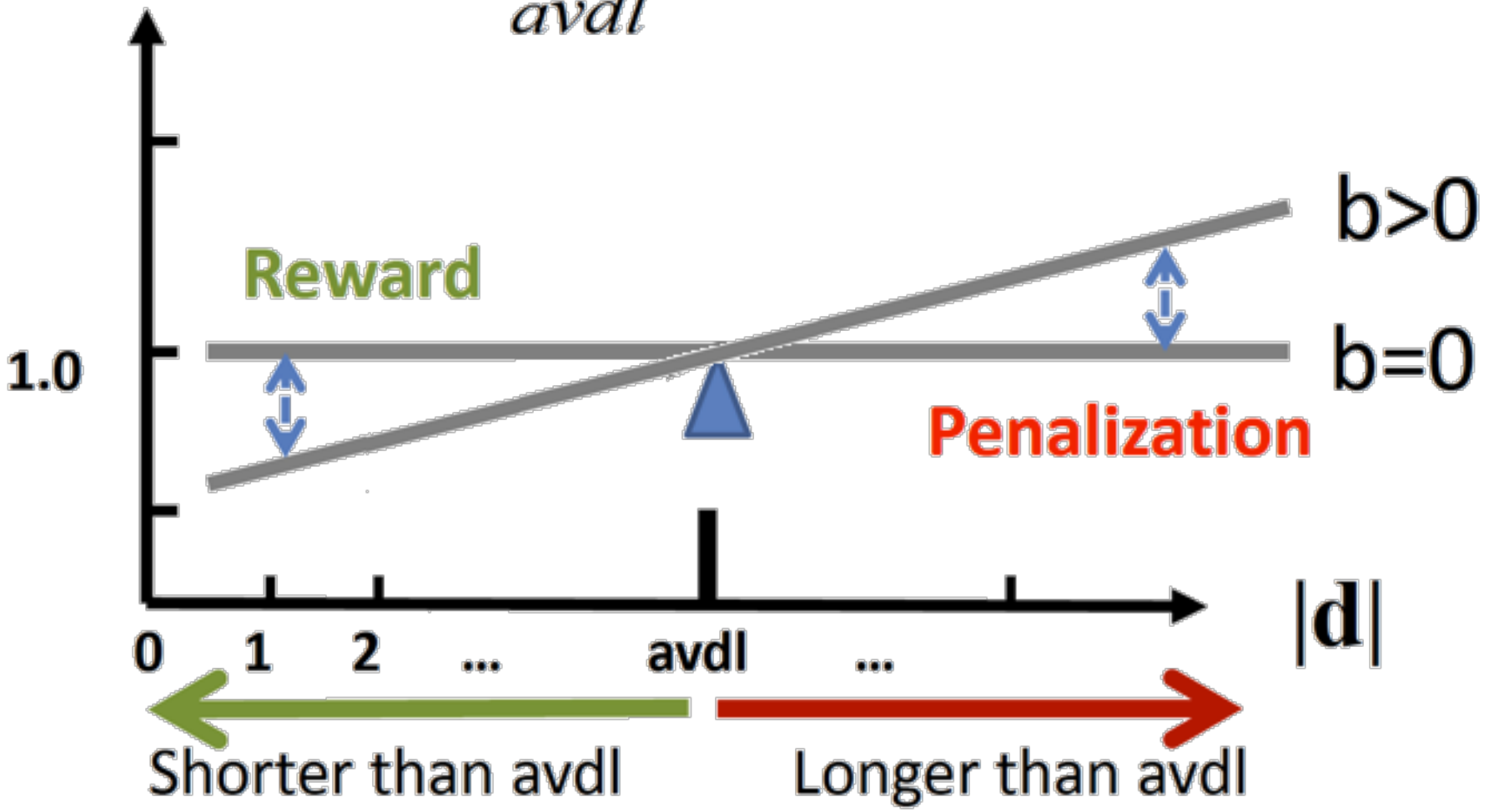
$$b \in [0,1]$$



Pivoted Length Normalization

$$normalizer = 1 - b + b\frac{|d|}{avdl}$$

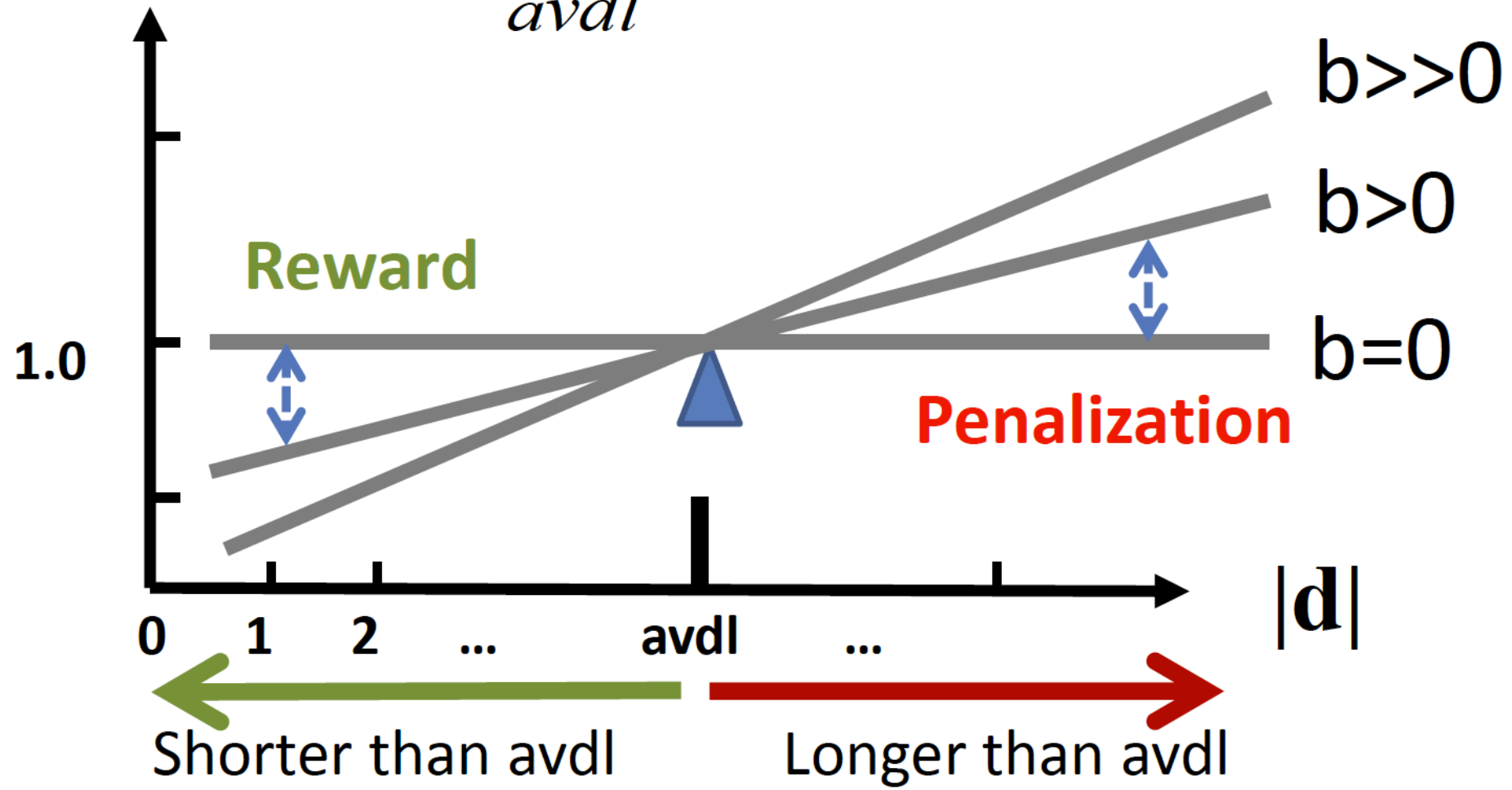
$$b \in [0,1]$$



Pivoted Length Normalization

$$normalizer = 1 - b + b\frac{|d|}{avdl}$$

$$b \in [0,1]$$



State of the Art VSM Ranking Functions

- Pivoted Length Normalization VSM [Singhal et al 96]

$$f(q,d)=\sum_{w\in q\cap d} c(w,q)\frac{\ln[1+\ln[1+c(w,d)]]}{1-b+b\frac{|d|}{avdl}}\log\frac{M+1}{df(w)}$$

- BM25/Okapi [Robertson & Walker 94]

$$b\in[0,1]$$
$$k\in[0,+\infty]$$

$$f(q,d)=\sum_{w\in q\cap d} c(w,q)\frac{(k+1)c(w,d)}{c(w,d)+k(1-b+b\frac{|d|}{avdl})}\log\frac{M+1}{df(w)}$$

Further Improvement of VSM?

- Improved instantiation of **dimension?**
  - stemmed words, stop word removal, phrases, latent semantic indexing (word clusters), character n-grams, …
  - bag-of-words with phrases is often sufficient in practice
  - Language-specific and domain-specific tokenization is important to ensure "normalization of terms" **phonetic representation of words**
- Improved instantiation of **similarity function?**
  - cosine of angle between two vectors?
  - Euclidean?
  - dot product seems still the best (sufficiently general especially with appropriate term weighting)

Further Improvement of BM25

- BM25F [Robertson & Zaragoza 09]
  - Use BM25 for documents with structures ("F"=fields)
  - Key idea: combine the frequency counts of terms in all fields and then apply BM25 (instead of the other way)
- BM25+ [Lv & Zhai 11]
  - Address the problem of over penalization of long documents by BM25 by adding a small constant to TF
  - Empirically and **analytically** shown to be better than BM25

Structured documents
e.g. title, author keywords etc.

Summary of Vector Space Model

- Relevance(q,d) = similarity(q,d)
- Query and documents are represented as vectors
- Heuristic design of ranking function
- Major term weighting heuristics
  - TF weighting and transformation
  - IDF weighting
  - Document length normalization
- BM25 and Pivoted normalization seem to be most effective

Additional Readings

- A. Singhal, C. Buckley, and M. Mitra. Pivoted document length normalization. In *Proceedings of ACM SIGIR 1996.*
- S. E. Robertson and S. Walker. Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval, *Proceedings of ACM SIGIR 1994*. **BM25**
- S. Robertson and H. Zaragoza. The Probabilistic Relevance Framework: BM25 and Beyond, *Found. Trends Inf. Retr.* 3, 4 (April 2009). **BM25F, HTML Tags?**
- Y. Lv, C. Zhai, Lower-bounding term frequency normalization. In *Proceedings of ACM CIKM 2011.* **BM25 Improvements**