TF-IDF

$$m_{d,t} = \left[1 + \log\left[tf_{d,t}\right]\right] \cdot \left[\log\left[\frac{N}{\mathrm{df}_t}\right]\right]$$

Cosine similarity

$$\frac{\sum_{j=1}^{|V|} m_{q,j} \cdot m_{i,j}}{\sqrt{\sum_{j=1}^{|V|} m_{i,j}^2}}$$

Okapi BM25

$$\mathcal{B}_{i,j} = \frac{(K_1 + 1) \cdot \mathrm{tf}_{i,j}}{K_1 \cdot \left[(1-b) + b \cdot \frac{\mathrm{len}(d_i)}{\mathrm{avg\_doclen}}\right] + \mathrm{tf}_{i,j}}$$

$$\mathrm{sim}_{\mathrm{BM25}}(d_i, q) \sim \sum_{w_j \in q \wedge w_j \in d_i} \mathcal{B}_{i,j} \cdot \log\left[\frac{N - n_j + 0.5}{n_j + 0.5}\right]$$

k=1, b = 0.75, n_j = docs with w_j

Jelinek-Mercer Smoothing

$$P(t|d) = \lambda \cdot P(t|M_d) + (1-\lambda) \cdot P(t|M_c)$$

Dirichlet Smoothing

$$P(t|d) = \frac{\mathrm{tf}_{d,t} + \mu \cdot P(t|M_c)}{|d| + \mu}$$

Model:	**What:**	**Advantages:**	**Disadvantages:**
Boolean	Boolean matching by term	- Easy/simple	- Term-Document Matrix is sparse - No ranking - Only exact matches
Vector Space	Unit vectors compared by cosine similarity	- Allows partial matching - Ranking by similarity (q and D) - Term weighting - Length normalization	- Terms assumed mutually independent - May get many non-relevant docs
Probabilistic	Estimates probability that doc is relevant to q. Improves with feedback	- Ranking on probability of being relevant - Partial matches	- Guess initial separation of relevant and non-relevant docs - Terms assumed independent - No TF or length normalization
Okapi BM25	Built on all principles. TF, length normalization, IDF	- No need for relevance info	- Bag-of-words - Not precise estimation
Language	Based on probability. Find P of D generating q	- Ranking based on probability - No assuming of term independence	- Phrase and Boolean-search is hard - URF is difficult - Similarity between q and D is unrealistic

Precision: relevant/retrieved
Recall: retrieved/relevant
F-measure: (2*P*R)/(P+R)
Accuracy: (TP + TN)/N

$$\mathrm{MAP}_i = \frac{1}{|R_i|} \cdot \sum_{k=1}^{|R_i|} P(R_i[k])$$

R-Precision: r/R

$$\mathrm{MRR} = \frac{1}{Q} \sum_{i=1}^{Q} \frac{1}{\mathrm{rank}_i}$$

Query Expansion
Explicit relevance feedback:
Rocchio: Get closer to neighborhood of relevant docs and away from non-relevant docs
Rocchio-formler: Standard Rocchio, Ide Regular, Ide Dec hi

Implicit relevance feedback:
Local analysis: extract information from the local set of documents retrieved to expand the query
Global analysis: expand the query using information from the whole set of documents. Uses either similarity or a statistical thesaurus