III.3 Probabilistic Retrieval Models

1. **Probabilistic Ranking Principle**
2. **Binary Independence Model**
3. **Okapi BM25**
4. **Tree Dependence Model**
5. **Bayesian Networks for IR**

Based on **MRS Chapter 11**

TF*IDF vs. Probabilistic IR vs. Statistical LMs

- **TF*IDF** and **VSM** produce sufficiently good results in practice but often criticized for being **“too ad-hoc”** or **“not principled”**
- Typically outperformed by **probabilistic retrieval models** and **statistical language models** in IR benchmarks (e.g., TREC)
- **Probabilistic retrieval models**
  - use **generative models** of documents as bags-of-words
  - explicitly model **probability of relevance** $P[R \mid d, q]$
- **Statistical language models**
  - use **generative models** of documents and queries as sequences-of-words
  - consider **likelihood** of generating query from document model or **divergence** of document model and query model (e.g., Kullback-Leibler)

Probabilistic Information Retrieval

- **Generative model**
  - **probabilistic mechanism** for producing documents (or queries)
  - usually based on a **family of parameterized probability distributions**

- **Powerful model** but restricted through practical limitations
  - often **strong independence assumptions** required for **tractability**
  - **parameter estimation** has to deal with **sparseness** of available data (e.g., collection with M terms has 2^M distinct possible documents, but model parameters need to be estimated from $N << 2^M$ documents)

Multivariate Bernoulli Model

- For generating document **d** from joint (multivariate) term distribution Φ
  - consider **binary random variables**: $d_t = 1$ if term in **d**, 0 otherwise
  - postulate **independence** among these random variables

$$P[d|\Phi] = \prod_{t \in V} \phi_t^{d_t} (1 - \phi_t^{1-d_t})$$

$$\phi_t = P[\text{term } t \text{ occurs in a document}]$$

- Problems:
  - underestimates probability of short documents
  - product for absent terms underestimates probability of likely documents
  - too much probability mass given to very unlikely term combinations

1. Probability Ranking Principle (PRP)

*"If a reference retrieval system's response to each request is a ranking of the documents in the collection in order of decreasing **probability of relevance** to the user who submitted the request, where the probabilities are **estimated as accurately as possible** on the basis of whatever data have been made available to the system for this purpose, **the overall effectiveness of the system** to its user **will be the best** that is obtainable on the basis of those data."*

[van Rijsbergen 1979]

- **PRP with costs** [Robertson 1977] defines cost of retrieving d as the next result in a ranked list for query q as

$$cost(d, q) = C_1 \, P[R|d, q] + C_0 \, P[\bar{R}|d, q]$$

  with **cost constants**

  - C_1 as cost of retrieving a **relevant document**
  - C_2 as cost of retrieving an **irrelevant document**

- For $C_1 < C_0$, cost is minimized by choosing $\underset{d}{arg\,max} \; P[R|d, q]$

Derivation of Probability Ranking Principle

- Consider document *d* to be retrieved next, because it is preferred (i.e, has lower cost) over all other candidate documents *d'*

$$
\begin{aligned}
& cost(d,q) && \leq \quad cost(d',q) \\
\Leftrightarrow \quad & C_1\,P[R|d,q] + C_0\,P[\bar{R}|d,q] && \leq \quad C_1\,P[R|d',q] + C_0\,P[\bar{R}|d',q] \\
\Leftrightarrow \quad & C_1\,P[R|d,q] + C_0\,(1 - P[R|d,q]) && \leq \quad C_1\,P[R|d',q] + C_0\,(1 - P[R|d',q]) \\
\Leftrightarrow \quad & C_1\,P[R|d,q] - C_0\,P[R|d,q] && \leq \quad C_1\,P[R|d',q] - C_0\,P[R|d',q] \\
\Leftrightarrow \quad & (C_1 - C_0)\,P[R|d,q] && \leq \quad (C_1 - C_0)\,P[R|d',q] \\
\Leftrightarrow \quad & P[R|d,q] && \geq \quad P[R|d',q] \quad \textbf{(assuming } C_1 < C_0\textbf{)}
\end{aligned}
$$

Probability Ranking Principle (cont'd)

- Probability ranking principle makes **two strong assumptions**
  - $P[R \mid d, q]$ can be **determined accurately**
  - $P[R \mid d, q]$ and $P[R \mid d', q]$ are **pairwise independent** for documents d, d'
- **PRP without costs** (based on Bayes' optimal decision rule)
  - returns **set of documents** d for which $P[R \mid d, q] > (1 - P[R \mid d, q])$
  - minimizes the **expected loss** (aka. Bayes' risk) under the 1/0 loss function

2. Binary Independence Model (BIM)

- **Binary independence model** [Robertson and Spärck-Jones 1976] has traditionally been used with the probabilistic ranking principle
- Assumptions:
  - relevant and irrelevant documents **differ in their term distribution**
  - probabilities of term occurrences are **pairwise independent**
  - documents are **sets of terms**, i.e., **binary term weights** in {0,1}
  - **non-query terms** have the same probability of occurring in relevant and non-relevant documents
  - **relevance** of a document is **independent** of relevance **others document**

Ranking Proportional to Relevance Odds

$$
\begin{aligned}
O(R|d) &= \frac{P[R|d]}{P[\bar{R}|d]} && \textbf{(odds for ranking)} \\
&= \frac{P[d|R] \times P[R]}{P[d|\bar{R}] \times P[\bar{R}]} && \textbf{(Bayes' theorem)} \\
&\propto \frac{P[d|R]}{P[d|\bar{R}]} && \textbf{(rank equivalence)} \\
&= \prod_{t \in V} \frac{P[d_t|R]}{P[d_t|\bar{R}]} && \textbf{(independence assumption)} \\
&= \prod_{t \in q} \frac{P[d_t|R]}{P[d_t|\bar{R}]} && \textbf{(non-query terms)} \\
&= \prod_{\substack{t \in d \\ t \in q}} \frac{P[D_t|R]}{P[D_t|\bar{R}]} \times \prod_{\substack{t \notin d \\ t \in q}} \frac{P[\bar{D}_t|R]}{P[\bar{D}_t|\bar{R}]}
\end{aligned}
$$

with d_t indicating if **document *d*** includes **term *t***
and D_t indicating if **random document** includes **term *t***

Ranking Proportional to Relevance Odds (cont'd)

$$= \prod_{\substack{t \in d \\ t \in q}} \frac{P[D_t|R]}{P[D_t|\bar{R}]} \times \prod_{\substack{t \notin d \\ t \in q}} \frac{P[\bar{D}_t|R]}{P[\bar{D}_t|\bar{R}]}$$

$$= \prod_{\substack{t \in d \\ t \in q}} \frac{p_t}{q_t} \times \prod_{\substack{t \notin d \\ t \in q}} \frac{(1-p_t)}{1-q_t} \qquad (\textbf{shortcuts } p_t \textbf{ and } q_t)$$

$$= \prod_{t \in q} \frac{p_t^{d_t}}{q_t^{d_t}} \times \prod_{t \in q} \frac{(1-p_t)^{1-d_t}}{(1-q_t)^{1-d_t}}$$

$$\propto \sum_{t \in q} log\left(\frac{p_t^{d_t}\,(1-p_t)}{(1-p_t)^{d_t}}\right) - log\left(\frac{q_t^{d_t}\,(1-q_t)}{(1-q_t)^{d_t}}\right)$$

$$= \sum_{t \in q} d_t\, log\, \frac{p_t}{1-p_t} + \sum_{t \in q} d_t\, log\, \frac{1-q_t}{q_t} + \sum_{t \in q} log\, \frac{1-p_t}{1-q_t}$$

$$\propto \sum_{t \in q} d_t\, log\, \frac{p_t}{1-p_t} + \sum_{t \in q} d_t\, log\, \frac{1-q_t}{q_t} \qquad (\textbf{invariant of } d)$$

Estimating p_t and q_t with a Training Sample

- We can estimate p_t and q_t based on a **training sample** obtained by evaluating the query q on a **small sample of the corpus** and asking the user for **relevance feedback** about the results

- Let ***N*** be the # documents in our sample
 R be the # relevant documents in our sample
 $\boldsymbol{n_t}$ be the # documents in our sample that contain t
 $\boldsymbol{r_t}$ be the # relevant documents in our sample that contain t
 we estimate

$$p_t = \frac{r_t}{R} \qquad q_t = \frac{n_t - r_t}{N - R}$$

 or with **Lidstone smoothing** ($\lambda = 0.5$)

$$p_t = \frac{r_t + 0.5}{R + 1} \qquad q_t = \frac{n_t - r_t + 0.5}{N - R + 1}$$

Smoothing (with Uniform Prior)

- Probabilities p_t and q_t for term t are estimated by **MLE for Binomial distribution**
  - repeated coin tosses for term t in relevant documents (p_t)
  - repeated coin tosses for term t in irrelevant documents (q_t)
- Avoid **overfitting** to the training sample by **smoothing estimates**
  - **Laplace smoothing** (based on Laplace's law of succession)

$$p_t = \frac{r_t + 1}{R + 2} \qquad q_t = \frac{n_t - r_t + 1}{N - R + 2}$$

  - **Lidstone smoothing** (heuristic generalization with $\lambda > 0$)

$$p_t = \frac{r_t + \lambda}{R + 2\,\lambda} \qquad q_t = \frac{n_t - r_t + \lambda}{N - R + 2\,\lambda}$$

Binary Independence Model (Example)

- Consider query $q = \{t_1, \ldots, t_6\}$ and **sample of four documents**

	t_1	t_2	t_3	t_4	t_5	t_6	R
d_1	1	0	1	1	0	0	**1**
d_2	1	1	0	1	1	0	**1**
d_3	0	0	0	1	1	0	**0**
d_4	0	0	1	0	0	0	**0**
n_t	2	1	2	3	2	0	
r_t	2	1	1	2	1	0	
p_t	5/6	1/2	1/2	5/6	1/2	1/6	
q_t	1/6	1/6	1/2	1/2	1/2	1/6	

$R = 2$
$N = 4$

- For document $d_6 = \{t_1, t_2, t_6\}$ we obtain

$$P[R|d_6, q] \propto log\,5 \;+\; log\,1 \;+\; log\,\frac{1}{5} \;+\; log\,5 \;+\; log\,5 \;+\; log\,5$$

Estimating p_t and q_t without a Training Sample

- When **no training sample** is available, we estimate p_t and q_t as

$$p_t = (1 - p_t) = \frac{1}{2} \qquad q_t = \frac{df_t}{|D|}$$

  - p_t reflects that we have **no information about relevant documents**
  - q_t under the assumption that **# relevant documents <<< # documents**

- When we plug in these estimates of p_t and q_t, we obtain

$$P[R|d,q] = \sum_{t \in q} d_t \, log \, 1 \; + \sum_{t \in q} d_t \, log \, \frac{|D| - df_t}{df_t} \approx \sum_{t \in q} d_t \, log \, \frac{|D|}{df_t}$$

which **can be seen as TF*IDF** with binary term frequencies and logarithmically dampened inverse document frequencies

Poisson Model

- For generating document **d** from joint (multivariate) term distribution Φ
  - consider **counting random variables**: $d_t = tf_{t,d}$
  - postulate **independence** among these random variables
- **Poisson model** with term-specific parameters μ_t:

$$P[d|\mu] = \prod_{t \in V} \frac{e^{-\mu_t} \cdot \mu_t^{d_t}}{d_t!} = e^{-\sum_{t \in V} \mu_t} \prod_{t \in d} \frac{\mu_t^{d_t}}{d_t!}$$

- MLE for μ_t from n sample documents $\{d_1, \ldots, d_n\}$: $\hat{\mu}_t = \frac{1}{n} \sum_{i=1}^{n} tf_{t,d_i}$
  - no penalty for absent words
  - no control of document length

3. Okapi BM25

- Generalizes term weight

$$w = log \frac{p(1-q)}{q(1-p)}$$

into

$$w = log \frac{p_{tf} q_0}{q_{tf} p_0}$$

where p_i and q_i denote the probability that **term occurs *i* times** in a relevant or irrelevant document, respectively

- Postulates Poisson (or 2-Poisson-mixture) distributions for terms

$$p_{tf} = e^{-\lambda} \frac{\lambda^{tf}}{tf!} \qquad q_{tf} = e^{-\mu} \frac{\mu^{tf}}{tf!}$$

Okapi BM25 (cont'd)

- Reduces the number of parameters that have to be learned and **approximates Poisson model** by similarly-shaped function

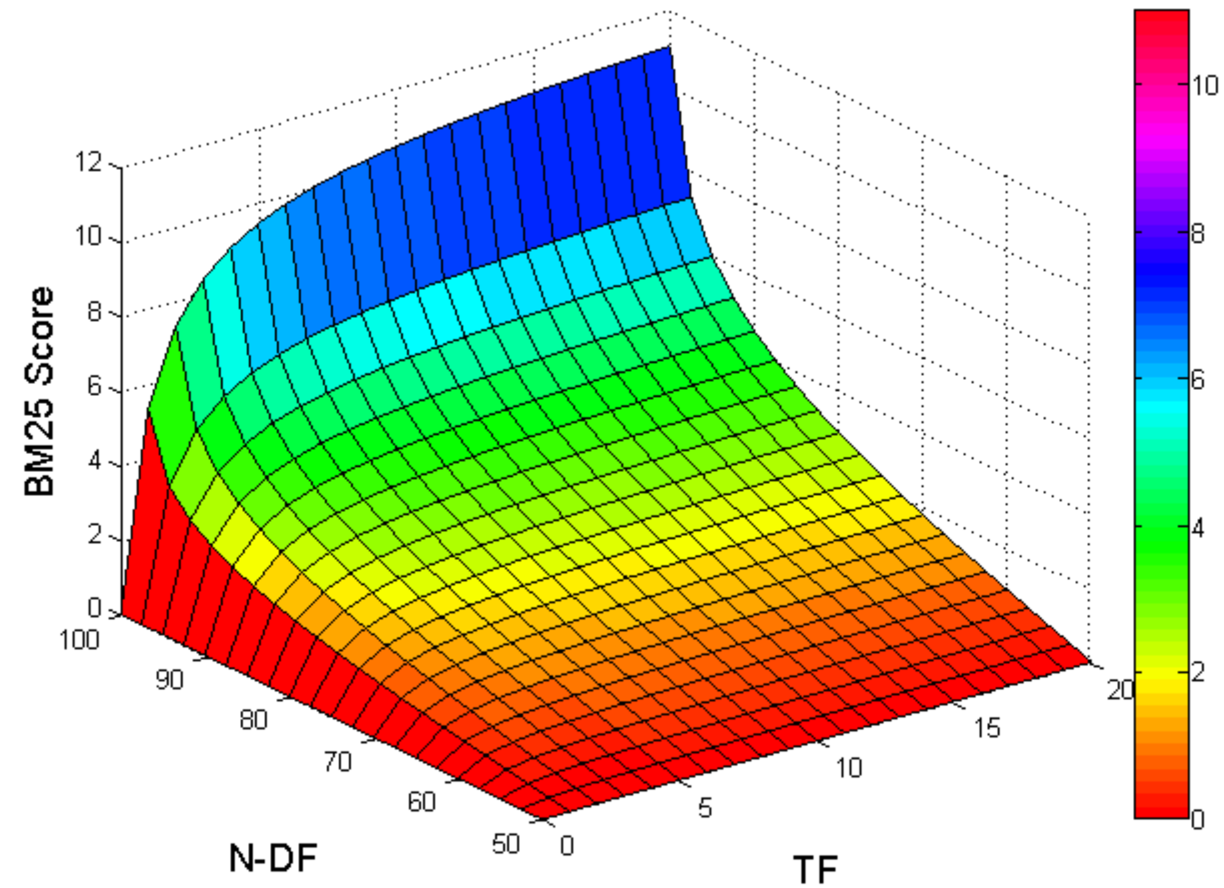
$$w = \frac{tf}{k_1 + tf} \; log \, \frac{p(1-q)}{q(1-p)}$$

- Finally leads to Okapi BM25 as **state-of-the-art retrieval model** (with top-ranked results in TREC)

$$w_{t,d} = \frac{(k_1 + tf_{t,d})}{k_1((1-b) + b \, \frac{|d|}{avdl}) + tf_{t,d}} \; log \, \frac{|D| - df_j + 0.5}{df_j + 0.5}$$

  - k_1 controls **impact of term frequency** (common choice $k_1 = 1.2$)
  - b controls **impact of document length** (common choice $b = 0.75$)

Okapi BM25 (Example)



- 3D plot of a simplified BM25 scoring function using $k_1 = 1.2$ as parameter (DF mirrored for better readability)
- Scores for $df_t > N/2$ are negative

$$w_t = \frac{(k_1 + 1)\, tf_{t,d}}{k_1 + tf_{t,d}} \; log \, \frac{|D| - df_t + 0.5}{df_t + 0.5}$$

4. Tree Dependence Model

- Consider term correlations in documents (with binary RV X_i) requires estimating *m-dimensional* probability distribution

$$P[X_1 = .., \ldots, X_m = ..] = f_X(X_1, \ldots, X_m)$$

- **Tree dependence model** [van Rijsbergen 1979]
  - considers **only 2-dimensional probabilities** for term pairs (i, j)

$$\begin{aligned} f_{ij}(X_i, X_j) &= P[X_i = .., X_j = ..] \\ &= \sum_{X_1} \cdots \sum_{X_{i-1}} \sum_{X_{i+1}} \cdots \sum_{X_{j-1}} \sum_{X_{j+1}} \cdots \sum_{X_m} P[X_1 = .., \ldots, X_m = ..] \end{aligned}$$

  - estimates for each (i, j) the **error made by independence assumptions**
  - constructs a tree with **terms as nodes** and ***m*-1 weighted edges** connecting the **highest-error term pairs**

Two-Dimensional Term Correlations

- **Kullback-Leibler divergence** estimates error of approximating *f* by *g* assuming pairwise term independence

$$\epsilon(f,g) = \sum_{\mathbf{X} \in \{0,1\}^m} f(\mathbf{X})\ log \frac{f(\mathbf{X})}{g(\mathbf{X})} = \sum_{\mathbf{X} \in \{0,1\}^m} f(\mathbf{X})\ log \frac{f(\mathbf{X})}{\prod_{i=1}^{m} g(X_i)}$$

- **Correlation coefficient** for term pairs

$$\rho(X_i, X_j) = \frac{Cov(X_i, X_j)}{\sqrt{Var(X_i)}\sqrt{Var(X_j)}}$$

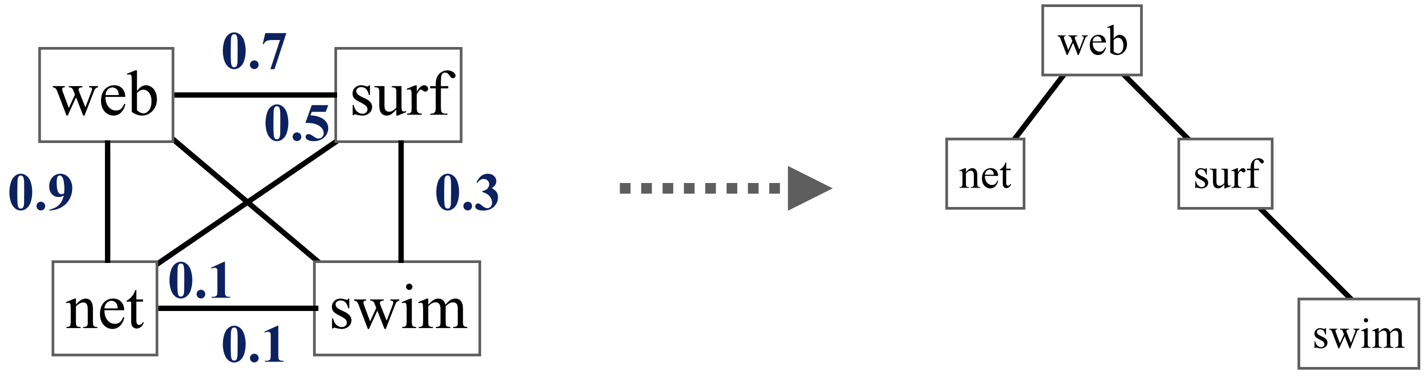
- *p*-values of X^2 **test of independence**

Kullback-Leibler Divergence (Example)

- Given are documents d_1=(1,1), d_2=(0,0), d_3=(1,1), d_4=(0,1)
- **2-dimensional probability distribution** f:
  $f(1,1) = P[X_1 = 1, X_2 = 1] = 2/4$
  $f(0,0) = 1/4$, $f(0,1) = 1/4$, $f(1,0) = 0$
- **1-dimensional marginal distributions** g_1 and g_2
  $g_1(1) = P[X_1{=}1] = 2/4$, $g_1(0) = 2/4$
  $g_2(1) = P[X_2{=}1] = 3/4$, $g_2(0) = 1/4$
- **2-dimensional probability distribution** assuming independence
  $g(1,1) = g_1(1)\, g_2(1) = 3/8$
  $g(0,0) = 1/8$, $g(0,1) = 3/8$, $g(1,0) = 1/8$
- **approximation error** ε (Kullback-Leibler divergence)
  $\varepsilon = 2/4 \log 4/3 + 1/4 \log 2 + 1/4 \log 2/3 + 0$

Constructing the Term Dependence Tree

- Input: Complete graph (V, E) with m nodes $X_i \in V$ and m^2 undirected edges $(i, j) \in E$ with weights ε
- Output: Spanning tree (V, E') with maximum total edge weight
- Algorithm:
  - **Sort** m^2 edges in descending order of weights
  - $E' = \varnothing$
  - **Repeat until** $|E'| = m-1$
    - $E' = E' \cup$
      $\{(i, j) \in E \setminus E' \mid (i, j)$ has **maximal weight** and E' **remains acyclic**$\}$
- Example:

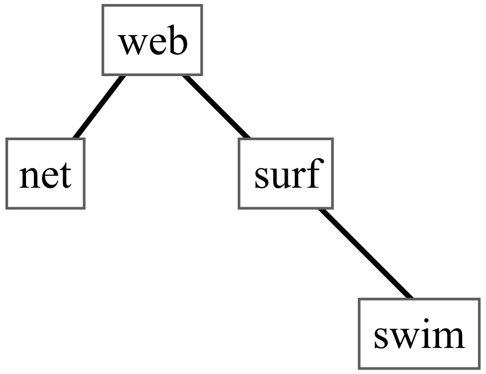


Estimation with Term Dependence Tree

- Given a term dependence tree ($V=\{X_1, \ldots, X_m\}$, E') with preorder-labeled nodes (i.e., X_1 is root) and assuming that X_i and X_j are independent for $(i, j) \notin E'$

$$\begin{aligned} & P[X_1 = .., \ldots, X_m = ..] \\ = \; & P[X_1 = ..]\, P[X_2 = .., \ldots, X_m = .. | X_1 = ..] && (\textbf{conditional probability}) \\ = \; & \prod_{i=1}^{m} P[X_i = .. | X_1 = .., \ldots, X_{i-1} = ..] && (\textbf{chain rule}) \\ = \; & P[X_1] \prod_{(i,j) \in E'} P[X_j | X_i] && (\textbf{independence assumption}) \\ = \; & P[X_1] \prod_{(i,j) \in E'} \frac{P[X_j, X_i]}{P[X_i]} && (\textbf{conditional probability}) \end{aligned}$$

- <u>Example:</u>



$$\begin{aligned} & P[\text{web}, \text{net}, \text{surf}, \text{swim}] \\ = \; & P[\text{web}]\, P[\text{net}|\text{web}]\, P[\text{surf}|\text{web}]\, P[\text{swim}|\text{surf}] \end{aligned}$$

5. Bayesian Networks

- A Bayesian network (BN) is a **directed, acyclic graph** (V, E) with
  - Vertices V representing **random variables**
  - Edges E representing **dependencies**
  - For a root $R \in V$ the BN captures the **prior probability** $P[R = \ldots]$
  - For a vertex $X \in V$ with **parents** $parents(x) = \{P_1, \ldots, P_k\}$ the BN captures the **conditional probability** $P[X \mid P_1, \ldots, P_k]$
  - The vertex X is **conditionally independent** of a non-parent node Y given its parents $parents(x) = \{P_1, \ldots, P_k\}$, i.e.:

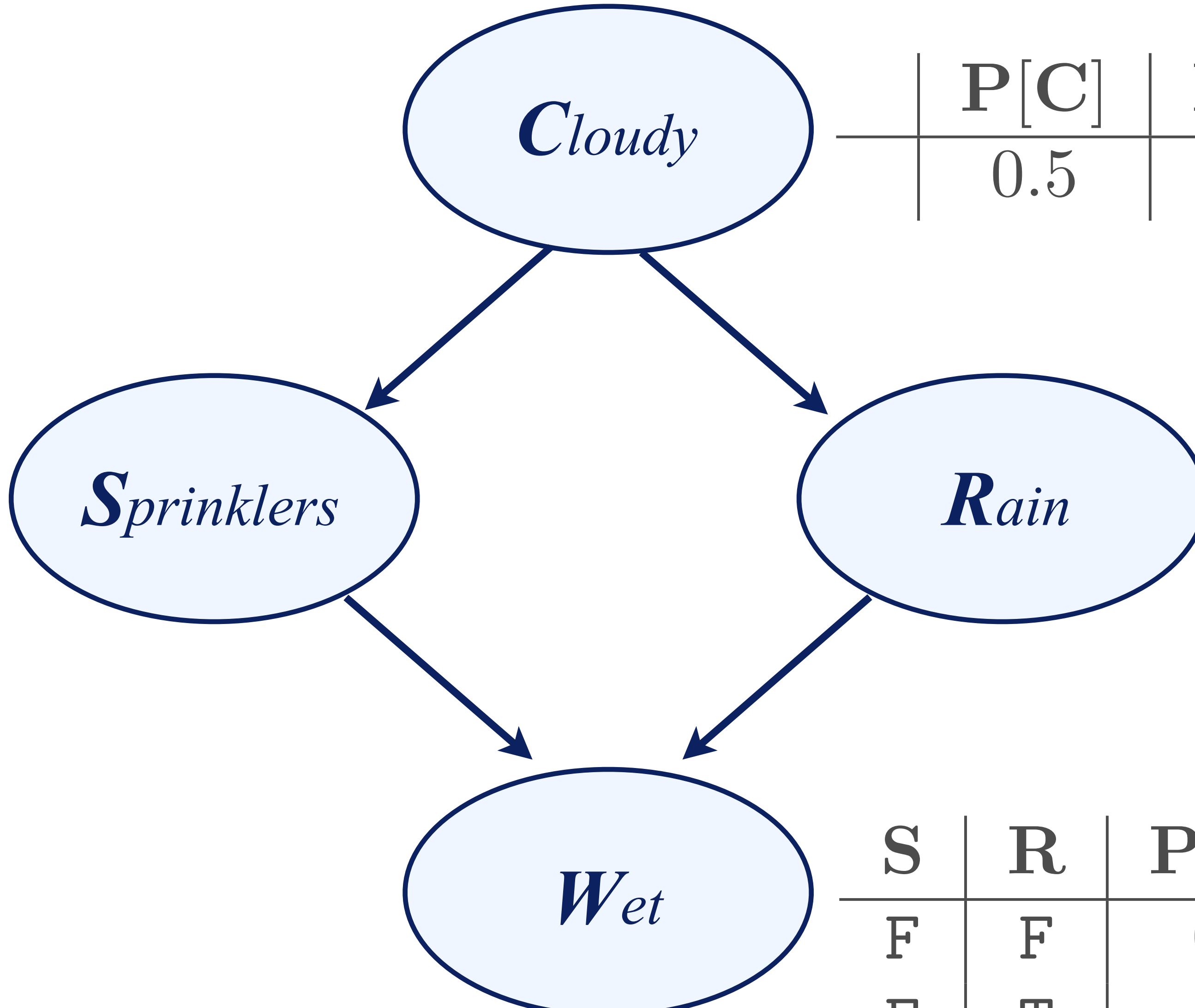
$$P[X|P_1,\ldots,P_k,Y] = P[X|P_1,\ldots,P_k]$$

Bayesian Networks (cont'd)

- We can determine any **joint probability** using the BN

$$
\begin{aligned}
& P[X_1, \ldots, X_n] \\
= \quad & P[X_1 | X_2, \ldots, X_n] \, P[X_2, \ldots, X_n] \\
= \quad & \textstyle\prod_{i=1}^{n} P[X_i | X_{i+1}, \ldots, X_n] & (\textbf{chain rule}) \\
= \quad & \textstyle\prod_{i=1}^{n} P[X_i | parents(X_i), \text{other nodes}] & (\textbf{conditional independence}) \\
= \quad & \textstyle\prod_{i=1}^{n} P[X_i | parents(X_i)]
\end{aligned}
$$

Bayesian Networks (Example)



$\mathbf{P[C]}$	$\mathbf{P[\bar{C}]}$
0.5	0.5

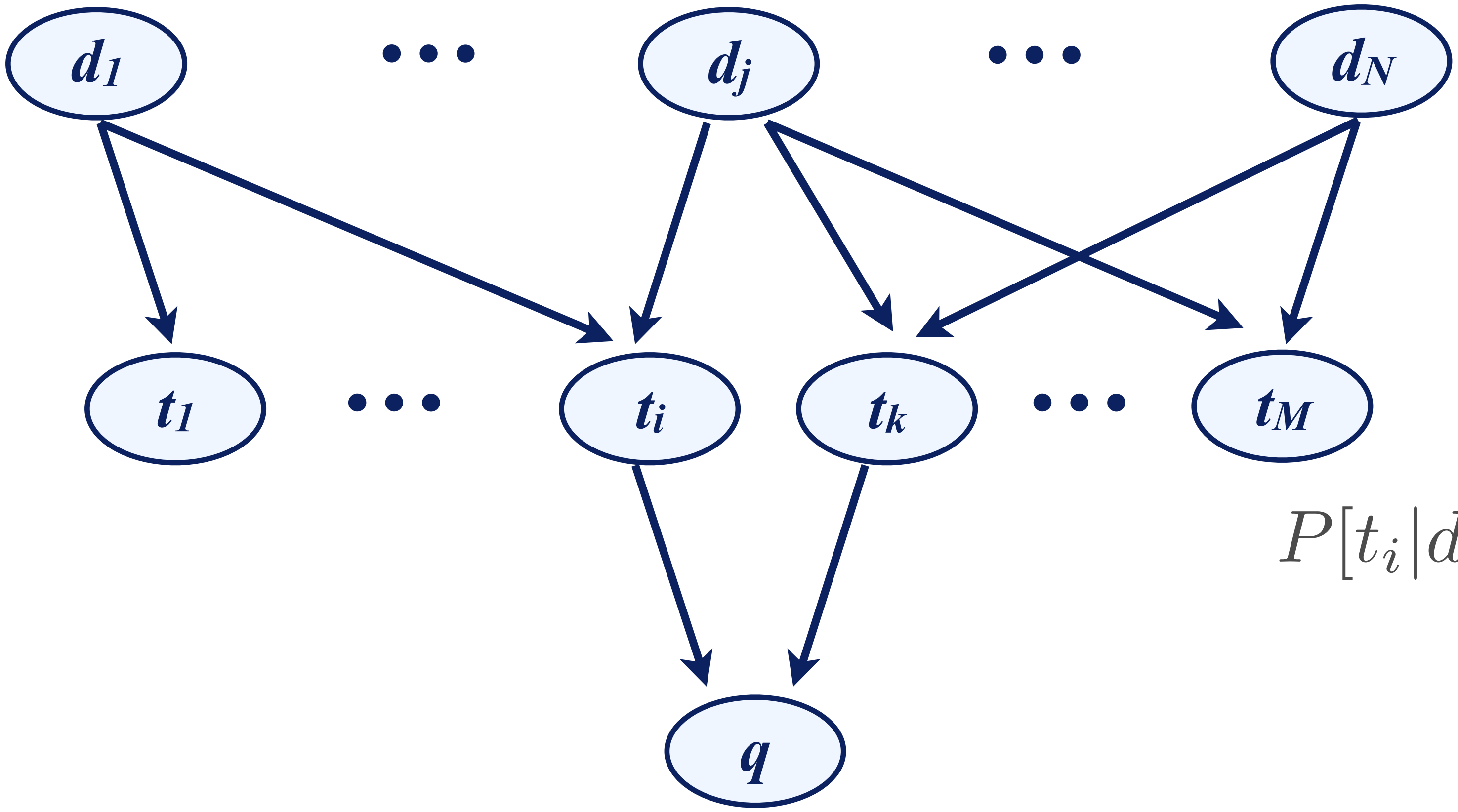
C	$\mathbf{P[S]}$	$\mathbf{P[\bar{S}]}$
F	0.5	0.5
T	0.1	0.9

C	$\mathbf{P[R]}$	$\mathbf{P[\bar{R}]}$
F	0.2	0.8
T	0.8	0.2

S	R	$\mathbf{P[W]}$	$\mathbf{P[\bar{W}]}$
F	F	0.0	1.0
F	T	0.9	0.1
T	F	0.9	0.1
T	T	1.0	0.0

$$P[C, S, \bar{R}, W] = P[C]\, P[S|C]\, P[\bar{R}|C]\, P[W|S, \bar{R}] = 0.5 \times 0.1 \times 0.2 \times 0.9$$

Bayesian Networks for IR



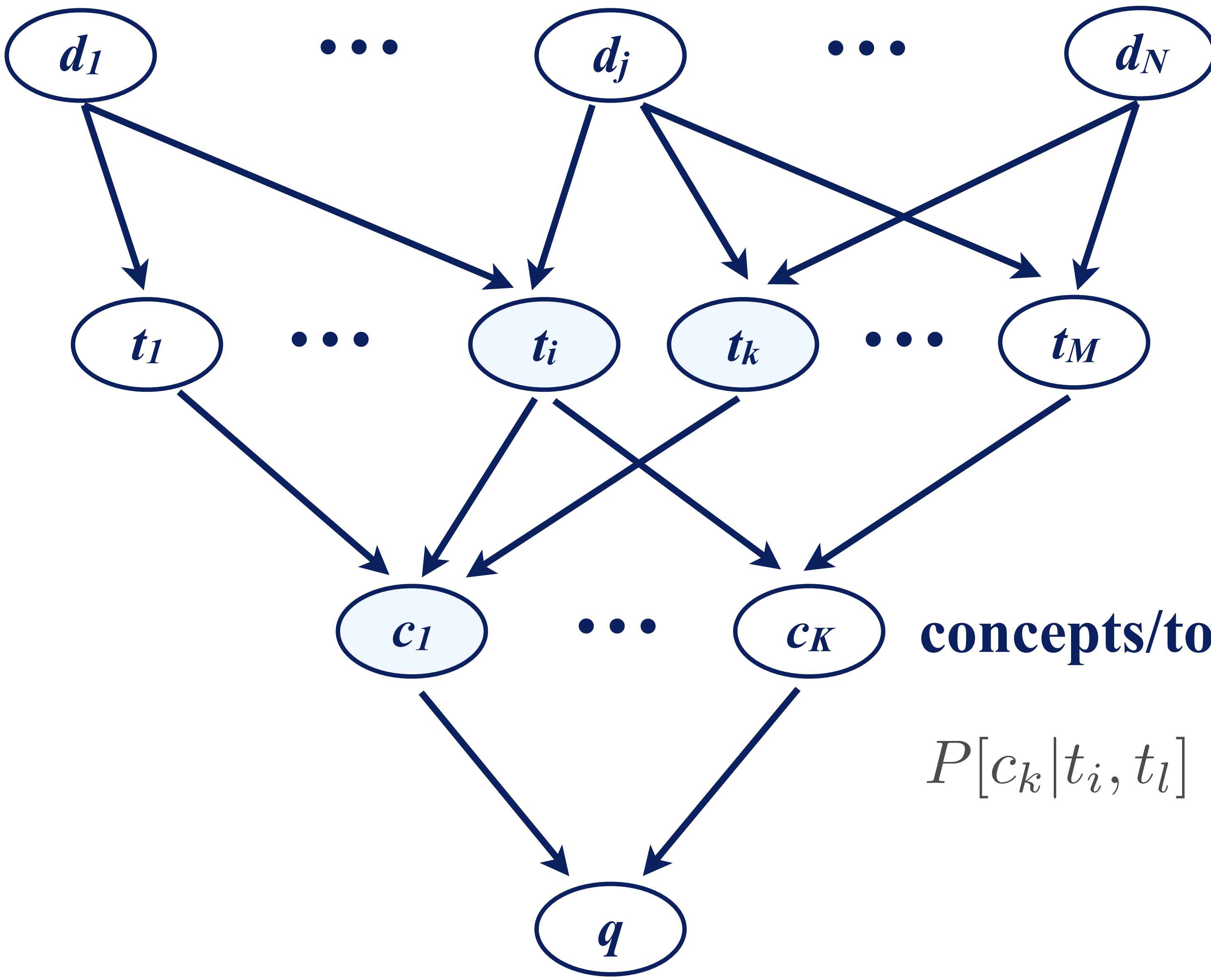
$$P[d_j] = 1/N$$

$$P[t_i|d_j] = \begin{cases} 1 & : \quad t_i \in d_j \\ 0 & : \quad \text{otherwise} \end{cases}$$

$$P[q|parents(q)] = \begin{cases} 1 & : \quad \exists t \in parents(q) \; : \; rel(t,q) \\ 0 & : \quad \text{otherwise} \end{cases}$$

$$\begin{aligned} P[q, d_j] &= \sum_{(t_1,\ldots,t_M)} P[q, d_j, t_1, \ldots, t_M] \\ &= \sum_{(t_1,\ldots,t_M)} P[q|d_j, t_1, \ldots, t_M]\, P[d_j, t_1, \ldots, t_M] \\ &= \sum_{(t_1,\ldots,t_M)} P[q|t_1, \ldots, t_M]\, P[t_1, \ldots, t_M|d_j]\, P[d_j] \end{aligned}$$

Advanced Bayesian Networks for IR



concepts/topics c_l:

$$P[c_k|t_i, t_l] \approx \frac{df_{il}}{df_i + df_l - df_{il}}$$

- BN **not widely adopted** in IR due to challenges in parameter estimation, representation, efficiency, and practical effectiveness

Summary of III.3

- **Probabilistic IR** as a family of (more) principled approaches relying on generative models of documents as bags of words
- **Probabilistic ranking principle** as the foundation establishing that ranking documents by $P[R| d, q]$ is optimal
- **Binary independence model** puts that principle into practice based on a multivariate Bernoulli model
- **Smoothing** to avoid overfitting to the training sample
- **Okapi BM25** as a state-of-the-art retrieval model based on an approximation of a 2-Poisson mixture model
- **Term dependence model** and **Bayesian networks** can consider term correlations (but are often intractable)

Additional Literature for III.3

- **F. Crestani, M. Lalmas, C. J. Van Rijsbergen, and I. Campbell**: *"Is This Document Relevant? ... Probably": A Survey of Probabilistic Models in Information Retrieval*, ACM Computing Surveys 30(4):528-552, 1998

- **S.E. Robertson, K. Spärck Jones**: *Relevance Weighting of Search Terms*, JASIS 27(3), 1976

- **S.E. Robertson, S. Walker**: *Some Simple Effective Approximations to the 2-Poisson Model for Probabilistic Weighted Retrieval*, SIGIR 1994

- **T. Roelleke**: *Information Retrieval Models: Foundations and Relationships* Morgan & Claypool Publishers, 2013

- **K. Spärck-Jones, S. Walter, S. E. Robertson**: *A probabilistic model of information retrieval: development and comparative experiments*, IP&M 36:779-840, 2000

- **K. J. van Rijsbergen**: *Information Retrieval*, University of Glasgow, 1979 http://www.dcs.gla.ac.uk/Keith/Preface.html