CS657A: Information Retrieval
Bayesian Network Models

Arnab Bhattacharya
arnabb@cse.iitk.ac.in

Computer Science and Engineering,
Indian Institute of Technology, Kanpur
http://web.cse.iitk.ac.in/~cs657/

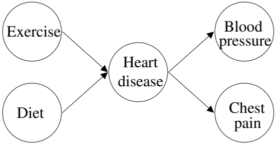
2nd semester, 2021-22

Tue 1030-1145, Thu 1200-1315

Bayesian Networks

- Bayesian networks or Bayesian belief networks or Bayes nets or belief nets
- Takes into account the correlations of parameters by modeling them as conditional probabilities
- Forms a directed acyclic graph (DAG)
- Edges model the *dependencies*
- Parent is the *cause* and children are the *effects*
- A node is conditionally independent of all its non-descendants given its parents
- For every node, there is a conditional probability table (CPT) that describes its values given its parents' values
- CPT for node X is of the form $P(X|parents(X))$

Example



- CPTs: rows are values; columns are parents (i.e., conditionals)
- Last rows can be inferred, and therefore, omitted

Exercise (E)	Φ
regular (r)	0.70
irregular (i)	0.30

Diet (D)	Φ
healthy (h)	0.25
unhealthy (u)	0.75

Heart disease (H)	E=r, D=h	E=r, D=u	E=i, D=h	E=i, D=u
yes (y)	0.25	0.40	0.55	0.80
no (n)	0.75	0.60	0.45	0.20

Blood pressure (B)	H=y	H=n
normal (l)	0.15	0.80
high (g)	0.85	0.20

Chest pain (C)	H=y	H=n
normal (m)	0.70	0.45
pain (p)	0.30	0.55

Probabilities using Bayesian Networks

- Given no prior information, is a person suffering from heart disease?
- Note that no other information (e.g., chest pain, etc.) are known
- Compute $P(H = y)$; if it is greater than $P(H = n)$, then predict "heart disease"

$$
\begin{aligned}
P(H = y) &= \sum_{\alpha,\beta} [P(H = y | E = \alpha, D = \beta).P(E = \alpha, D = \beta)] \\
&= \sum_{\alpha,\beta} [P(H = y | E = \alpha, D = \beta).P(E = \alpha).P(D = \beta)] \\
&= 0.25 \times 0.70 \times 0.25 + 0.40 \times 0.70 \times 0.75 \\
&\quad + 0.55 \times 0.30 \times 0.25 + 0.80 \times 0.30 \times 0.75 \\
&= 0.475
\end{aligned}
$$

Probabilities using some Information

- Given a person has high blood pressure, is she suffering from heart disease?
- Note that not all information (e.g., chest pain, etc.) are known
- Compute $P(H = y|B = g)$; if it is greater than $P(H = n|B = g)$, then predict "heart disease"

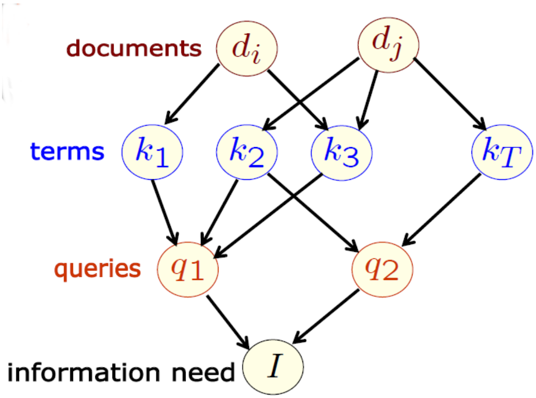
$$
\begin{aligned}
P(H = y|B = g) &= \frac{P(B = g|H = y).P(H = y)}{P(B = g)} \\
&= \frac{P(B = g|H = y).P(H = y)}{\sum_{\alpha} [P(B = g|H = \alpha).P(H = \alpha)]} \\
&= \frac{0.85 \times 0.475}{0.85 \times 0.475 + 0.20 \times 0.525} \\
&= 0.794
\end{aligned}
$$

Probabilities using Information about Parents

- Given a person has high blood pressure, unhealthy diet and irregular exercise, is she suffering from heart disease?
- Note that not all information (e.g., chest pain, etc.) are known
- Compute $P(H = y|B = g, D = u, E = i)$; if it is greater than $P(H = n|B = g, D = u, E = i)$, then predict "heart disease"

$$
\begin{aligned}
&P(H = y|B = g, D = u, E = i) \\
&\qquad = \frac{P(B = g|H = y, D = u, E = i).P(H = y|D = u, E = i)}{P(B = g|D = u, E = i)} \\
&\qquad = \frac{P(B = g|H = y).P(H = y|D = u, E = i)}{\sum_\alpha [P(B = g|H = \alpha).P(H = \alpha|D = u, E = i)]} \\
&\qquad = \frac{0.85 \times 0.80}{0.85 \times 0.80 + 0.20 \times 0.20} \\
&\qquad = 0.944
\end{aligned}
$$

Inference Network



- Non-independent model
- Epistemological view instead of frequentist
- Information need *I* can be represented by multiple queries
  - Query expansion
  - Multiple queries can satisfy *I*

Ranking of Documents

- How much evidence does document d_j provide for query q
- Ranking is based on $P(q|d_j)$ or equivalently $P(q \wedge d_j)/P(d_j)$
- Using terms k_i
- $\vec{k}$ represents the Boolean state of all t term variables k_i

$$\begin{aligned} P(q \wedge d_j) &= \sum_{\forall \vec{k}} P(q \wedge d_j|\vec{k}) \times P(\vec{k}) = \sum_{\forall \vec{k}} P(q \wedge d_j \wedge \vec{k}) \\ &= \sum_{\forall \vec{k}} P(q|d_j \wedge \vec{k}) \times P(d_j \wedge \vec{k}) = \sum_{\forall \vec{k}} P(q|\vec{k}) \times P(\vec{k}|d_j) \times P(d_j) \end{aligned}$$

- Terms k_i are *conditionally independent* given the document

$$P(\vec{k}|d_j) = \prod_{\forall k_i \in d_j} P(k_i|d_j) \times \prod_{\forall k_i \notin d_j} (1 - P(k_i|d_j))$$

  - $P(\vec{k}|d_j)$ are *pre-computed*
  - Only $P(q|\vec{k})$ is computed at query time

Priors for Boolean Model

- Document prior probability is simply *uniform*

$$P(d_j) = \frac{1}{n}$$

- Probability of term k_i in d_j is Boolean

$$P(k_i|d_j) = \begin{cases} 1 & \text{if } k_i \in d_j \\ 0 & \text{otherwise} \end{cases}$$

- Probability of query q given term vector $\vec{k}$

$$P(q|\vec{k}) = \begin{cases} 1 & \text{if } \vec{k} \text{ satisfies } q \\ 0 & \text{otherwise} \end{cases}$$

- This retrieves the original Boolean model
- $P(I|d_j) = \sum_{\forall q} P(I|q).P(q|d_j)$

Tf-idf

- $P(k_i|d_j) \propto tf$ (normalized tf)
- $P(q|k_i) \propto idf$ (normalized idf)
- $P(d_j) \propto 1/dl$ (document length)
- Model closely resembles tf-idf with document length normalization
- However, different normalization constants for different documents
  - Such that probabilities add up to 1
- Hence, not exactly tf-idf

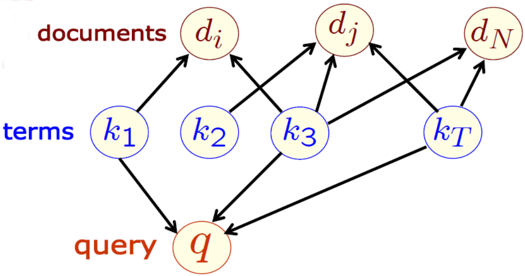
Link Matrices

- Given multiple parents, probability of child node requires exponential computation
- Link matrices simplify by specifying (Boolean) formulae
- For example, if $P(I|q_1, q_2)$ is AND

$$
\begin{aligned}
P(I) &= \sum_{\forall q_1=a, q_2=b} P(q_1 = T|q_1 = a, q_2 = b).P(q_1 = a).P(q_2 = b) \\
&= P(t|f, f).(1-p_1).(1-p_2) + P(t|f, t).(1-p_1).p_2 \\
&\qquad + P(t|t, f).p_1.(1-p_2) + P(t|t, t).p_1.p_2 \\
&= 0.(1-p_1).(1-p_2) + 0.(1-p_1).p_2 + 0.p_1.(1-p_2) + 1.p_1.p_2 \\
&= p_1.p_2
\end{aligned}
$$

- OR: $P(I) = 1 - (1-p_1).(1-p_2)$
- SUM: $P(I) = (p_1 + p_2)/2$
- Reduces to linear computation

Belief Network



- Direction between terms and documents is reversed
- Documents are also caused by terms
- Query is treated like a document, similar to vector space model

Ranking of Documents

- How much evidence does document d_j provide for query q
- Ranking is based on $P(d_j|q)$ or equivalently $P(d_j \wedge q)$
- Using terms k_i
- $\vec{k}$ represents the Boolean state of all t term variables k_i

$$\begin{aligned} P(d_j \wedge q) &= \sum_{\forall \vec{k}} P(d_j \wedge q|\vec{k}) \times P(\vec{k}) \\ &= \sum_{\forall \vec{k}} P(d_j|\vec{k}) \times P(q|\vec{k}) \times P(\vec{k}) \end{aligned}$$

- $P(d_j|\vec{k})$ and $P(q|\vec{k})$ are length-normalized tf's

$$P(d_j|\vec{k}) = \frac{tf_i}{dl_j} \ \forall t_i \in \vec{k}$$

$$dl_j = ||tf_{j,i}||_2$$

Discussion on Bayesian Networks

- Inference network "believes" that documents dictate words
- Belief network models reality better and is easier to explain
- Inference network uses $P(\vec{k}|d_j)$ which is nicely separable into individual terms
- Belief network uses $P(d_j|\vec{k})$ which is not separable but can use any function
- Belief network is more natural
- Since query and document are separated and follow the same process, it is easier to incorporate other evidential information such as user history and relevance feedback into belief networks
- Belief networks can subsume other IR models and are more general