Retrieval Models: Probabilistic

Intelligent Information Retrieval

The Notion of Relevance

- Relevance
  - **Relevance constraints** (Fang et al. 04)
  - $\Delta(Rep(q), Rep(d))$ **Similarity**
    - **Different rep & similarity**
      - **Vector space model** (Salton et al., 75)
      - …
      - **Prob. distr. model** (Wong & Yao, 89)
  - $P(r = 1|q, d) \quad r \in \{0, 1\}$ **Probability of Relevance**
    - **Regression Model** (Fox 83)
    - **Generative Model**
      - **Doc generation**
        - **Classical prob. Model** (Robertson & Sparck Jones, 76)
      - **Query generation**
        - **LM approach** (Ponte & Croft, 98) (Lafferty & Zhai, 01a)
  - $P(d \rightarrow q)$ *or* $P(q \rightarrow d)$ **Probabilistic inference**
    - **Different inference system**
      - **Prob. concept space model** (Wong & Yao, 95)
      - **Inference network model** (Turtle & Croft, 91)

Probability Ranking Principle

[Robertson 77]

- Returning a ranked list of documents in descending order of probability that a document is relevant to the query is the optimal strategy under the following two assumptions:

1. The utility of a document to a user is independent of the utility of any other document.
2. A user would browse the results sequentially

According to the PRP, all we need is

"A relevance measure function f"

which satisfies

For all q, d_1, d_2,

$f(q, d_1) > f(q, d_2)$ **iff** $p(Rel|q, d_1) > p(Rel|q, d_2)$

The Basic Question

What is the probability that THIS document is relevant to THIS query?

Formally…

3 random variables: query Q, document D, relevance $R \in \{0,1\}$

Given a particular query q, a particular document d, $p(R = 1|Q = q, D = d) = ?$

Detour ...

Brief Review of Probability

Basic Concepts in Probability

- **Random experiment: an experiment with uncertain outcome** (e.g., tossing a coin, picking a word from text)
- **Sample space: all possible outcomes, e.g.,**
  - Tossing 2 fair coins, $S = \{HH, HT, TH, TT\}$
- **Event: $\boldsymbol{E \subseteq S}$, $\boldsymbol{E}$ happens iff outcome is in $\boldsymbol{E}$, e.g.,**
  - $E = \{HH\}$ (all heads)
  - $E = \{HH, TT\}$ (same face)
- **Probability of Event : $\boldsymbol{1 \geq P(E) \geq 0}$, s.t.**
  - $P(S) = 1$ (outcome always in S)
  - $P(A \cup B) = P(A) + P(B)$ if $(A \cap B) = \emptyset$ (e.g., A=same face, B=different face)

Basic Concepts of Prob. (cont.)

- **Conditional Probability:** $\boldsymbol{P(B|A) = P(A \cap B)/P(A)}$
  - $P(A \cap B) = P(A)P(B|A) = P(B)P(A|B)$
  - So, $P(A|B) = P(B|A)P(A)/P(B)$ **(Bayes' Rule)**
  - For **independent events**, $P(A \cap B) = P(A)P(B)$, so $P(A|B) = P(A)$
- **Total probability: If $\boldsymbol{A_1, \ldots, A_n}$ form a partition of $\boldsymbol{S}$, then**
  - $P(B) = P(B \cap S) = P(B \cap A_1) + \cdots + P(B \cap A_n)$
  - So, $P(A_i|B) = P(B|A_i)P(A_i)/P(B)$

    $= P(B|A_i)P(A_i)/[P(B|A_1)P(A_1) + \cdots + P(B|A_n)P(A_n)]$
  - This allows us to compute $P(A_i|B)$ based on $P(B|A_i)$

Interpretation of Bayes' Rule

Hypothesis space: $H = \{H_1, \dots, H_n\}$ Evidence: E

$$P(H_i|E) = \frac{P(E|H_i)P(H_i)}{P(E)}$$

If we want to pick the most likely hypothesis H^*, we can drop $P(E)$

Posterior probability of H_i **Prior probability of H_i**

$$P(H_i|E) \propto P(E|H_i)P(H_i)$$

Likelihood of data/evidence if H_i is true

End of Detour...

Now, let's go back to retrieval!

Probabilistic Retrieval Models: Intuitions

Suppose we have a large number of relevance judgments
(e.g. clickthroughs: “1”=clicked; “0”=skipped)

Query(Q)	Doc(D)	Rel(R) ?
Q1	D1	1
Q1	D2	1
Q1	D3	0
Q1	D4	0
Q1	D5	1
…		
Q1	D1	0
Q1	D2	1
Q1	D3	0
Q2	D3	1
Q3	D1	1
Q4	D2	1
Q4	D3	0
…		

We can score documents based on:

$$f(q,d) = p(R=1|d,q) = \frac{count(q,d,R=1)}{count(q,d)}$$

$$P(R=1|Q1,D1) = 1/2$$
$$P(R=1|Q1,D2) = 2/2$$
$$P(R=1|Q1,D3) = 0/2$$

What if we don’t have sufficient search log?
Unseen documents/queries?

We can approximate $P(R=1|Q,D)$

Probability of Relevance

- Three random variables
  - Query Q
  - Document D
  - Relevance $R \in \{0,1\}$
- Goal: rank D based on $P(R = 1|Q, D)$
  - Evaluate $P(R = 1|Q, D)$
  - Actually, only need to compare $P(R = 1|Q, D1)$ with $P(R = 1|Q, D2)$, i.e., rank documents
- Several different ways to refine $P(R = 1|Q, D)$

Refining $P(R=1|Q,D)$ Method 1: conditional models

- Basic idea: relevance depends on how well a query matches a document
  - Define features on $Q \times D$, e.g., #matched terms, the highest IDF of a matched term, doclen, or even $score(Q,D)$ given any other retrieval function,…
  - $P(R=1|Q,D) = g(f_1(Q,D), f_2(Q,D), \ldots, f_n(Q,D), \theta)$
  - Using training data (known relevance judgments) to estimate parameter θ
  - Apply the model to rank new documents

Refining $P(R = 1|Q, D)$ Method 2: generative models

- Basic idea
  - Define $P(Q, D|R)$
  - Compute $O(R = 1|Q, D)$ using Bayes' rule

$$O(R = 1|Q, D) = \frac{P(R = 1|Q, D)}{P(R = 0|Q, D)} = \frac{P(Q, D|R = 1)}{P(Q, D|R = 0)} \times \boxed{\frac{P(R = 1)}{P(R = 0)}}$$

Ignored for ranking D

- Special cases
  - Document "generation": $P(Q, D|R) = P(D|Q, R)P(Q|R)$
  - Query "generation": $P(Q, D|R) = P(Q|D, R)P(D|R)$

Document Generation

$$O(R=1|\mathrm{Q},\mathrm{D}) = \frac{P(R=1|Q,D)}{P(R=0|Q,D)} = \frac{p(Q,D|R=1)P(R=1)}{P(Q,D|R=0)P(R=0)}$$

$$\propto \frac{P(Q,D|R=1)}{P(Q,D|R=0)} = \frac{P(D|Q,R=1)P(Q|R=1)}{P(D|Q,R=0)P(Q|R=0)}$$

$$\propto \frac{P(D|Q,R=1)}{P(D|Q,R=0)}$$

Model of relevant docs for Q

Model of non-relevant docs for Q

Document Generation (cont'd)

- **Assume independent attributes $\boldsymbol{A_1 \dots A_k}$**
- **Let $\boldsymbol{D = d_1 \dots d_k}$, where $\boldsymbol{d_i \in \{0, 1\}}$ is the value of attribute $\boldsymbol{A_i}$ (Similarly $\boldsymbol{Q = q_1 \dots q_k}$)**

$$\dots \propto \frac{P(D|Q,R=1)}{P(D|Q,R=0)}$$

$$= \prod_{i=1}^{k} \frac{P(A_i = d_i|Q,R=1)}{P(A_i = d_i|Q,R=0)}$$

$$= \prod_{i=1,d_i=1}^{k} \frac{P(A_i = 1|Q,R=1)}{P(A_i = 1|Q,R=0)} \prod_{i=1,d_i=0}^{k} \frac{P(A_i = 0|Q,R=1)}{P(A_i = 0|Q,R=0)}$$

Document Generation (cont'd)

$$\ldots = \prod_{i=1,d_i=1}^{k} \frac{P(A_i = 1|Q, R = 1)}{P(A_i = 1|Q, R = 0)} \prod_{i=1,d_i=0}^{k} \frac{P(A_i = 0|Q, R = 1)}{P(A_i = 0|Q, R = 0)}$$

$$\times \prod_{i=1,d_i=1}^{k} \frac{P(A_i = 0|Q, R = 1)}{P(A_i = 0|Q, R = 0)} \times \prod_{i=1,d_i=1}^{k} \frac{P(A_i = 0|Q, R = 0)}{P(A_i = 0|Q, R = 1)}$$

$$= \prod_{i=1,d_i=1}^{k} \frac{P(A_i = 1|Q, R = 1)}{P(A_i = 1|Q, R = 0)} \cdot \frac{P(A_i = 0|Q, R = 0)}{P(A_i = 0|Q, R = 1)} \times \boxed{\prod_{i=1}^{k} \frac{P(A_i = 0|Q, R = 1)}{P(A_i = 0|Q, R = 0)}}$$

Ignore for ranking

Document Generation (cont'd)

- Assume $P(A_i = 1|Q, R = 1) = P(A_i = 1|Q, R = 0)$ if the term does not appear in the query ($q_i = 0$)

$$\ldots = \prod_{i=1, d_i=1}^{k} \frac{P(A_i = 1|Q, R = 1)}{P(A_i = 1|Q, R = 0)} \cdot \frac{P(A_i = 0|Q, R = 0)}{P(A_i = 0|Q, R = 1)} \times \boxed{\prod_{i=1}^{k} \frac{P(A_i = 0|Q, R = 1)}{P(A_i = 0|Q, R = 0)}}$$

Ignore for ranking

$$= \prod_{i=1, d_i=q_i=1}^{k} \frac{P(A_i = 1|Q, R = 1)}{P(A_i = 1|Q, R = 0)} \cdot \frac{P(A_i = 0|Q, R = 0)}{P(A_i = 0|Q, R = 1)}$$

Robertson-Sparck Jones Model

(Robertson & Sparck Jones 76)

$$\log O(R=1|Q,D) \overset{Rank}{\approx} \sum_{i=1,d_i=q_i=1}^{k} \log \frac{p_i(1-q_i)}{q_i(1-p_i)} \qquad \text{(RSJ model)}$$

Two parameters for each term A_i:

$\boldsymbol{p_i = P(A_i = 1|Q, R = 1)}$: **prob. that term A_i occurs in a relevant doc**

$\boldsymbol{q_i = P(A_i = 1|Q, R = 0)}$: **prob. that term A_i occurs in a non-relevant doc**

How to estimate parameters?

Suppose we have relevance judgments,

$$\widehat{p_i} = \frac{\#(rel.doc\ with\ A_i) + 0.5}{\#(rel.doc) + 1} \qquad \widehat{q_i} = \frac{\#(nonrel.doc\ with\ A_i) + 0.5}{\#(nonrel.doc) + 1}$$

RSJ Model: No Relevance Info

(Croft & Harper 79)

$$\log O(R=1|Q,D) \overset{Rank}{\approx} \sum_{i=1,d_i=q_i=1}^{k} \log \frac{p_i(1-q_i)}{q_i(1-p_i)} \qquad \text{(RSJ model)}$$

How to estimate parameters?

Suppose we do not have relevance judgments,

- We will assume p_i to be a constant
- Estimate q_i by assuming all documents to be non-relevant

$$\log O(R=1|Q,D) \overset{Rank}{\approx} \sum_{i=1,d_i=q_i=1}^{k} \log \frac{N-n_i+0.5}{n_i+0.5}$$

N: # documents in collection

n_i: # documents in which term A_i occurs

RSJ Model: Summary

- The most important classic prob. IR model
- Use only term presence/absence, thus also referred to as Binary Independence Model
- Essentially Naïve Bayes for doc ranking
- Most natural for relevance/pseudo feedback
- When without relevance judgments, the model parameters must be estimated in an ad hoc way
- Performance isn’t as good as tuned VS model

Improving RSJ: Adding TF

Basic doc. generation model: $$\frac{P(R=1|Q,D)}{P(R=0|Q,D)} \propto \frac{P(D|Q,R=1)}{P(D|Q,R=0)}$$

Let $D = d_1 \ldots d_k$, where d_k is the frequency count of term A_k

$$\frac{P(R=1|Q,D)}{P(R=0|Q,D)} \propto \prod_{i=1}^{k} \frac{P(A_i = d_i | Q,R=1)}{P(A_i = d_i | Q,R=0)}$$
$$= \prod_{i=1, d_i \ge 1}^{k} \frac{P(A_i = d_i | Q,R=1)}{P(A_i = d_i | Q,R=0)} \prod_{i=1, d_i = 0}^{k} \frac{P(A_i = 0 | Q,R=1)}{P(A_i = 0 | Q,R=0)}$$
$$\propto \prod_{i=1, d_i \ge 1}^{k} \frac{P(A_i = d_i | Q,R=1) P(A_i = 0 | Q,R=0)}{P(A_i = d_i | Q,R=0) P(A_i = 0 | Q,R=1)}$$

2-Poisson mixture model $$p(A_i = f | Q,R) = p(E|Q,R) p(A_i = f | E) + P(\overline{E} | Q,R) p(A_i = f | \overline{E})$$

Many more parameters to estimate!

BM25/Okapi Approximation
(Robertson et al. 94)

- Idea: Approximate $p(R=1|Q,D)$ with a simpler function that shares similar properties
- Observations:
  - $\log O(R=1|Q,D)$ is a sum of term weights W_i
  - $W_i = 0$, if $TF_i = 0$
  - W_i increases monotonically with TF_i
  - W_i has an asymptotic limit
- The simple function is $W_i = \frac{TF_i(k_1+1)}{k_1+TF_i} \log \frac{p_i(1-q_i)}{q_i(1-p_i)}$

Adding Doc. Length & Query TF

- Incorporating doc length
  - “Carefully” penalize long doc
- Incorporating query TF
  - A similar TF transformation
- The final formula is called BM25, achieving top TREC performance

The BM25 Formula

$$\sum_{T \in Q} w^{(1)} \frac{(k_1+1)tf}{K+tf} \frac{(k_3+1)\,qtf}{k_3+qtf} \tag{1}$$

"Okapi TF/BM25 TF"

where

Q is a query, containing terms T
$w^{(1)}$ is the Robertson/Sparck Jones weight [5] of T in Q

$$\log \frac{(r+0.5)/(R-r+0.5)}{(n-r+0.5)/(N-n-R+r+0.5)} \tag{2}$$

N is the number of items (documents) in the collection
n is the number of documents containing the term
R is the number of documents known to be relevant to a specific topic
r is the number of relevant documents containing the term
K is $k_1((1-b)+b.dl/avdl)$
k_1, b and k_3 are parameters which depend on the on the nature of the queries and possibly on the database; k_1 and b default to 1.2 and 0.75 respectively, but smaller values of b are sometimes advantageous; in long queries k_3 is often set to 7 or 1000 (effectively infinite)
tf is the frequency of occurrence of the term within a specific document
qtf is the frequency of the term within the topic from which Q was derived
dl and $avdl$ are respectively the document length and average document length measured in some suitable unit.

Extensions of “Doc Generation” Models

- Capture term dependence (Rijsbergen & Harper 78)
- Alternative ways to incorporate TF (Croft 83, Kalt96)
- Feature/term selection for feedback (Okapi’s TREC reports)
- Estimate of the relevance model based on pseudo feedback [Lavrenko & Croft 01]

Questions?