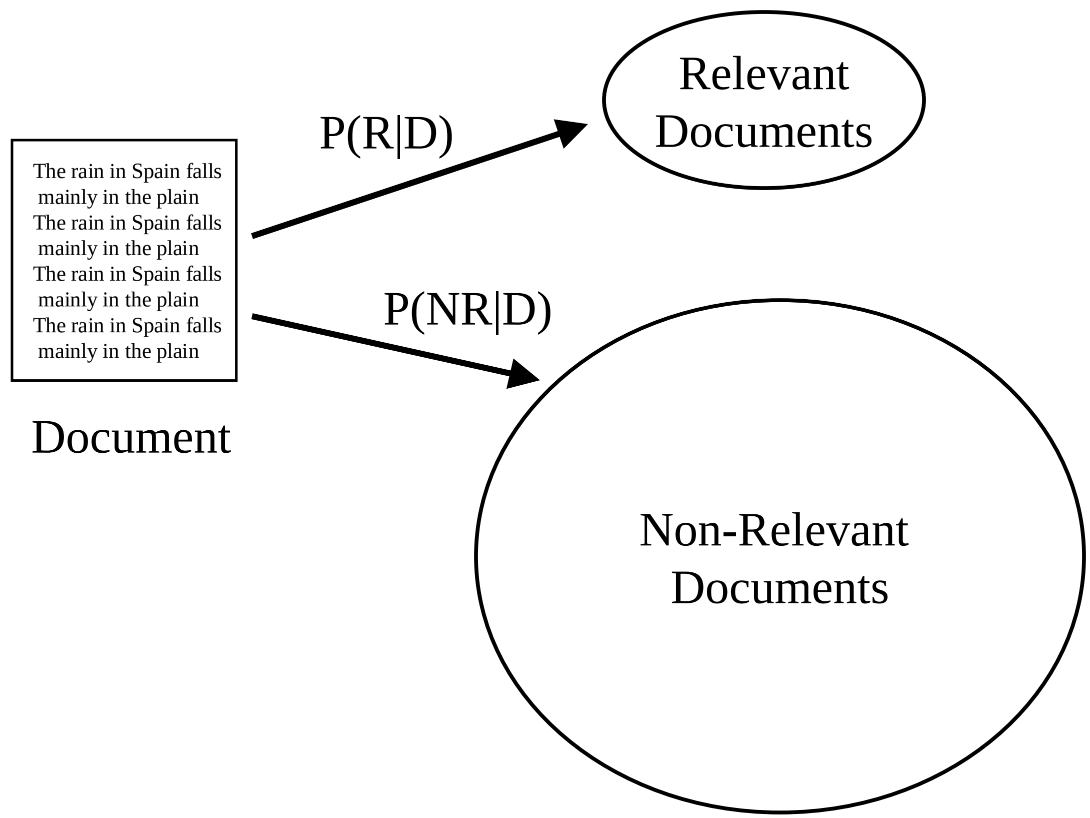
Basic Probabilistic Retrieval Model

- Retrieval is modeled as a classification process
- Two classes for each query: the *relevant* and *non-relevant* documents
  - could easily be extended to three classes (i.e. add a *don't care*)
- Given a particular document D, calculate the probability of belonging to the relevant class, retrieve if greater than probability of belonging to non-relevant class
  - i.e. retrieve if $P(R|D) > P(NR|D)$
- Equivalently, rank by *likelihood ratio* $P(D|R)/P(D|NR)$
- Different ways of estimating these probabilities lead to different models

Event Space

- The probability of an event is the sum of the probabilities of the sample points associated with the event
- Sample points represent the possible outcomes of a statistical “experiment”
- For a retrieval model, the event space is $Q \times D$, where each sample point is a query-document pair and has an associated relevance judgement
- Because Q and D are representations, there (conceptually) may be many pairs with Q and D the same but with different relevance judgements
- $P(R|Q,D)$ is the proportion of the identical (Q,D) pairs that are judged relevant
- In the following, we assume a given Q and use $P(R|D)$

Basic Probabilistic Model



Probability Ranking Principle

- Assume relevance of a document is independent of other documents in the collection
- Ranking documents in decreasing order of probability of relevance to the user who submitted the query, where probabilities are estimated using all available evidence, produces the best possible effectiveness
- Bayes Decision Rule: Retrieve if $P(R|D) > P(NR|D)$
  - minimizes the average probability of error
  - Probability of error

$$P(\text{error} \mid D) = \begin{cases} P(R \mid D) & \text{if we decide NR} \\ P(NR \mid D) & \text{if we decide R} \end{cases}$$

  - equivalent to optimizing recall/fallout tradeoff

Basic Probabilistic Model

- Next assume D is represented by a binary vector $\mathbf{d} = (d_1, d_2, \ldots d_n)$ where $d_i = 0$ or 1 indicates the absence or presence of the index term
- $p_i = P(d_i = 1|R)$
- $q_i = P(d_i = 1|NR)$
- Assume conditional independence, write $P(\mathbf{d}|R)$ as the product of the probabilities for the components of **d** (I.e. product of probabilities of getting a particular vector of 1's and 0's
- Likelihood functions are

$$P(\mathbf{d} \mid R) = \prod_{i=1}^{n} p_i^{d_i} (1 - p_i)^{1-d_i}$$

$$P(\mathbf{d} \mid NR) = \prod_{i=1}^{n} q_i^{d_i} (1 - q_i)^{1-d_i}$$

$$\frac{P(\mathbf{d} \mid R)}{P(\mathbf{d} \mid NR)} = \prod_{i=1}^{n} \left(\frac{p_i}{q_i} \right)^{d_i} \left(\frac{1 - p_i}{1 - q_i} \right)^{1-d_i}$$

(sometimes called "linked dependence" assumption)

Basic Probabilistic Model

- Convert to a linear discriminant function
  - $g(\mathbf{d}) = \log P(\mathbf{d}|R)/P(\mathbf{d}|NR) + \log P(R)/P(NR)$
  - $$g(\mathbf{d}) = \sum_{i=1}^{n}\left[d_i \log\frac{p_i}{q_i} + (1-d_i)\log\frac{1-p_i}{1-q_i}\right] + \log\frac{P(R)}{P(NR)}$$

$$g(\mathbf{d}) = \sum_{i=1}^{n} d_i \log\frac{p_i(1-q_i)}{q_i(1-p_i)} + \left[\sum_{i=1}^{n}\log\frac{1-p_i}{1-q_i} + \log\frac{P(R)}{P(NR)}\right]$$

- The second term is a constant for a given query and does not affect a ranking
- Assume that for terms not in the query, $p_i = q_i$
  - this can be changed
- The *retrieval status value* for a document is computed by summing the weight $\log(p_i/1-p_i) + \log(1-q_i/q_i)$ for all query terms in the document

Estimation

- Must estimate p_i and q_i
- Given information about relevant documents, we would have the following contingency table:

	Relevant	Not Relevant	
$d_i = 1$	r	n - r	n
$d_i = 0$	R – r	N – n – R + r	N – n
	R	N – R	

- Use maximum likelihood estimators
  - $p_i = r/R \quad q_i = (n-r)/(N-R)$
  - Robertson and Sparck Jones F4 weight: log (r/(R-r))/((n-r)/N-n-R+r)
- Relevance information is usually not available (except after relevance feedback)
- Estimate probabilities based on information in the query, the document collection
  - previous queries can also be used with some learning approaches

Estimation

- If q_i (probability of occurrence in non-relevant documents) is estimated as n/N, the second part of the weight is

$$\log \frac{1-q_i}{q_i} = \log \frac{N - n}{n}$$

  - which for large N is the IDF weight
  - non-relevant documents are approximated by the whole collection
  - note: could just calculate P(R|D) and then use P(D) instead of P(D|NR)
- p_i (probability of occurrence in relevant documents) can be estimated in various ways
  - constant (Croft and Harper combination match)
  - proportional to probability of occurrence in collection
  - more accurately, proportional to log(probability of occurrence)
    - Greif, 1998

Estimation

- Maximum likelihood estimates have problems with small samples or zero values
- Standard statistical practice is Bayesian estimates of the form $\hat{p} = \frac{x+a}{n+a+b}$
  - where x is the number of successes in n trials
  - a and b are parameters determined by the combination of assumptions about prior distributions and loss functions
- Typically use $a = b = 0.5$
- Estimating probabilities is the same problem as determining weighting formulae in less formal models