Information Retrieval

Retrieval Models II: LM and DFR

Qianni Zhang

Roadmap of this lecture

- A recap
- Language model
- Divergence from randomness model

Okapi BM25

$$RSV^{BM25} = \sum_{i \in q} \log \frac{N}{df_i} \cdot \frac{(k_1 + 1)tf_i}{k_1((1-b) + b\frac{dl}{avdl}) + tf_i}$$

- k_1 controls term frequency scaling
  - $k_1 = 0$ no term frequency - binary model;
  - $k_1 =$ large is raw term frequency
- b controls document length normalization
  - $b = 0$ is no scaling by document length;
  - $b = 1$ is relative frequency (fully scale by document length)
- Typically, k_1 is set around [1.2,2] and b around 0.75

Okapi BM25

- Many formulations and interpretations of BM25
- $\log \frac{N}{df_i}$: the IDF term
- Can be re-written as $\log \frac{N - n(q_i) + 0.5}{n(q_i) + 0.5}$
- N is the total number of documents in the collection
- $n(q_i)$ is the number of documents containing q_i

Generative Probabilistic Models

Generative Probabilistic Models

- The generative approach

  A generator which produces events/tokens with some probability

  - URN Metaphor: a bucket of different colour balls (10 red, 5 blue, 3 yellow, 2 white)
    - What is the probability of drawing a yellow ball?
    - What is the probability of drawing (with replacement) a red ball and a white ball?
  - IR Metaphor: Documents are urns, full of tokens (balls) of (in) different terms (colours)

Generative Probabilistic Models

Generative Probabilistic Models

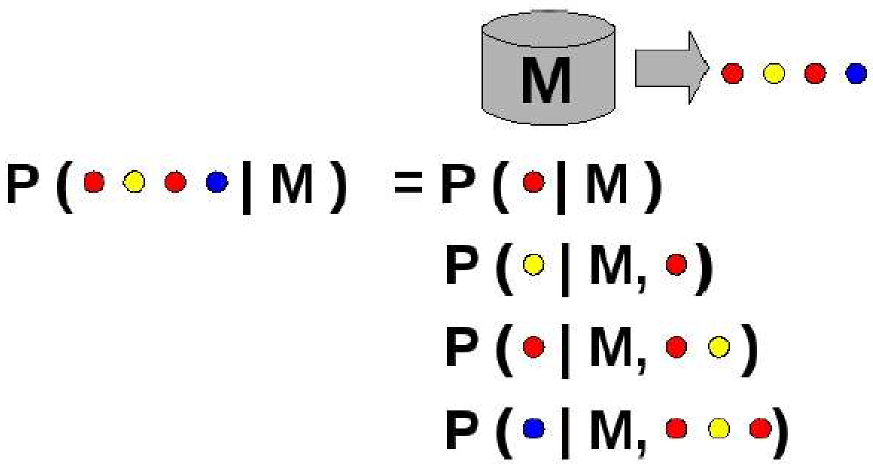
- What is the probability of producing the query from a document? $P(q|d)$
  - Referred to as the query-likelihood
- The query is generated as a representative of the "ideal" document
  - System's task is to estimate for each of the documents in the collection, which is most likely to be the "ideal" document

Generative Probabilistic Models

Generative Models - Language model

A statistical model for generating data

- Probability distribution over samples for a given language (document)
- M –> $t_1\ t_2\ t_3\ t_4$



Generative Probabilistic Models

Generative Probabilistic Models

- Assumptions:
  - The probability of a document being relevant is strongly correlated with the probability of a query given a document, i.e. $P(q|r)$ is correlated with $P(q|d)$
  - User has a reasonable idea of the terms that are likely to appear in the "ideal" document
  - Users query terms can distinguish the ideal document from the rest of the corpus

Language Models in IR

Statistical Language Models

- (Statistical) language models (LM) have been widely used for speech recognition and language (machine) translation for more than thirty years
- However, their use for information retrieval started only in 1998 [Ponte and Croft, SIGIR 1998]
    - Basically, a query is considered generated from an “ideal” document that satisfies the information need
    - The system’s job is then to estimate the likelihood of each document in the collection being the ideal document and rank them accordingly (in decreasing order)

Language Models in IR

Statistical Language Models

- What is LM Used for ?
  - Speech recognition
  - Spelling correction
  - Handwriting recognition
  - Optical character recognition
  - Machine translation
  - Document classification and routing
  - Information retrieval …

Language Models in IR

Language Models in IR

- Let us assume we point blindly, one at a time, at 3 words in a document
- What is the probability that I, by accident, pointed at the words “Master”, “computer”, and “Science”?
- Compute the probability, and use it to rank the documents.

Language Models in IR

Statistical Language Models

- A probabilistic mechanism for "generating" a piece of text
  - Define a distribution over all possible word sequences

$$T = t_1 t_2 \ldots t_L$$

$$P(T) = ?$$

  - Used LM to quantify the accept ability of a given word sequence

Language Models in IR

Query-Likelihood Language Models

- Criterion: Documents are ranked based on Bayes (decision) rule

$$P(D \mid Q) = \frac{P(Q \mid D) \cdot P(D)}{P(Q)}$$

  - $P(Q)$ is the same for all documents, and can be ignored
  - $P(D)$ might have to do with authority, length, genre, etc.
    - There is no general way to estimate it
    - Can be treated as uniform across all documents

Language Models in IR

Query-Likelihood Language Models

- Documents can therefore be ranked based on

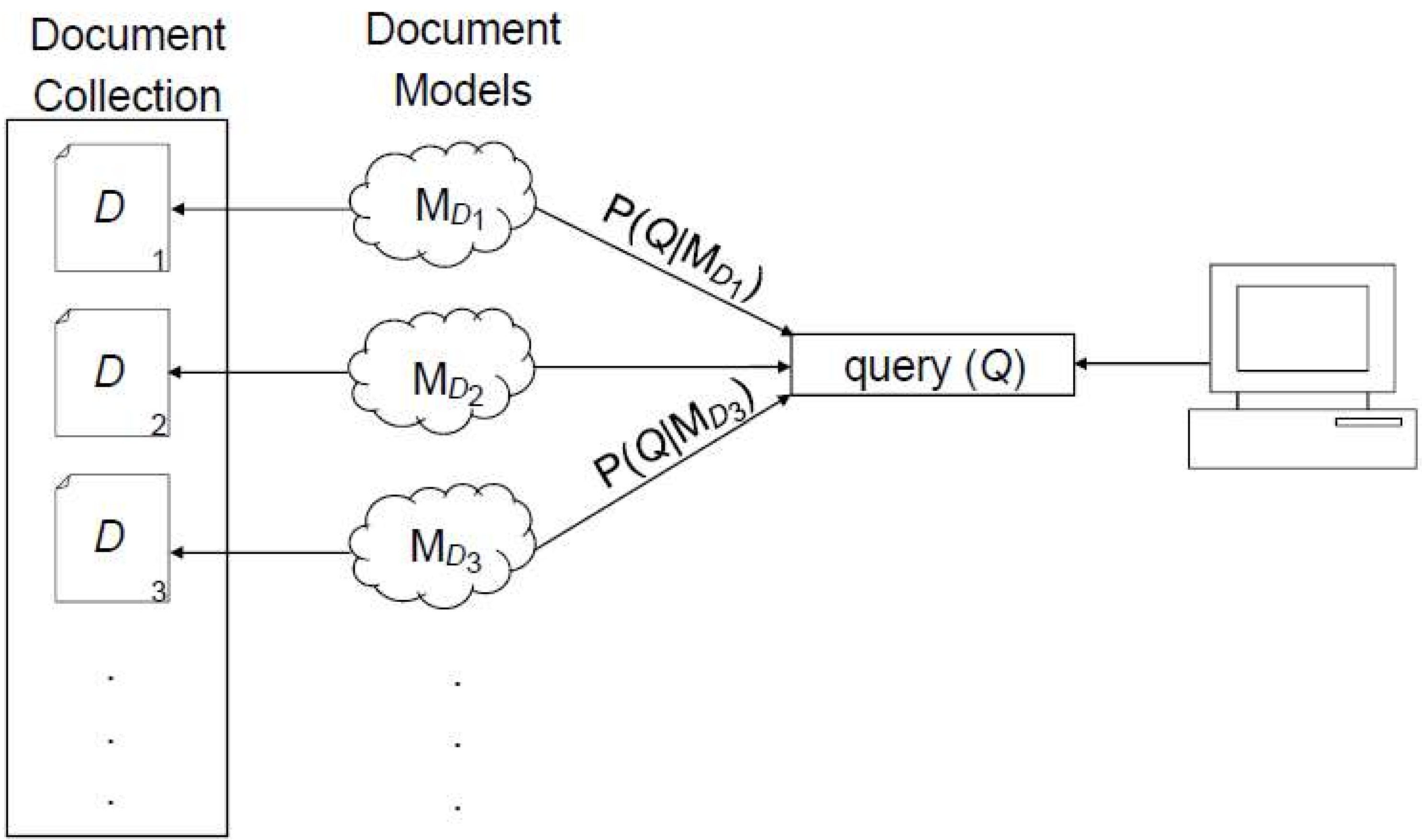
$$P(Q \mid D) \text{ or denoted as } P(Q \mid M_D)$$

Document models

  - The user has a prototype (ideal) document in mind, and generates a query based on words that appear in this document
  - A document D is treated as a model M_D to predict (generate) the query

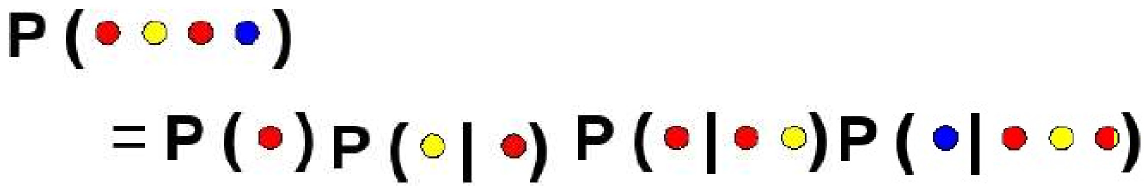
Language Models in IR

Schematic Depiction for Query-Likelihood Approach



Language Models in IR

Types of language models – urn metaphor



- Unigram Models

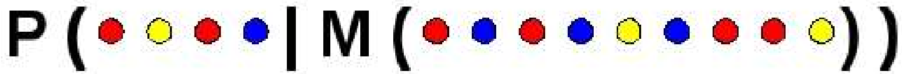
- Bigram Models

- There are others . . .
  - Most language-modelling work in IR has used unigram models
  - IR does not directly depend on the structure of sentences

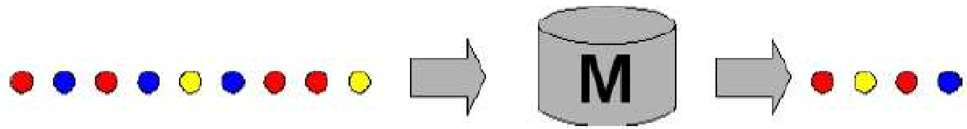
Language Models in IR

The fundamental problem

- Usually, we do not know the model M, but have a sample representative of that model



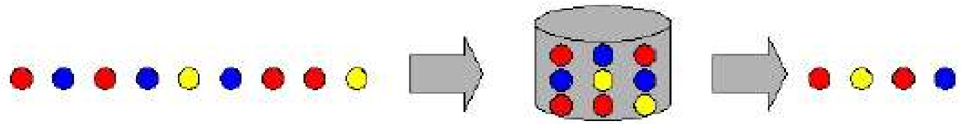
- First estimate a model from a sample
- Then compute the observation probability



Language Models in IR

Unigram Model - Example

(Urn metaphor)

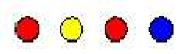
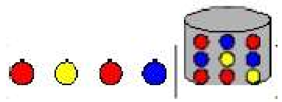
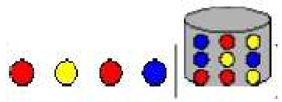
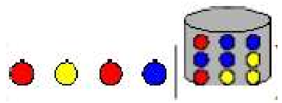
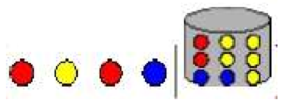


- P(● ○ ● ●) ~ P(●) P(○) P(●) P(●)

  = 4/9 * 2/9 * 4/9 * 3/9

Language Models in IR

Unigram Model - Example:
Ranking documents with unigram models

- Rank models (documents) by probability of generating the query
- Q: 
- *P* () =
- *P* () =
- *P* () =
- *P* () =

Language Models in IR

Build Document Models: *n*-grams

- Multiplication (Chain) rule

$$P(t_1 t_2 \ldots. t_L) = P(t_1)P(t_2 \mid t_1)P(t_3 \mid t_1 t_2) \cdots P(t_L \mid t_1 t_2 \ldots. t_{L-1})$$

- *n*-gram assumption
  - Unigram Models (Assume word independence)

$$P(t_1 t_2 \ldots. t_L) = P(t_1)P(t_2)P(t_3) \cdots P(t_L)$$

    - Each word occurs independently of the other words
    - The so-called “bag-of-words” model (e.g., how to distinguish “street market” from “market street)
  - Bigram Models

$$P(t_1 t_2 \ldots. t_L) = P(t_1)P(t_2 \mid t_1)P(t_3 \mid t_2) \cdots P(t_L \mid t_{L-1})$$

Language Models in IR

Unigram Model - Standard LM Approach

- Assume that query terms are drawn identically and independently from a document (unigram models)

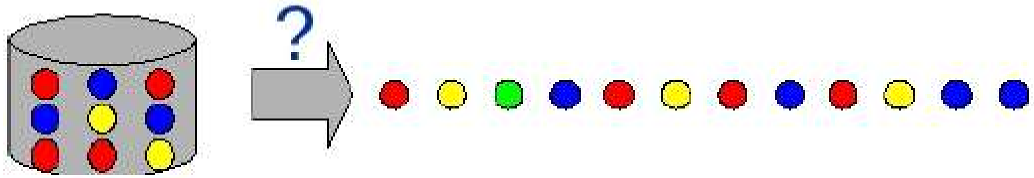
$$P(q \mid d) = \prod_{t \in q} P(t \mid d)^{n(t,q)}$$

(where $n(t, q)$ is the number of term t in query q)

- Maximum Likelihood Estimate of $P(t|d)$
  - Simply use the number of times the query term occurs in the document divided by the total number of term occurrences.
- Problem: Zero Probability (frequency) Problem

Unigram Model - The Zero-probability Problem

- Suppose some event not in sample (document)
  - Model will assign zero probability to that event
  - And to any set of events involving the unseen event
- Happens frequently with language
- It is incorrect to infer zero probabilities
  - Especially when dealing with incomplete samples



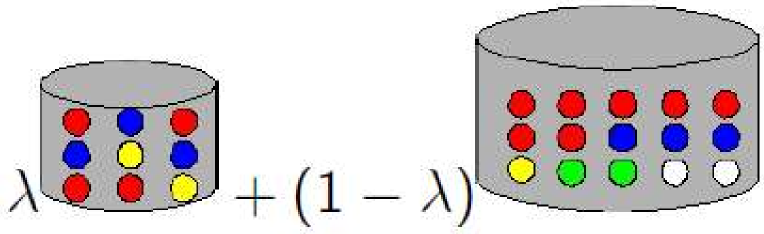
Language Models in IR

Unigram Model - Smoothing

- Standard approach is to use the probability of a term $P(t)$ to smooth the document model, thus

$$P(t \mid q_d) = \lambda P(t \mid d) + (1 - \lambda) P(t)$$

- Urn metaphor:



Language Models in IR

Unigram Model - Document Models

- Idea: shift part of probability mass to unseen events
- Interpolation with background (General English in our case)
  - Reflects expected frequency of events
  - Plays role of IDF in LM

$$P(t \mid q_d) = \lambda P(t \mid d) + \boxed{(1-\lambda)P(t)}$$

Background

Language Models in IR

Unigram Model - Estimating Document Models

- Basic Components
  - Probability of a term given a document (maximum likelihood estimate)

$$\boxed{P(t \mid d) = \frac{n(t,d)}{\sum_{t'} n(t',d)}}$$

Foreground model

  - Probability of a term given the collection

$$\boxed{P(t) = \frac{\sum_{d} n(t,d)}{\sum_{t'} \sum_{d'} n(t',d')}}$$

Background model

  - *n* (*t* , *d*) is the number of times term *t* occurs in document *d*

Unigram Model - Estimating Document Models

- Example of Smoothing methods
  - Laplace

$$P(t \mid q_d) = \frac{n(t,d) + \alpha}{\sum_{t'} n(t',d) + \alpha \mid T \mid}$$

$|T|$ is the number of term in the vocabulary

  - Jelinek-Mercer

$$\lambda \cdot P(t \mid d) + (1-\lambda) \cdot P(t)$$

  - Dirichlet

$$\frac{\mid d \mid}{\mid d \mid + \mu} \cdot P(t \mid d) + \frac{\mu}{\mid d \mid + \mu} \cdot P(t)$$

Language Models in IR

Unigram Model
Language Models - Implementation

We assume the following LM (Jelinek-Mercer smoothing):

$$P(q = t_1, t_2, \ldots, t_n \mid d) = \prod_{i=1}^{n} ((1-\lambda) \cdot P(t_i) + \lambda \cdot P(t_i \mid d))$$

It can be shown that the above leads to:

$$P(q = t_1, t_2, \ldots, t_n \mid d) \propto \sum_{i=1}^{n} \log(1 + \frac{\lambda \cdot P(t_i \mid d)}{(1-\lambda) \cdot P(t_i)})$$

for ranking purpose (again use log to obtain summation)

Language Models in IR

Unigram Model: example

$$P(q = t_1, t_2, \ldots, t_n \mid d) = \prod_{i=1}^{n} ((1-\lambda) \cdot P(t_i) + \lambda \cdot P(t_i \mid d))$$

Suppose the document collection contains two documents:

- *d*1: *Xyzzy reports a profit but revenue is down*
- *d*2: *Quorus narrows quarter loss but revenue decreases further*

The model will be unigram models from the documents and collection, mixed with $\lambda = 1/2$.

Suppose the query is *revenue down*. Then:

- $P(q|d1) = ?$
- $P(q|d2) = ?$

ranking *d*1 and *d*2

Language Models in IR

Unigram Model - Implementation as vector product

$$\text{Redefine}: P(t) = \frac{df(t)}{\sum_{t'} df(t')} \quad \text{and } P(t \mid d) = \frac{tf(t,d)}{\sum_{t'} tf(t',d)}$$

$$score(q,d) = \sum_{k(\text{Matching Text})} q_k d_k$$

$$q_k = tf(k,q)$$

$$d_k = \log \frac{tf(k,d) \cdot \sum_t df(t)}{df(k) \cdot \sum_t tf(t,d)} \frac{\lambda}{1-\lambda}$$

Language Models in IR

Unigram Model - Document Priors

- Remember $P(d|q) = P(q|d)P(d)/P(q)$
- $P(d)$ is typically assumed to be uniform so is usually ignored
- $P(d)$ provides an interesting avenue for encoding a priori knowledge about the document
  - Document length (longer doc $\rightarrow$ more relevant)
  - Average Word Length (bigger words $\rightarrow$ more relevant)
  - Time of publication (newer doc $\rightarrow$ more relevant)
  - Number of web links (more in links $\rightarrow$ more relevant)
  - PageRank (more popular $\rightarrow$ more relevant)

Language Models in IR

Unigram Model - “Language Modelling”

- Not just “English”
- But also, the language of
  - author
  - newspaper
  - text document
  - image
  - structure
  - …..

Summary LM

- Approach based on “probability” of relevance (like BIRM) but RSV is based on $P(q|d)$ and not $P(d|q, r)$
- Based on the probability that a term occurs in a sequence of terms.
- BIRM is based on the probability that term does or does not occur in a set of (retrieved) documents
- Relation to IDF:
  - Applying the logarithm after dividing by $P(t)$, t in q

$$\log\left(\frac{(1-\lambda)+\lambda\cdot P(t|d)}{P(t)}\right) \propto \mathrm{IDF}(t)$$

Language Models in IR

Summary LM

- A query *q* is viewed as a translation or distillation from a Document *d*
  - That is, the similarity measure is computed by estimating the probability that the query would have been generated as a translation of that document
  - Assumption of context-independence (the ability to handle the ambiguity of word senses is limited)

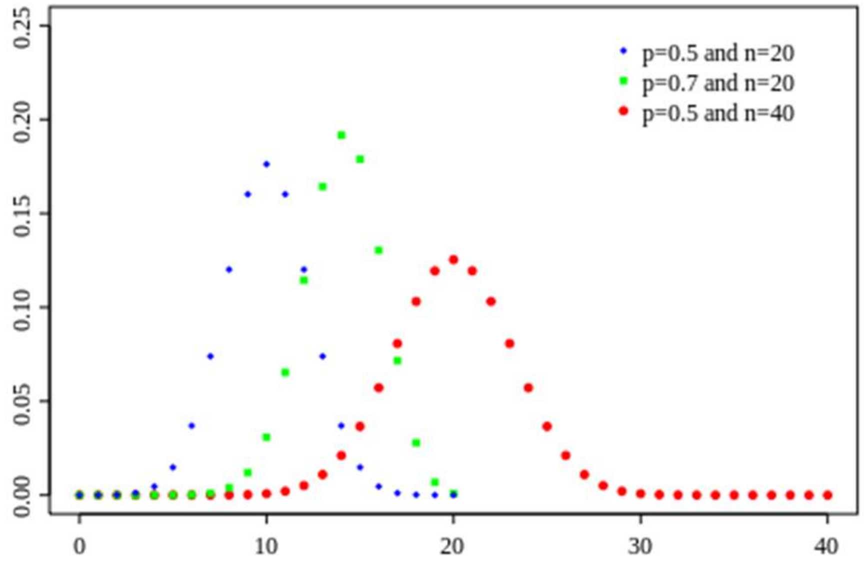
Summary - References

- Ponte and Croft, A language modelling approach to information retrieval, ACM-SIGIR 1998
- Hiemstra and de Vries, Relating the new language models of information retrieval to the traditional retrieval models, CTIT Technical Report, 2000
- Liu and Croft, Statistical Language Modelling for Information Retrieval, Annual Review of Information Science and Technology, 2003
- Zhai and Lafferty, A study of smoothing methods for language models applied to ad hoc information retrieval, ACM TOIS, 2004
- Larenko and Croft, Relevance Based Language Models, ACM SIGIR 2001
- Croft and Lafferty (eds), Language Modelling for Information Retrieval 2004

Binomial Distribution

Binomial distribution is the discrete probability distribution of the number of successes in a sequence of *n* independent *yes/no* experiments

- Each of which yields success with probability *p*.
- Such a success/failure experiment is also called a Bernoulli experiment or Bernoulli trial;
- When *n* = 1, the binomial distribution is a Bernoulli distribution.
- The binomial distribution is the basis for the popular binomial test of statistical significance.



Binomial Distribution

An example

$$P(n) = \binom{N}{n} \cdot p^n \cdot (1-p)^{N-n}$$

Imagine you go on a sailing trip on the East Coast. Every second day, there is a beautiful sunset, i.e. $p = 1/2$. You go sailing for a week ($N = 7$). What is your chance to have exactly three ($n = 3$) beautiful sunset?

Binomial Distribution

Just another example

$$P(n) = \binom{N}{n} \cdot p^n \cdot (1-p)^{N-n}$$

In your *company, 2 percent of the products delivered have a failure, i.e. p* = 2/100. Per day, you deliver 100 products (boats) (N = 100). In average, you expect 2 boats per day to be faulty. What is the probability that exactly one boat is faulty?

Divergence from Randomness (DFR)

- Basic idea: "The more the divergence of the within-document term frequency from its frequency within the collection, the more divergent from randomness the term is, meaning the more the information carried by the term in the document."

$$weight(t \mid d) \propto -\log P_M(t \in d \mid collection)$$

M stands for the type of model of the divergence from randomness employed to compute the probability.

In the next slide, the binomial distribution (*B*) is used as the model of the divergence from randomness.

See http://ir.dcs.gla.ac.uk/terrier/doc/dfr description.html

Divergence from Randomness (DFR)

Binomial Distribution as Randomness Model

- *TF* - Term frequency of term *t* (occurrence of *t*) in the collection
- *tf* - Term frequency of term *t* in the document *d*
- *p* - Probability to draw a document ($p = 1/N$, N is number of documents)

$$-\log P_B(t \in d \mid collection) = -\log \left(\binom{TF}{tf} \cdot p^{tf} \cdot (1-p)^{TF-tf} \right)$$

The probability that

- the event (that occurs with probability *p*) occurs *tf* times in *TF* trials
- a document occurs *tf* times in *TF* trials
- a sunny day (which occurs with $1/N$) occurs on *tf* days in a *TF* days holiday

Divergence from Randomness (DFR)

Binomial Distribution as Randomness Model

$$-\log P_B(t \in d \mid collection) = -\log \binom{TF}{tf} \cdot p^{tf} \cdot (1-p)^{TF-tf}$$

- If N is the number of documents, then $TF/N = \lambda$ is the average occurrence of term t.
- The above is minimum for $tf = \lambda$, meaning that term t has a random distribution.
- its distribution does not diverge from randomness
- and as such is not informative.

Divergence from Randomness (DFR)

Binomial Distribution as Randomness Model

Summary:

- Term-weights: measuring the divergence between a term distribution produced by a random process and the actual term distribution.
- Term-weight are inversely related to the probability of term-frequency within the document d obtained by a model of randomness
- There are many ways to choose the model of randomness, each of these provides a basic DFR model