Ranking

Claudio Lucchese

1 Ranking

We say that a document d satisfies a query q if all the terms in q are also contained in d. The task of discovering all the documents in a given collection that satisfy the query q is trivial using inverted files. Unfortunately, when it comes to large collection of documents, such as the Web, or whenever we must provide a fruitful user interaction, the set of satisfying documents is simply too large. We have to select a subset of documents which are likely to be more interesting for the user, or better, we must *rank* them from the most to the least interesting.

The definition itself of what is interesting and what is not is still an open problem endlessly investigated by computer scientist. Note that the interestingness of a document depends on many factors, one of them is the user: the same document may be interesting for a user and irrelevant for someone else.

Here, section after section we will introduce some "reasonable" interesting measure, without arguing on their goodness, but trying to understand how they've been used by web search engines.

We can discriminate between two main categories of interestingness measures, and these two techniques have been named *Content Analysis* and *Link Analysis*.

2 Content Analysis

The first approaches are aimed at finding a good similarity measure between a query q and a document d. The rationale is to find the documents which are more similar to q.

2.1 Boolean vectors.

Each document can be represented with a boolean vector of length l, where l is the number of terms in the whole collection of documents. Each term has a corresponding id in the interval $[0, l-1]$. A document vector is equal to 1 in its i-th component if and only if the i-th term is contained in the document, and 0 otherwise. In the very same way, we can represent any query q and thus use similarity or distance measures between vectors to estimate the interestingness of a document.

When using the Boolean vector model, the commonly used similarity measure is the inner product:

$$sim(q, d) = q \cdot t = \sum_t q_t \cdot d_t \tag{1}$$

question has each term the same importance ?

question what is the difference between a document where *computer science* occurs 10 time and a document where it appears just one ?

2.2 TF IDF

Vectors having only boolean values do not have enough expressive power. We need real valued vectors representing the weight (i.e. importance) of each term in representing a document. We will add a weight that answer the above two questions.

Intuitively, a term that occurs many times in a document is more representative of that document than a term occuring just once. *The weight of a term should be proportional to its frequency in the document.* This factor is called *term frequency.*

Some terms occur very often in the whole collecionts, such as article, adverbs, etc. Other terms occur seldomly. The former have no discriminative power, while the latter can indentify a precise topic and a precise subset of documents. Therefore *the weight of a term should be inversely proportional to the number of documents containing that term.* This factor is called *inverse document frequency.*

The product of the two quantities gives the $tf \cdot ifd$ coefficient of a term, which can be used as a component of the vector representing a given document.

question is length important to evaluate a document ?

2.3 Cosine Distance

In the vector space model, each document and each query can be represented by a vector in a multidimensional space. Notice that each vector has a different length, but length should not be important in our contest. In fact, a longer document is not more important that a shorter one. On the other hand, long documents are statistically more probable to match a query, and therefore may get a good score with no reason. The distance between a query and a document, may be measured by the cosine of the angle of the two corresponding vector.

$$cos(q, d) = \frac{q \cdot t}{\|q\| \|t\|} \tag{2}$$

Normalization is usually tought has the correct way to deal with with different length vectors, and, actually, the cosine is an implicit normalization of the inner product.

The first search engines (e.g. altavista) where based mostly on this kind of content analysis.

question Is this working ?

There are some weak points in this approach. Some are due to characteristics which are peculiar of the vector space model, while some are due the web search users.

One main issue is the dimensionality curse. As the dimensionality of the problem increases, i.e. the number of terms, the concept of similarity loses its meaning.

Secondly, the average query length is 2 or 3, therefore it is not enough expressive.

Finally, natural language has ambiguities.

SVD

3 Link Analysis

Use structure of the web graph to enhance search.

question Why should we use structure ??

A link is a vote from a user !!! this is the best way to evaluate user tastes. The goal of a search engine is not to satisfy my needs, but to satisfy average needs. Think about web-logs.

try to search university on google and altavista.

3.1 In-Degree

article `Measuring the Web.`

The importance **visibility** of a page is proportional to the number of pages pointing to it, i.e. the number of backlinks or incoming links:

$$R(p) = |In(p)| \tag{3}$$

question Any problem with hierarchies ?

article `The quest for correct information on Web: Hyper search engines.`

The information of a page is related to the textual content plus hte hyper content.

Marchiori introduced an enrichment technique with a fading factor, which by the way does not remove the problem.

3.2 Page Rank

article `The anatomy of a large scale hypertextual Web search engine.` **article** `The PageRank Citation Ranking: Bringing order to the Web.` **article** `Inside Page Rank.`

trial 1: Trying to improve In-Degree...

page rank 1: proportional to backlinks importance

$$I(p) = \sum_{q \in In(p)} I(q) \tag{4}$$

page rank 2: inversely proportional to out links

$$I(p) = \sum_{q \in In(p)} \frac{I(q)}{|Out(q)|} \tag{5}$$

question Does it converge ?
dangling nodes ...
trial 2: Random Walk...
The higher the probability to reach a page by randomly browsing the Web, the more important is the page.

Each link is equally probable to be selected. Therefore the probability of going from a page q to a neighbor page p is:

$$P(p) = \frac{1}{|Out(q)|} \tag{6}$$

The probability of reaching p increases if it is highly probable to reach q:

$$P(p) = \frac{P(q)}{|Out(q)|} \tag{7}$$

The probability of reaching p increases if it has many pages pointing to it:

$$P(p) = \sum_{q \in In(p)} \frac{P(q)}{|Out(q)|} \tag{8}$$

question Does it converge ?
question Does it model a web navigation ?
Add some randomness jumping from one node to another.

$$P(p) = \alpha \sum_{q \in In(p)} \frac{P(q)}{|Out(q)|} + (1 - \alpha)V(p) \tag{9}$$

question Does it converge ?
trial 3: Markov Process...
Let's use some well know theory. The Out-Degree of a page q tells us the probability of reaching a page p in its neighborhood. For each page p we can define the probability of reaching in one step any other page in the web. This set of probabilities defines a *status transition matrix* of a random process, i.e. a Markov process.

definition A Markov process, is a stochastic process satisfying the Markov property.
definition stochastic = random
definition Markov property = the next state of the system depends only on the current state, and not on the past states.

Given the transition matrix, the probability of reaching p in the next time step is:

$$P_{t+1}(p) = \sum_{q \in Web} P_t(q) P(q \to p) \tag{10}$$

question Does it converge ?

The transformation converges if $P_{t+1}(p) = P_t(p)$ for every p. At the point of convergence $P(p)$ is the probability of being at page p, and the probability distribution over all pages is called stationary distribution π.

stationary A Markov chain is stationery if it is time independent.

convergence The Markov chain converges, i.e. $\lim_{t\to+\infty} p_t(p) = \pi(p)$ and the point of convergence is unique if it is irreducible and aperiodic.

irreducibility It is irreducible id there is non zero probability to get to state p from state q for every p and q.

aperiodic A state p has period k if we can come back to state p in the future only after a multiple of k steps. The period of a state is the greatest common divisor of all the periods of p. The chain is aperiodic if all of its state are aperiodic , i.e. have period 1.

question is the web Irreducible ? is the web aperiodic ?

No. We add jumping factor allowing to move at random from one page to another.

$$P_{t+1}(p) = \alpha \sum_{q \in Web} P_t(q)P(q \to p) + (1-\alpha)V(p) \tag{11}$$

If we start from page p and apply the transformation P we get a probability for each other page in the web, i.e. a vector of probabilities x. The initial position can be thought as a vector x_0, and the transition simply consists in multiplying P times x_0. To reach the converging state we must apply the transformation several times, and this will lead to the stationary distribution regardless x_0:

$$P(P(P(P(...(P \times x_0)...)))) \to \pi \tag{12}$$

This is also called power method:

$$P^n \times x_0 \to \pi \tag{13}$$

and, by recalling some linear algebra, we know that the power method finds the eigenvector corresponding to the maximum eigenvalue.

article `The Second Eigenvalue of the Google Matrix.`

question How much fast does it converge ??

The convergence speed of the power method is $|\lambda_2/\lambda_1|$. It can be show that (a) since P is stochastic then $\lambda_1 = 1$ and that (b) given the web structure $\lambda_2| = \alpha$. Therefore the convergence rate of the power method applied to the web graph is equal to α, which is typically set to .85.

question what about conditioning of the problem ??

The larger $\lambda_1 - \lambda_2 = 1 - \alpha$ the more stable is the stationary distribution.

question what is $V(p)$??

$V(p)$ is called personalization vector, and it allow to increase the importance of a web page, diregarding the web graph itself.

important PageRank is query independent.

3.3 HITS

article `Authoritative Sources in a Hyperlinked Environment`

Hiper text induced topic selection

Select a subset of interesting pages using traditional first generation search engines: *the root set*.

Expand the set with its neighborhood *the base set*.

Try to find Hubs and Authorities. Each page p has an initial hubness value $H(p) = 1$. Then update the authority value of each node $A(p)$ summing the hubness values from backward link. Repeat similarity to get new hubness values. The process converges to a set of hubness and authority values.

Autority vector converges to the eigenvector of $A^T A$, while hubs vector converges to the eigenvector of AA^T.

Non principal eigenvectors may resolve synonym.

(PCA ??)

question how to find similar pages ...

Find authorities using backlinks of the first page as the root set.

Important HITS ranking is query dependent.

problems topic drift

3.4 SALSA

SALSA: the stochastic approach for link-structure analysis

Random walk in the hub graph.

Statistically proven to behave better than hits in precence of TKC. Few hubs that link every authority may have greater rank than good authorities in a larger subset of pages.

3.5 What is best?

Mixed approach.

3.6 SPAM

TF x IDF

Page splitting (PageRank)

Topic Drift (Topic Drift)

3.7 Other applications of PageRank.

Focused crawling

Different indexes for different ranks