Vector Space Model

Yufei Tao

KAIST

March 5, 2013

In this lecture, we will study a problem that is (very) fundamental in information retrieval, and must be tackled by all search engines.

Let S be a set of data documents D_1, D_2, ..., D_t, where $t = |S|$. Let Q be a query document. Each (data/query) document is a sequence of terms. We want to rank the data documents in descending order of their relevance to Q. Namely, the most relevant document should be placed first, the second most relevant placed next, and so on.

We want to design a way for a computer to do it for us. Let us refer to the problem as the relevancy problem.

Example

Consider that each document $D_i \in S$ $(1 \le i \le t)$ is a news article. Now, given a query article Q about basketballs, we would like to find the documents that are similar to Q. In other words, ideally, we should rank the articles about basketballs before those about, say, soccer, and sports articles before non-sports articles.

How is this related to web search?

- Let each D_i be a webpage in the Internet. Let Q, on the other hand, be the (super-short) document that is the sequence of terms a user inputs into Google's search box. By ranking the data documents, we are essentially listing them in descending order of how relevant they are to Q. This is the order by which we will present the documents to the user.

Think

Some search engines provide an option "see more results like this". How can we implement this functionality if we can solve the relevancy problem?

WARNING

The relevancy problem does not have an unambiguous correct answer (relevancy judgment is a subjective matter).

Thus, the quality of a method can often be evaluated only empirically, i.e., to see how well it actually works in practice. There are a few methods that have been empirically shown to be quite effective. Among them, we will discuss an approached based on the vector space model because it is acknowledged by many scientists as being highly effective. It is also the approach taken by today's search engines.

The vector space model is based on the following rationale:

> **Rationale**
>
> If two documents have similar terms, then they are relevant.

You can easily challenge the above rationale by giving many counter-examples. Even though you are absolutely right, the chance is that, every solution would fail when confronted with a carefully crafted counter-example. Indeed, it is not our goal to find a perfect approach that guarantees a correct answer in all scenarios – remember that even correctness is hard to define. We will be satisfied if our heuristic approach works sufficiently well empirically, by which standard the above rationale is really not a bad one.

Let *DICT* be a dictionary which contains all the terms we want to consider in evaluating text relevancy.

Think

DICT should not include all the English words. Why?

Let $d = |DICT|$, namely, the number of terms in *DICT*. Let us denote those terms as $w_1, w_2, ..., w_d$, respectively.

Remark

In practice, words like "study", "studies", "studying" and "studied" are all regarded the same. They can be all converted to the same form by a process commonly known as stemming.

Vector Space Model

We will convert each document $D_i \in S$ $(1 \leq i \leq t)$ into a point p_i of dimensionality d.

More specifically, let the coordinates of p_i be $p_i[1], p_i[2], ..., p_i[d]$, respectively. Then, for each $j \in [1, d]$, $p_i[j]$ gives the relevance of term w_j to document D_i.

Next, we will clarify how $p_i[j]$ is computed.

Computation of $p_i[j]$

Heuristic 1

If w_j appears more often in D_i, then w_j is more relevant to D_i.

Definition (Term Frequency)

The term frequency (TF) f_{ij} of w_j in D_i equals the number of occurrences of w_j in D_i.

Computation of $p_i[j]$

Heuristic 2

If w_j appears in many documents, then it has low relevance to all of them.

Definition (Inverse Document Frequency)

The inverse document frequency (IDF) idf_j of w_j equals

$$idf_j = \log_2 \frac{|S|}{\lambda}$$

where λ is the number of documents in S containing w_j.

Computation of $p_i[j]$

Now we can compute $p_i[j]$ as:

$$p_i[j] \quad = \quad \log_2(1 + f_{ij}) \cdot idf_j$$

The above equation is well known as the tf-idf formula.

Think

How does this formula reflect Heuristics 1 and 2? Also, why the logs? Why the $+1$?

At this point, we have represented each document $D_i \in S$ $(1 \le i \le t)$ as a d-dimensional point p_i, where $d = |S|$. Let P be the set $\{p_1, ..., p_t\}$.

In a similar manner, we can also convert the query document to a d-dimensional point $q = (q[1], ..., q[d])$, where

$$q[j] \quad = \quad \log_2(1 + f_j) \cdot idf_j$$

where f_j is the term frequency of term w_j (i.e., the j-term in the dictionary $DICT$) in Q.

We will evaluate the relevance of D_i to Q, denoted as $score(D_i, Q)$, by looking at only p_i and q.

Think

It is a bad idea to set $score(D_i, Q)$ simply to the Euclidean distance (i.e., the "line segment distance") between p_i and q. Why?

Cosine Metric

Let us view p_i as a vector: from the origin to p_i. Similarly, let us also view q as a vector. The norms of p_i and q are calculated as:

$$
\begin{aligned}
|p_i| &= \sqrt{\sum_{j=1}^{d} p_i[j]^2} \\
|q| &= \sqrt{\sum_{j=1}^{d} q[j]^2}
\end{aligned}
$$

Their dot product is:

$$p_i \cdot q = \sum_{j=1}^{d} (p_i[j] \cdot q[j])$$

Cosine Metric

Let θ be the angle between the vectors p_i and q. We know:

$$\cos(\theta) = \frac{p_i \cdot q}{|p_i| \cdot |q|}$$

We then take the above to be $score(D_i, Q)$, namely:

$$score(D_i, Q) = \cos(\theta)$$

Hence, the larger the cosine value is, the more relevant D_i is to q. The above formula is well known as the cosine metric.

Putting Everything Together

We have introduced a full set of procedures required to tackle the relevancy problem. In a nutshell, the procedures are:

1. Convert each document $D_i \in S$ to a point p_i.
2. Convert the query document to a point q.
3. Calculate $score(D_i, Q)$ for all $i \in [1, |S|]$.
4. Sort the documents of S in descending order of their scores.

Think

What are the disadvantages of the vector space model?