

# 63

## Information Retrieval and Web Search

---

Amy N. Langville

*The College of Charleston*

Carl D. Meyer

*North Carolina State University*

63.1	The Traditional Vector Space Method .....	63-1
63.2	Latent Semantic Indexing .....	63-3
63.3	Nonnegative Matrix Factorizations .....	63-5
63.4	Web Search .....	63-8
63.5	Google's PageRank .....	63-10
	References .....	63-14

Information retrieval is the process of searching within a document collection for information most relevant to a user's query. However, the type of document collection significantly affects the methods and algorithms used to process queries. In this chapter, we distinguish between two types of document collections: traditional and Web collections. Traditional information retrieval is search within small, controlled, nonlinked collections (e.g., a collection of medical or legal documents), whereas Web information retrieval is search within the world's largest and linked document collection. In spite of the proliferation of the Web, more traditional nonlinked collections still exist, and there is still a place for the older methods of information retrieval.

### 63.1 The Traditional Vector Space Method

---

Today most search systems that deal with traditional document collections use some form of the vector space method [SB83] developed by Gerard Salton in the early 1960s. Salton's method transforms textual data into numeric vectors and matrices and employs matrix analysis techniques to discover key features and connections in the document collection.

#### Definitions:

For a given collection of documents and for a dictionary of  $m$  terms, document  $i$  is represented by an  $m \times 1$  **document vector**  $\mathbf{d}_i$  whose  $j$ th element is the number of times term  $j$  appears in document  $i$ .

The **term-by-document matrix** is the  $m \times n$  matrix

$$A = [\mathbf{d}_1 \mathbf{d}_2 \cdots \mathbf{d}_n]$$

whose columns are the document vectors.

**Recall** is a measure of performance that is defined to be

$$0 \leq \text{Recall} = \frac{\# \text{ relevant docs retrieved}}{\# \text{ relevant docs in collection}} \leq 1.$$

**Precision** is another measure of performance defined to be

$$0 \leq \text{Precision} = \frac{\# \text{ relevant docs retrieved}}{\# \text{ docs retrieved}} \leq 1.$$

Query processing is the act of retrieving documents from the collection that are most related to a user's query, and the **query vector**  $\mathbf{q}_{m \times 1}$  is the binary vector defined by

$$q_i = \begin{cases} 1 & \text{if term } i \text{ is present in the user's query,} \\ 0 & \text{otherwise.} \end{cases}$$

The **relevance** of document  $i$  to a query  $\mathbf{q}$  is defined to be

$$\delta_i = \cos \theta_i = \mathbf{q}^T \mathbf{d}_i / \|\mathbf{q}\|_2 \|\mathbf{d}_i\|_2.$$

For a selected tolerance  $\tau$ , the **retrieved documents** that are returned to the user are the documents for which  $\delta_i > \tau$ .

#### Facts:

1. The term-by-document matrix  $A$  is sparse and nonnegative, but otherwise unstructured.
2. [BB05] In practice, weighting schemes other than raw frequency counts are used to construct the term-by-document matrix because weighted frequencies can improve performance.
3. [BB05] Query weighting may also be implemented in practice.
4. The tolerance  $\tau$  is usually tuned to the specific nature of the underlying document collection.
5. Tuning can be accomplished with the technique of relevance feedback, which uses a revised query vector such as  $\tilde{\mathbf{q}} = \delta_1 \mathbf{d}_1 + \delta_3 \mathbf{d}_3 + \delta_7 \mathbf{d}_7$ , where  $\mathbf{d}_1$ ,  $\mathbf{d}_3$ , and  $\mathbf{d}_7$  are the documents the user judges most relevant to a given query  $\mathbf{q}$ .
6. When the columns of  $A$  and  $\mathbf{q}$  are normalized, as they usually are, the vector  $\boldsymbol{\delta}^T = \mathbf{q}^T A$  provides the complete picture of how well each document in the collection matches the query.
7. The vector space model is efficient because  $A$  is usually very sparse, and  $\mathbf{q}^T A$  can be executed in parallel, if necessary.
8. [BB05] Because of linguistic issues such as polysomes and synonyms, the vector space model provides only decent performance on query processing tasks.
9. The underlying basis for the vector space model is the standard basis  $\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_m$ , and the orthogonality of this basis can impose an unrealistic independence among terms.
10. The vector space model is a good starting place, but variations have been developed that provide better performance.

#### Examples:

1. Consider a collection of seven documents and nine terms (taken from [BB05]). Terms not in the system's index are ignored. Suppose further that only the titles of each document are used for indexing. The indexed terms and titles of documents are shown below.



























