Lesson 6

Inverted Indexing for Text Retrieval

The three in one structure of the Unified Field

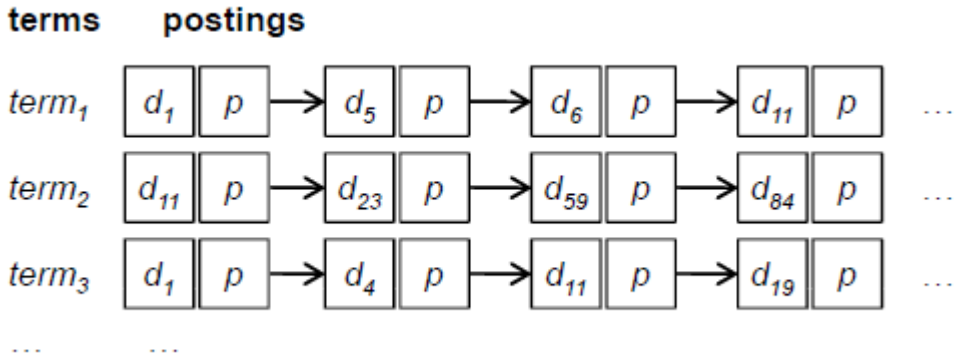
WHOLENESS OF THE LESSON

Nearly all retrieval engines for full-text search today rely on a data structure called an inverted index, which given a term provides access to the list of documents that contain the term.

Rishi (knower), Devata (process of knowing) and Chhandas (known) are the three basic qualities that structure natural law.

INVERTED INDEXES

- An inverted index is a collection of **postings lists**.
- One posting list for each “term”.



- A **posting** consists of docId and **payload**.
- The simplest payload is “nothing”.
- In our example, payload is **term frequency**.
- Term frequency is the number of times the term appear in the doc.

INVERTED INDEXES

- Postings are kept in sorted order. In our example, they are kept in sorted by docId.
- The document ids have no inherent semantic meaning, although assignment of numeric ids to documents need not be arbitrary.
- Pages from the same domain may be consecutively numbered.
- Alternatively, pages that are higher in quality (based on PageRank values) might be assigned smaller numeric values.
- An auxiliary data structure is necessary to maintain the mapping from integer document ids to some other more meaningful handle, such as a URL.

INVERTED INDEXES

- Given a query, retrieval involves:
  - fetching postings lists associated with query terms
  - In the simplest case, boolean retrieval involves set operations (union for boolean OR and intersection for boolean AND) on postings lists.

INVERTED INDEXING

```
class Mapper
    method Initialize
        H ← new AssociativeArray
    method Map(docid n, doc d)
        for all term t in doc d do
            H{t} ← H{t} + 1
    method close
        for all term t in H do
            Emit((t, n), H{t})
```

INVERTED INDEXING

```
class Reducer
    method Initialize
        tprev ← Ø, P ← new PostingsList
    method Reduce((t, n), [f])
        if (t ≠ tprev && tprev ≠ Ø) then
            Emit(term tprev, postings P),    P.Reset()
        P.Add(new Pair(n, f)), tprev ←  t
    method Close
        Emit(term tprev; postings P)
```

Main Point 1

An inverted index consists of postings lists, one associated with each term that appears in the collection. Thus an inverted index is a linked list of postings. *A graphical technique employed by Vedic science is the unified field chart which gives a holistic overview of a discipline and links all knowledge with the Self.*