Information Retrieval
Topic-Distributed Indexing
Lecture-19

Prepared By

Dr. Rasmita Rautray & Dr. Rasmita Dash
Associate Professor
Dept. of CSE

Content

- Distributed Indexing
  - Parallel tasks
  - Parsers
  - Inverters
  - Mapreduce

Index construction

- **Blocked Sort-Based Indexing**
  - implement a term to term-ID mapping
  - very slow index construction
- **Single-pass in-memory indexing**
  - Generate separate dictionaries for each block – no need to maintain term-termID mapping
  - Don’t sort

Distributed Indexing

- Collections are often so large that we cannot perform index construction efficiently on a single machine.
- This is particularly true of the World Wide Web for which we need large computer *clusters* to construct any reasonably sized web index.
- Web search engines use *distributed indexing algorithms* for index construction.
- The result of the construction process is a distributed index that is partitioned across several machines – either according to term or according to document.

Distributed indexing

- Maintain a master machine directing the indexing job – considered “safe”
- Break up indexing into sets of parallel tasks
- Master machine assigns each task to an idle machine from a pool.

Parallel tasks

- We will define two sets of parallel tasks and deploy two types of machines to solve them:
  - Parsers
  - Inverters
- Break the input document collection into splits (corresponding to blocks in BSBI/SPIMI)
- Each split is a subset of documents.

Parsers

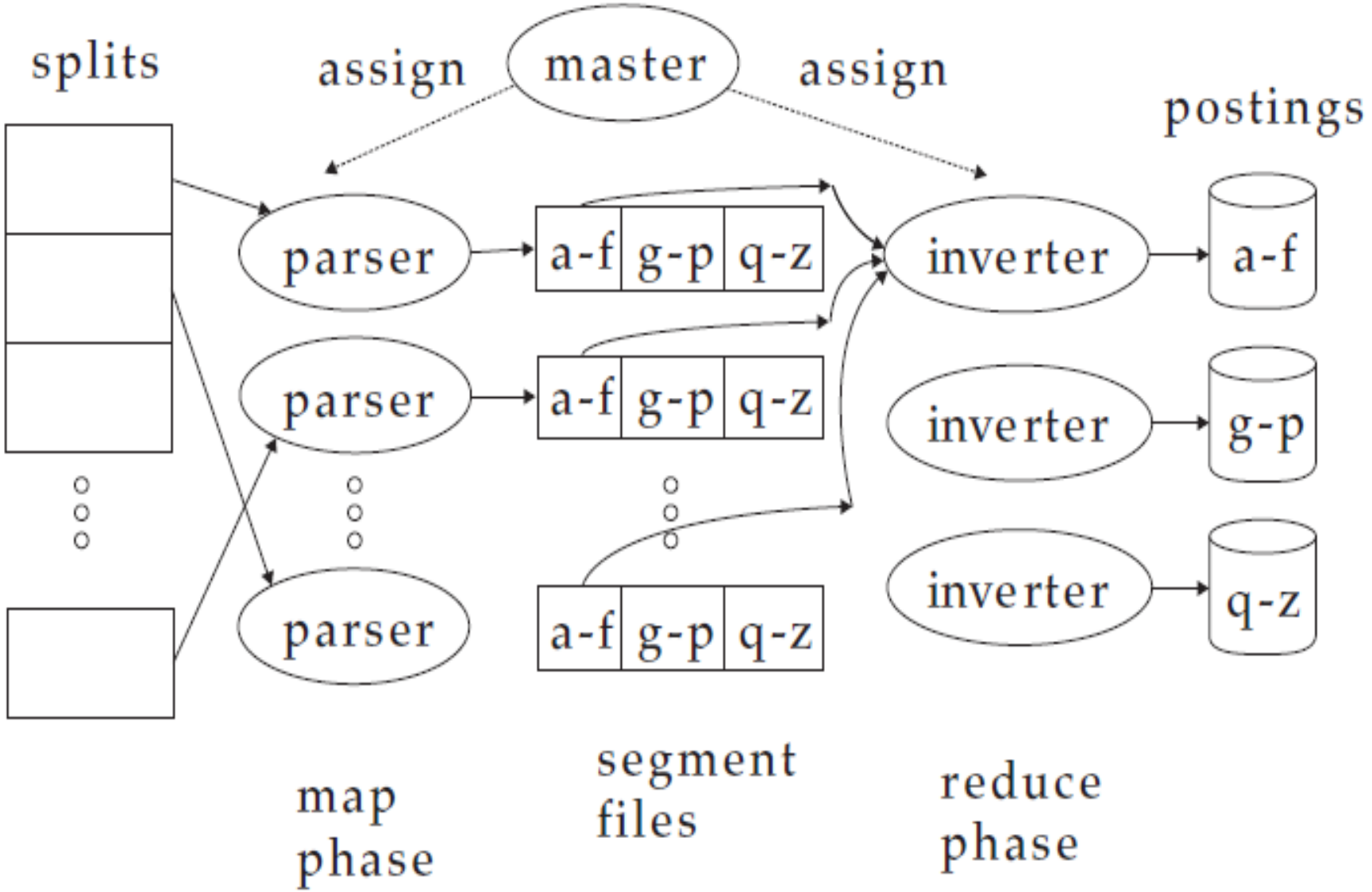
- Master assigns a split to an idle parser machine.
- Parser reads a document at a time and emits (term,docID)-pairs.
- Parser writes pairs into j term-partitions.
- Each for a range of terms' first letters

  e.g., a-f, g-p, q-z (here: j = 3)

Inverters

- An inverter collects all (term,docID) pairs (= postings) for one term-partition (e.g., for a-f)
- Sorts and writes to postings lists

Data flow


splits
assign
master
assign
postings
parser
parser
parser
a-f g-p q-z
a-f g-p q-z
a-f g-p q-z
inverter
inverter
inverter
a-f
g-p
q-z
map phase
segment files
reduce phase

MapReduce

- MapReduce is a robust and conceptually simple framework for distributed computing .
- Index construction was just one phase.
- Another phase: transform term-partitioned into document-partitioned index.

Thank You