Information Retrieval
K-gram Index
Lecture-11

Prepared By

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

Content

- Limitation of Permuterm index
- K-gram indexes
- Processing wildcard queries

Limitation of Permuterm index

- can lead to a considerable blowup from the number of rotations per term; for a dictionary of English terms
- can represent an almost ten-fold space increase.
- Number of rotation is high.

We now present a second technique, known as the K-gram index, for processing wildcard queries.

K-gram indexes

- More space-efficient than permuterm index
- Enumerate all character k-grams (sequence of k characters)
- occurring in a term
- 2-grams are called bigrams.
- Example: from April is the cruelest month we get the bigrams:
- a, ap, pr, ri, il, l, $i, is s$, t, th, he, e, $c, cr, ru, ue, el, le, es, s,t t$, m, mo, on, nt, h.

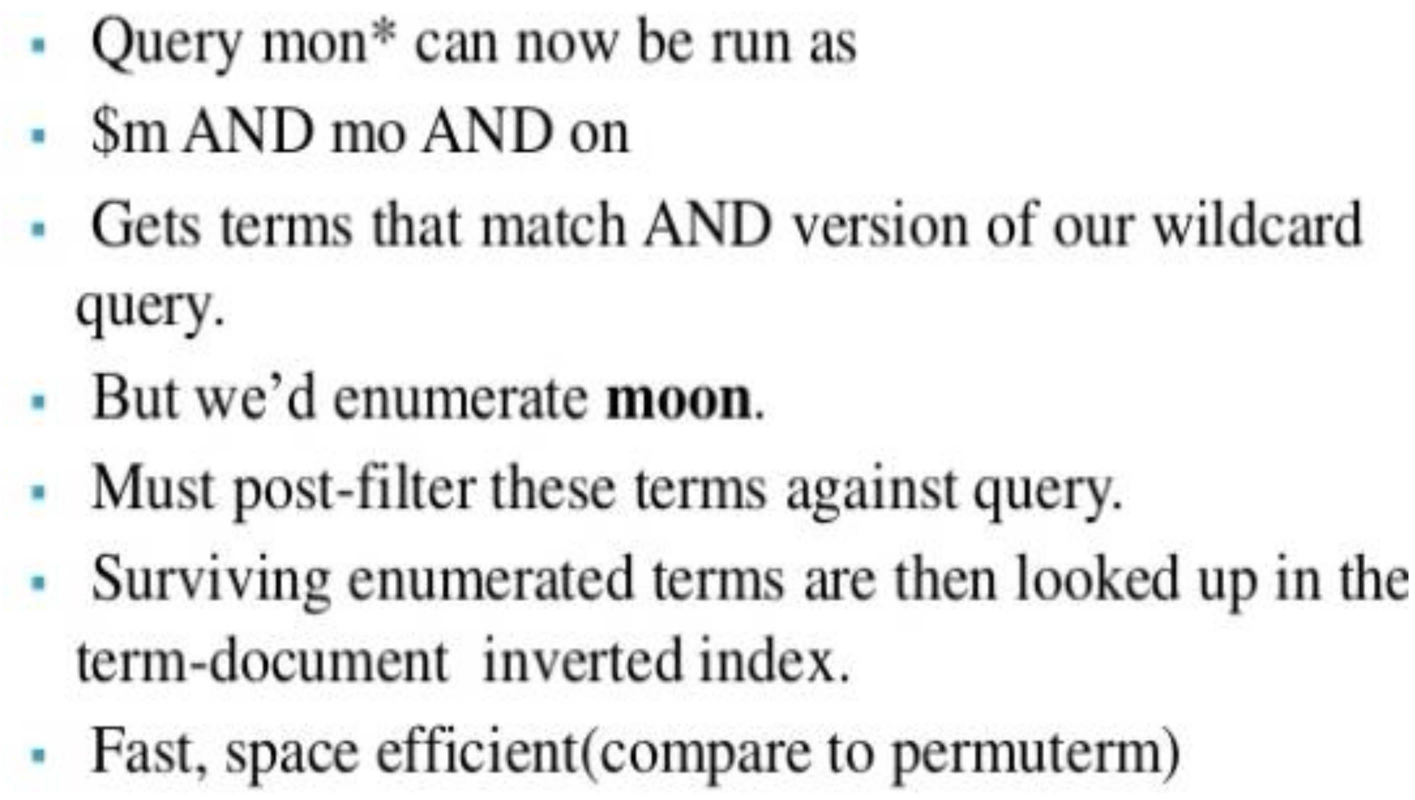
Contd..

- $ is a special word boundary symbol, as before.
- Maintain an inverted index from bigrams to the terms that contain the bigram

Contd..

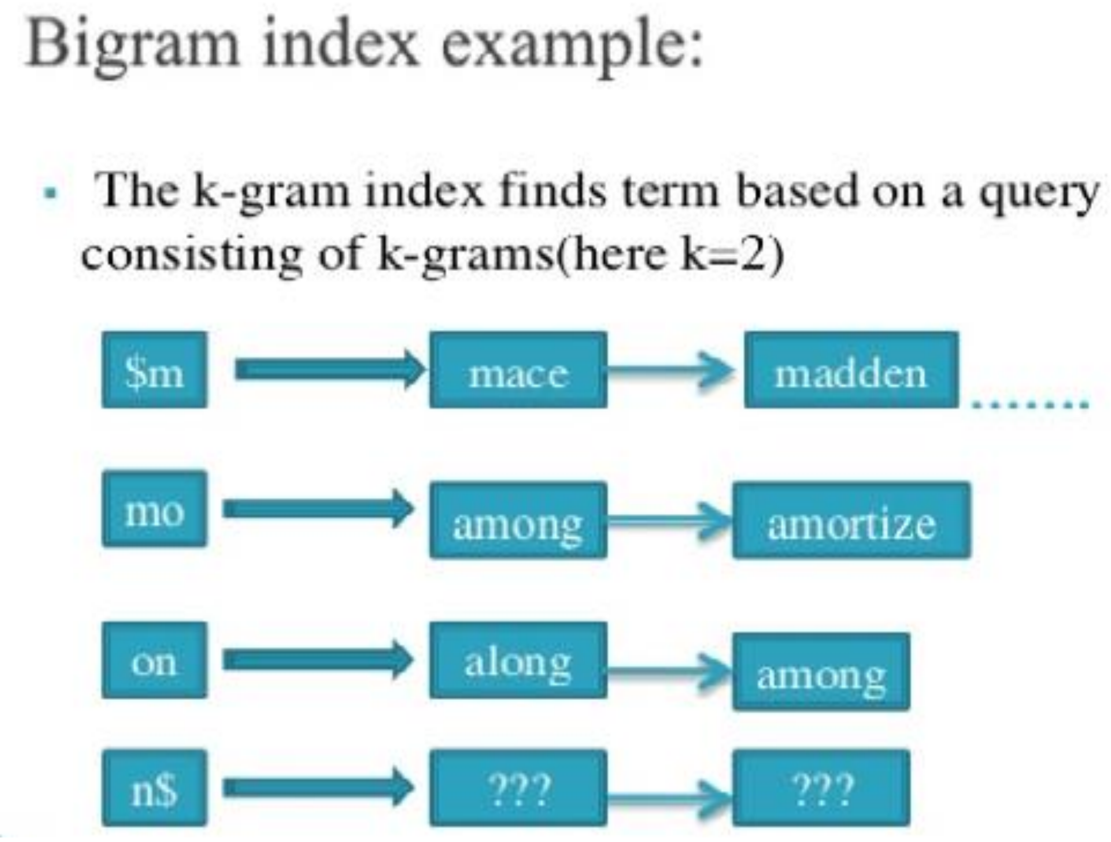
- Note that we now have two different types of inverted indexes
- The term-document inverted index for finding documents
- based on a query consisting of terms
- The k-gram index for finding terms based on a query
- consisting of k-grams

Processing wild cards



- Query mon* can now be run as
- $m AND mo AND on
- Gets terms that match AND version of our wildcard query.
- But we'd enumerate **moon**.
- Must post-filter these terms against query.
- Surviving enumerated terms are then looked up in the term-document inverted index.
- Fast, space efficient(compare to permuterm)

For the term moon



Bigram index example:

- The k-gram index finds term based on a query consisting of k-grams(here k=2)

$m → mace → madden ....

mo → among → amortize

on → along → among

n$ → ??? → ???

Thank You