Accessing WARC files via SQL

Sebastian Nagel

sebastian@commoncrawl.org

IIPC Web Archiving Conference, 6–7 June 2019, Zagreb, Croatia

About Common Crawl

- we're a non-profit that makes web data accessible to programmers and data scientists
- for natural language processing, web science, semantic web, internet security research, ...
- hosted as Open Data set on Amazon Web Services
- 4 Petabytes of data are provided in total (May 2019)

Common Crawl data usage

- data usage per month:
  - 3 – 10 Petabytes of data requested (“downloaded”)
  - 1.5 – 2.5 billion requests – download WARC file, CDX index lookup, fetch single WARC record, etc.
- data formats and amount of data released monthly (TiB):

—	ARC (2008 – 2012)
50	WARC
20	WAT
10	WET
0.3	URL index (CDX)

WARC and CDX – What's the matter?

- simple questions should be easy to answer – however …
- e.g., frequency of content languages
  - WARC metadata records: read 50 Terabytes, decompress, (partially) parse WARC records
  - CDX: read 300 GB, encompasses, parse JSON
- WARC, WAT, WET, CDX are record-oriented or "row-oriented" formats
- need to read the entire record (in the worst case, the entire file) to read a single value
- CDX index: only efficient for look ups by URL or domain name

The columnar index

- column-oriented data formats allow to access the values of a single column with no or little overhead
- (SQL) aggregations and analytics on selected columns only
- frequency of content languages: read only 500 MB

```
SELECT COUNT(*) AS frequency,
       content_languages
FROM "ccindex"
WHERE crawl = 'CC-MAIN-2019-22'
  AND subset = 'warc'
GROUP BY  content_languages
ORDER BY  count DESC;
```

frequency	languages
1054153645	eng
135485424	rus
128253368	zho
91175718	deu
76808733	*(unknown)*
75964600	spa
69548865	fra
59563812	jpn
58518088	jpn,eng

The vertical use case – Accessing WARC files via SQL

- "vertical" access: pick only WARC records you're interested in
  - records of a single domain, one content language, matching a URL pattern, …
  - … or a combination of it
- use a SQL query to pick records and process the records using Spark and Python or Java
- at scale – pick and process millions of WARC records

Questions?

Questions? More examples and how it works?

Please visit my poster!

Thanks!