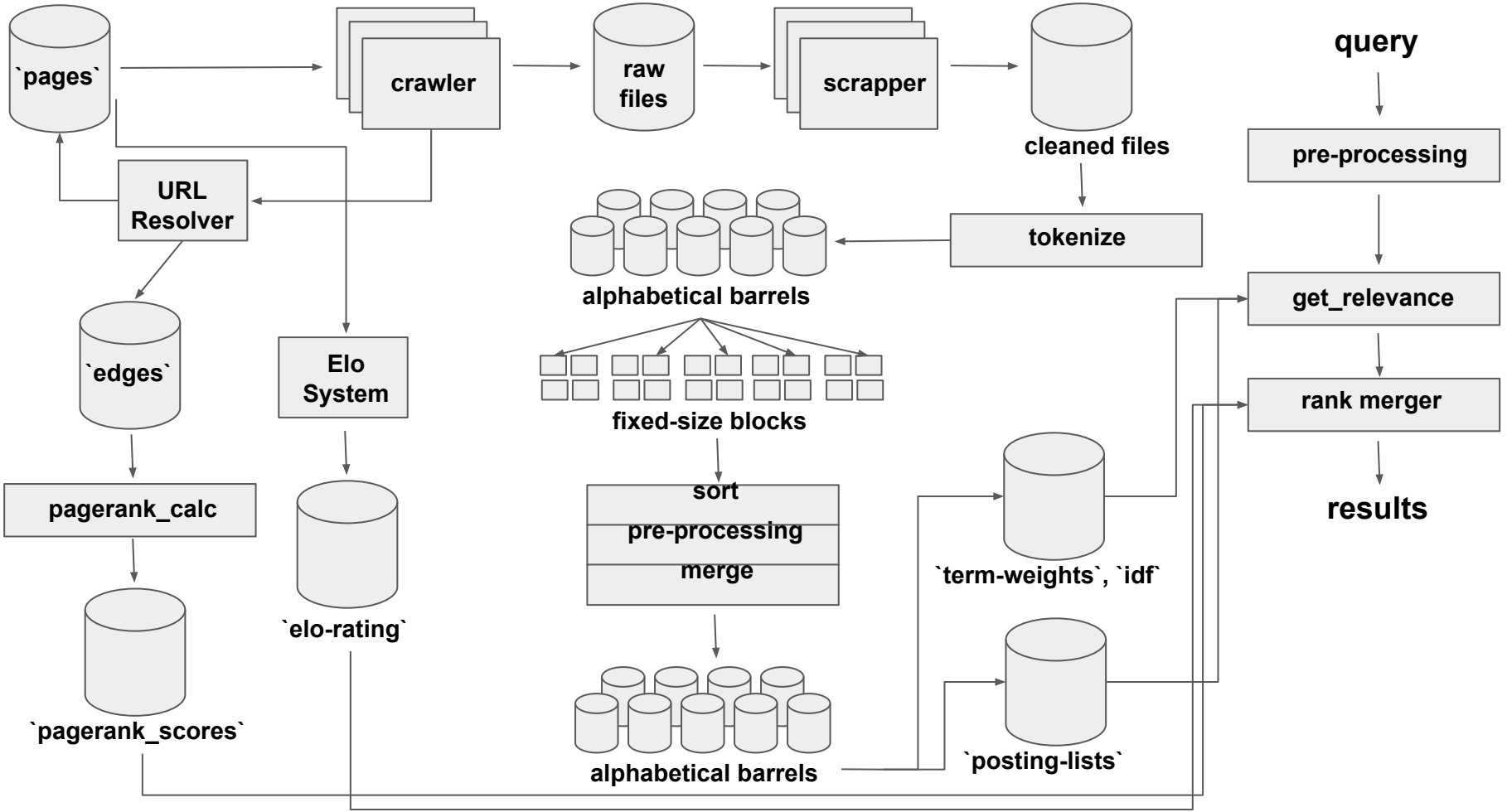
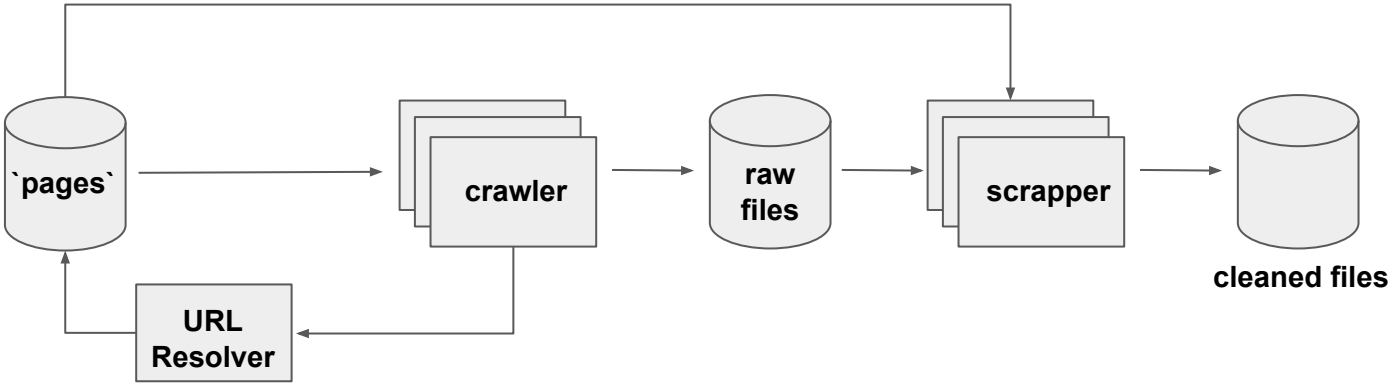
Improving Search Rankings by Incorporating Implicit User Feedback


`pages`
crawler
raw files
scrapper
cleaned files
query
pre-processing
URL Resolver
tokenize
alphabetical barrels
get_relevance
`edges`
Elo System
fixed-size blocks
rank merger
sort
pre-processing
merge
results
pagerank_calc
`term-weights`, `idf`
`elo-rating`
`pagerank_scores`
alphabetical barrels
`posting-lists`

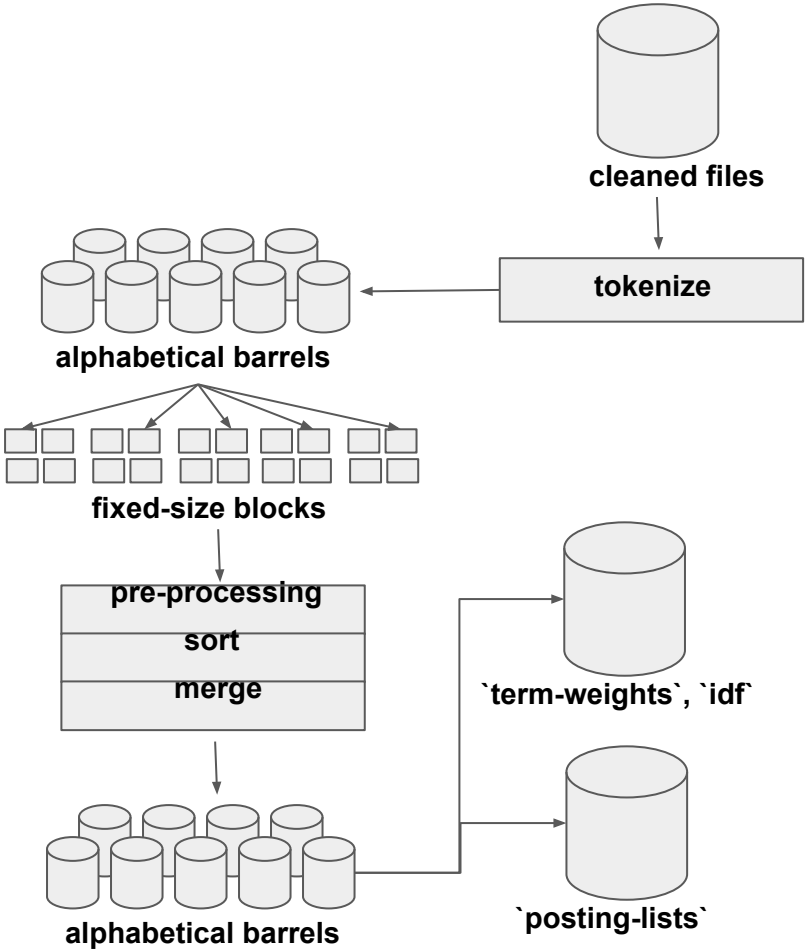
crawling


`pages`
crawler
raw
files
scrapper
cleaned files
URL
Resolver

crawling

- **Crawler :** Takes a URL from the database and fetches the corresponding page. Enumerates all the URL's to other Wikipedia pages and adds to the database.
- **Scrapper :** Takes a page from the raw files repository and processes the HTML document to remove HTML tags, irrelevant portions of the page.
- There are 4 threads each for crawler and scraper.

indexing


cleaned files
tokenize
alphabetical barrels
fixed-size blocks
pre-processing
sort
merge
`term-weights`, `idf`
alphabetical barrels
`posting-lists`

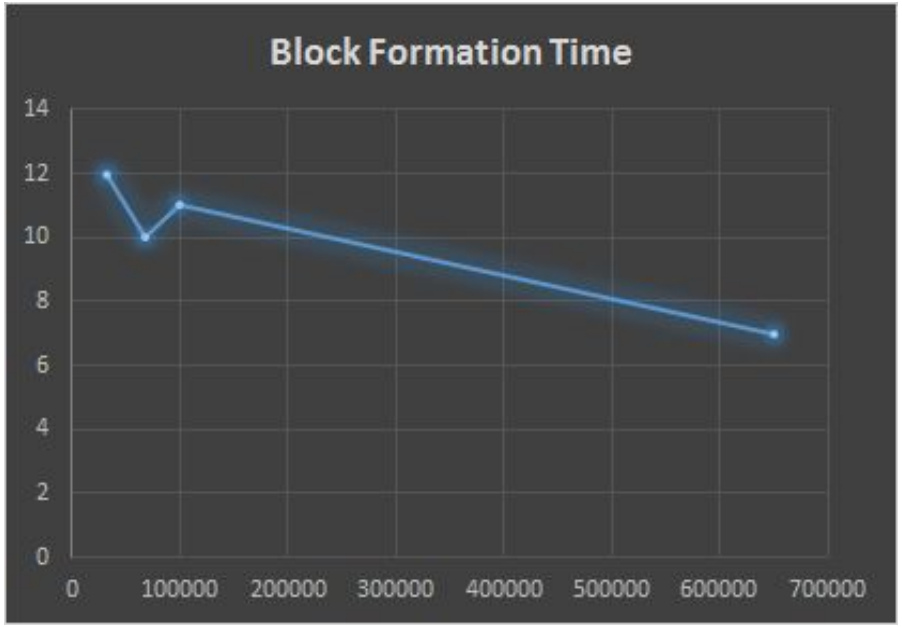
indexing

- 100,000 documents each document on an average has round about 5,000 words so lots of tokens which has to be stored rather efficiently
- **Tokenize :** read the document, map it to an integer id and tokenize it.
- **Alphabetical barrels :** to start with reduce the problem to simpler steps
- **Blocks :** now divide the created barrels into blocks of equal size for efficient sorting. Used samples to decide the barrel size- A,P,N.

Reduce number of files and time taken.

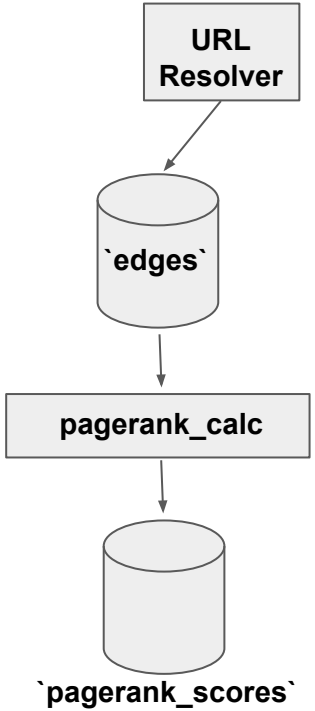
code profiling

Block Size fixed as - 6,50,000, to avoid stack smashing.



- Apply preprocessing. Implemented porter stemmer, removed stop words. Size of the list reduced significantly.
- Each block had around 6,50,000 /5 words in it after removing stop words
- Sort the blocks independently - above block size was the limit. Encountered stack smashing if size became higher- hardware limitation.
- Compress and merge-
    - Alone 21, Alone 22 could be combined to Alone 21->22 . Save space by compressing and meanwhile compute necessary frequencies for tf and df.
- Calculate tf and idf and postings store them in file.

pagerank



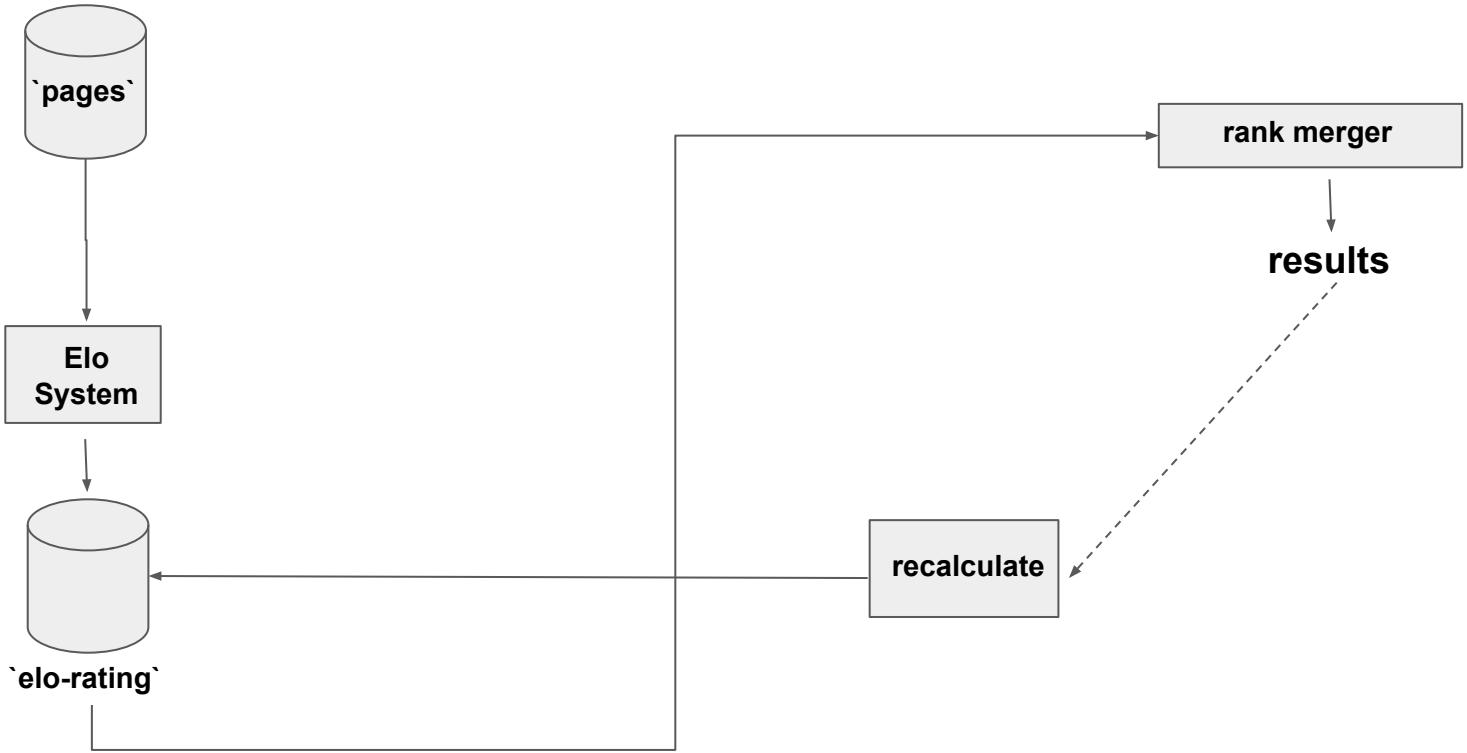
pagerank

- Uses the formula :

$$Rank^{(i+1)} = \beta \cdot M \cdot Rank^{(i)} + (1 - \beta) \cdot \vec{e}/n$$

- Converges to a point within the error in order of -8.
- Requires ~50 iterations
- We get **spider traps** and **dead ends** which consume page-rank scores, hence the sum of PageRank of all pages amounts to <1,so we scale up (normalize) the scores so that the sum of all PageRanks is 1.
- β = 0.85

elo rating


`pages`
rank merger
results
Elo System
recalculate
`elo-rating`

elo rating

- The relevant pages are ranked according to the elo rating of the page.
- The higher the rating of the page, the greater is the expectation that the page will be clicked by the user.

$$S_{exp} = 1 / (1+10^{\wedge}\textstyle\sum_{b}^{z} -(R_a - R_j))$$

- Also, the chance of being clicked decreases as the position of appearance of the result increases in a page. We damp the *Sexp* accordingly as :

$$S\exp' = S\exp \cdot e^{-0.025 \cdot p}$$

- *Sexp'* is finally normalized such that the sum is 1.
- Ratings are updated once a result is clicked from the served pages as :

$$r_{post} = r_{pre} + K(S - S_{exp}),$$

rank merging

- Relevant pages are ranked based on 3 measures :
  - Similarity with the query
  - PageRank
  - Elo Rating
- Rank from 3 sources are merged according to the following formulae to give the final score :

$$\textbf{Score(Merged)} = W_r * (1/R_r+1) + W_p * (1/R_p+1) + W_e * (1/R_e+1)$$

where,

R_r - Rank from similarity, R_p - Rank from pagerank score, R_e - Rank from Elo-rating score and W_r, W_p, W_e are respective weights given to each ranking

rank merging

- $W_R = 0.4$
- $W_P + W_E = 0.6$

- Weightage of W_E in 0.6 increase with the confidence we have in the elo ratings of the relevant pages. Confidence increases with the count of updates on each page.
- Initially, $W_E = 0$, and can increase maximum to 0.24 (0.4 of 0.6) , by the following formulae :

$$Conf = 0.4 \cdot \left(1 - e^{-\frac{hm}{K}}\right) \qquad W_E = 0.6 \cdot Conf$$