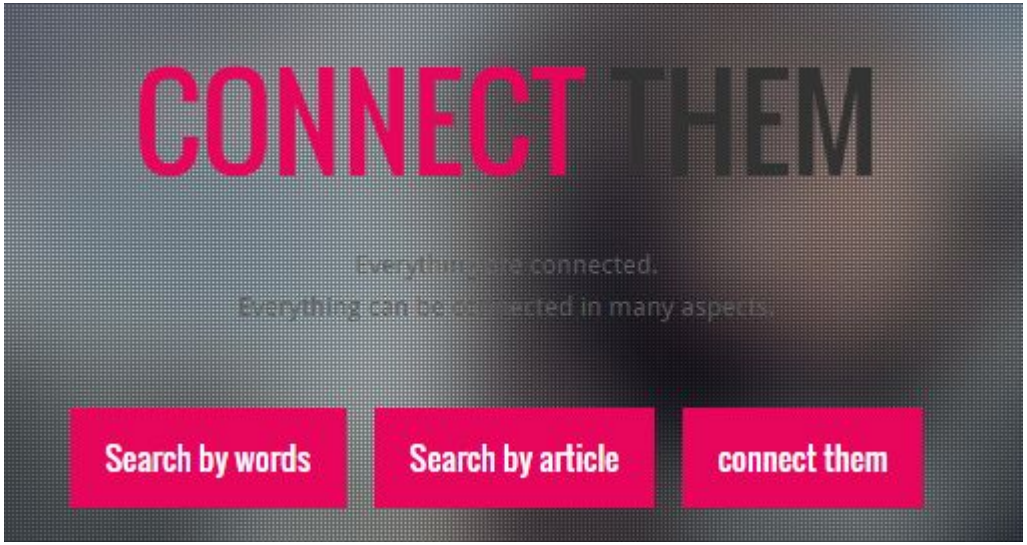
Connect Them

——A news search engine

Group2: 顾秀烨 施鹏 付春李

Functions

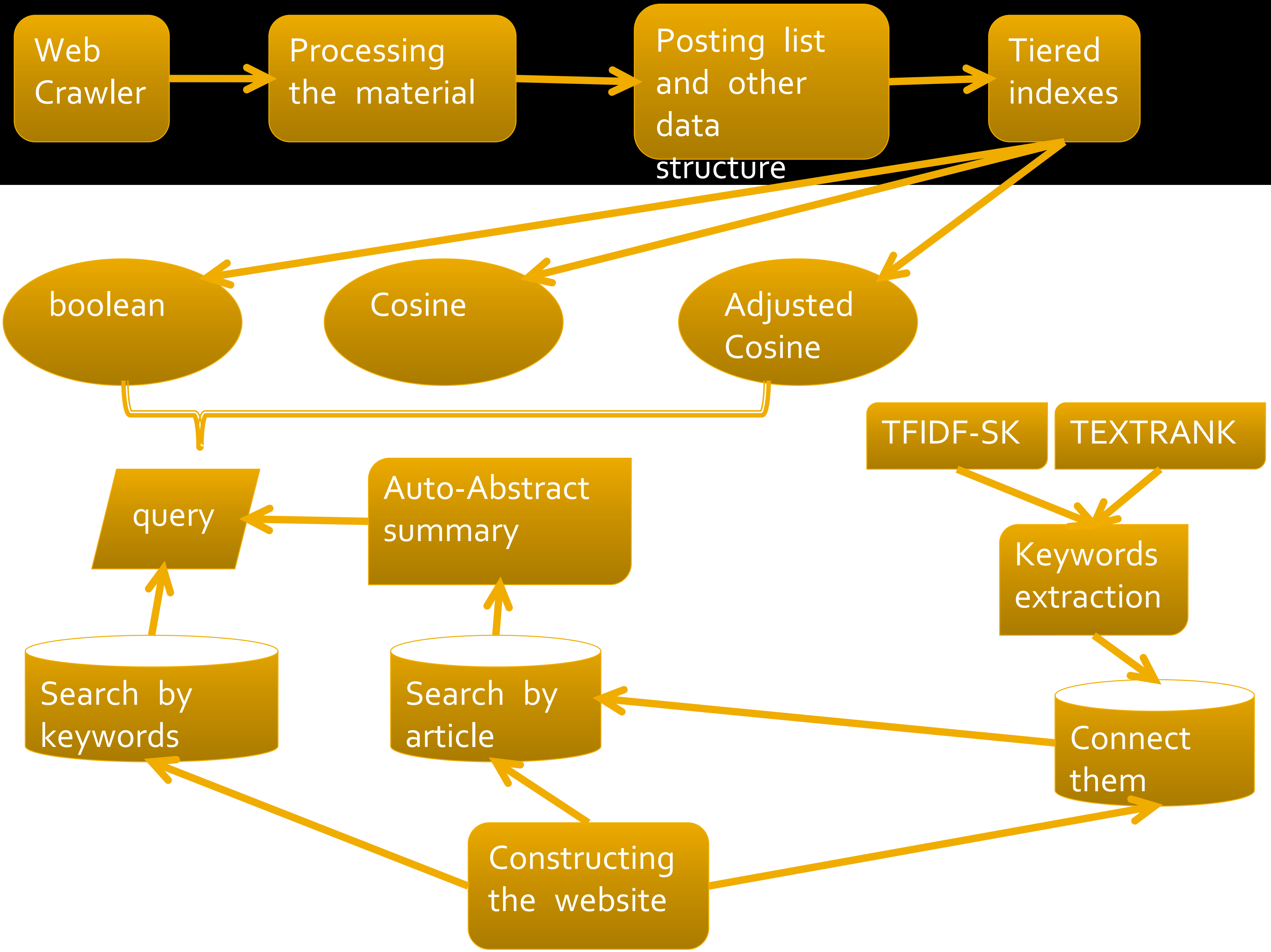
- ○ Search by words
- ○ Search by an article
- ○ Connect two different articles.



Everything are connected.
Everything can be connected in many aspects.

Design Philosophy

- Psychology
- Interdisciplinary
- Crossover
- Everything can be connected.
- Everything can be connected in many aspects.
- We can do more.



Web Crawler

Processing the material

Posting list and other data structure

Tiered indexes

boolean

Cosine

Adjusted Cosine

TFIDF-SK

TEXTRANK

query

Auto-Abstract summary

Keywords extraction

Search by keywords

Search by article

Connect them

Constructing the website

WEB CRAWLER

Crawl on the Guardian

- Queue, BFS
  - record the pages to crawl

- To continue crawling after stopping:
  - Save the queue and the list of the visited pages to the disk every time the crawler has stored 10 more articles.

- Regular Expression
  - Match the article pages

  -
    ```
    prog =
    re.compile("(http:\/\/)?www\.theguardian\.com\/(\w*?)
    \/\d{4}\/(\w*?)\/\d{2}\/.*”)
    ```

WEB CRAWLER

Crawl on the Guardian

- More details:
  - Encoding
    - Save the article: unicode -> utf-8   .encode(“utf-8”)
    - Reading the files: utf-8 ->unicode .decode(“utf-8”)
  - Try-Except mechanism
    - A necessity under the poor network condition
    - Avoid empty articles
  - Handle the url:
    - tag['href'] = urlparse.urljoin(url, tag['href'])
    - tag['href'] = tag['href'].split('#')[0]
    - nyprog.match(tag['href']) and tag['href'] not in page_visited
    - nyprog = re.compile("http\:\/\/www\.theguardian\.com.*”)

Basic search techniques

- Boolean
- Cosine
- Pivot normalized cosine

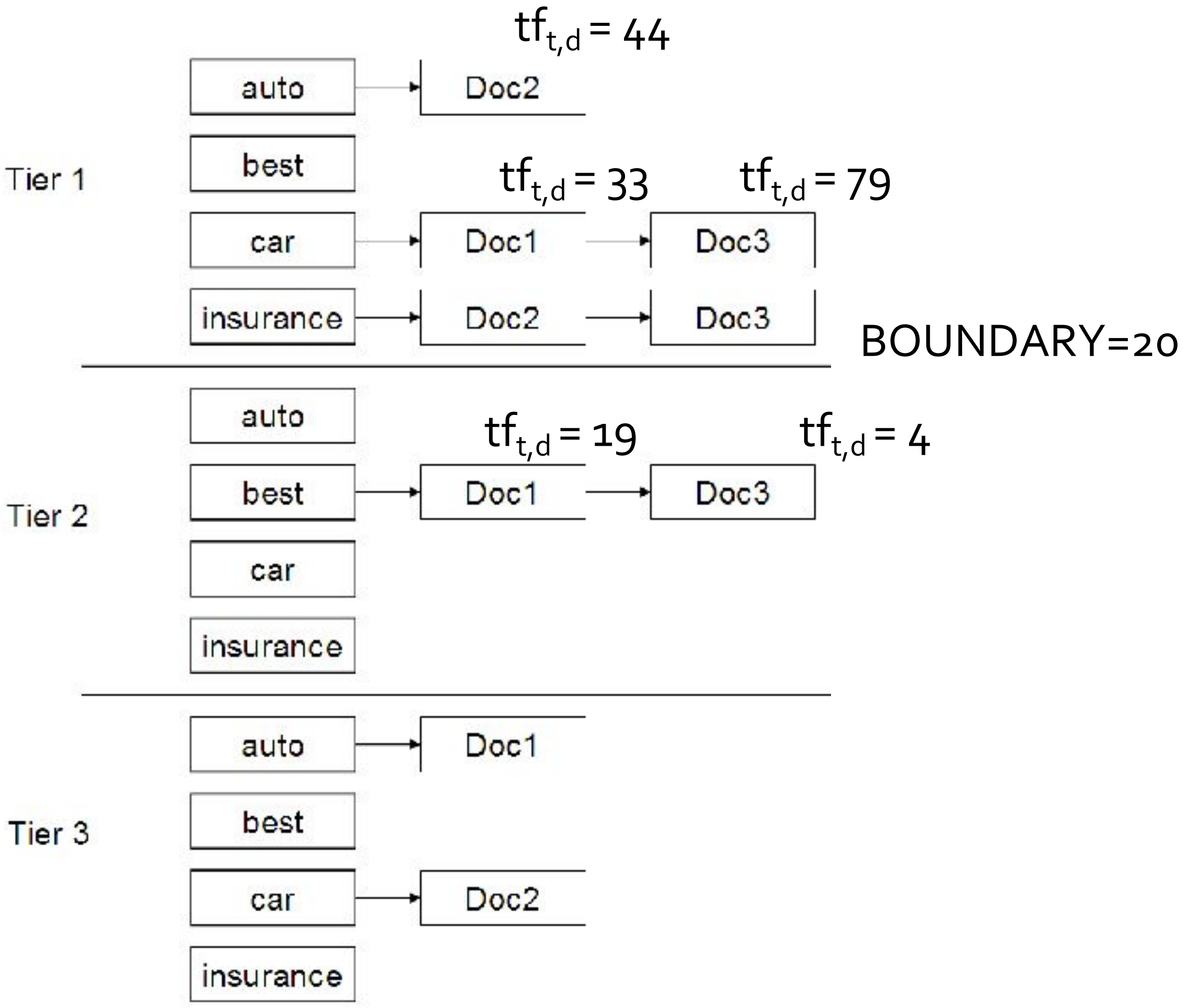
$$w_{ij} = \frac{\log(dtf)+1}{sumdtf} \times \frac{U}{1+0.0118U} \times \log\left(\frac{N-nf}{nf}\right)$$

- Levenshtein Distance

Tiered Indexes

- Pruning policy
  - Document pruning
  - extended keyword-specific document pruning based on tf
  - If $tf_{t,d}$ > BOUNDERY, Add the DocId to the term's 1st posting list
  - K
- A bold try
  - When making the posting list(only record frequency)
  - Title * 4, Description and 1st paragraph * 2
  - $tf_{t,d}$ is higher
  - Documents may have a better chance to appear in their title's 1st tier posting list

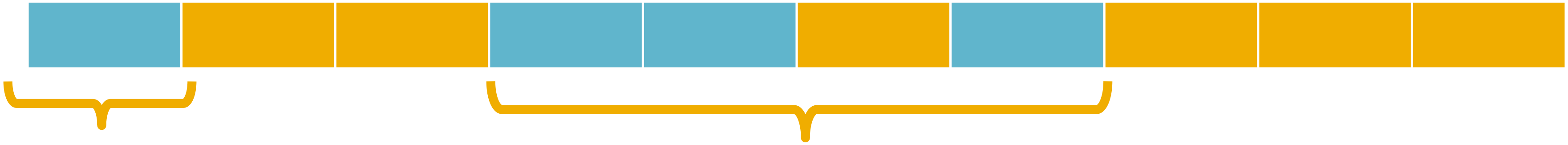
Tiered Indexes



Luhn’s Auto-Abstract Algorithm

- Score the sentences
  - Use the selected sentences to generate the summary
- Cluster
  - If important words are clustered in a sentence. The sentence will get a higher score.

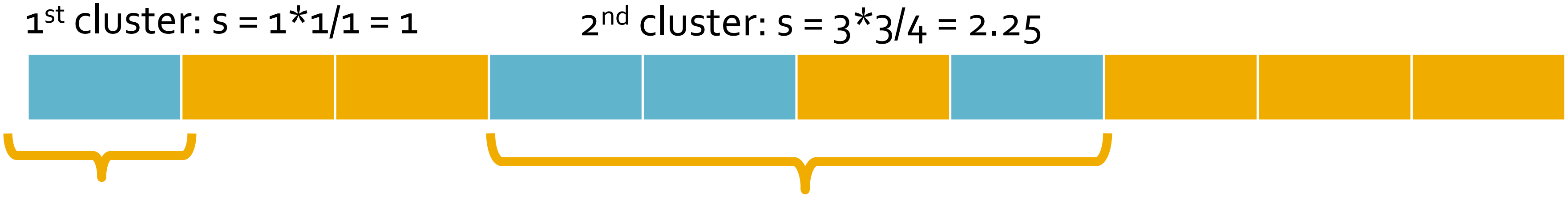
Luhn’s Auto-Abstract Algorithm



- Blue indicates important words

- Important words: top n frequent words in the whole article
    - nltk.probability.FreqDist or made by hand
- Cluster
    - CLUSTER_THRESHOLD = 3 (4 or 5 is suggested)
    - `if word_idx[i] - word_idx[i - 1] < CLUSTER_THRESOHLD:`
    - `    cluster.append(word_idx[i])`

Luhn’s Auto-Abstract Algorithm



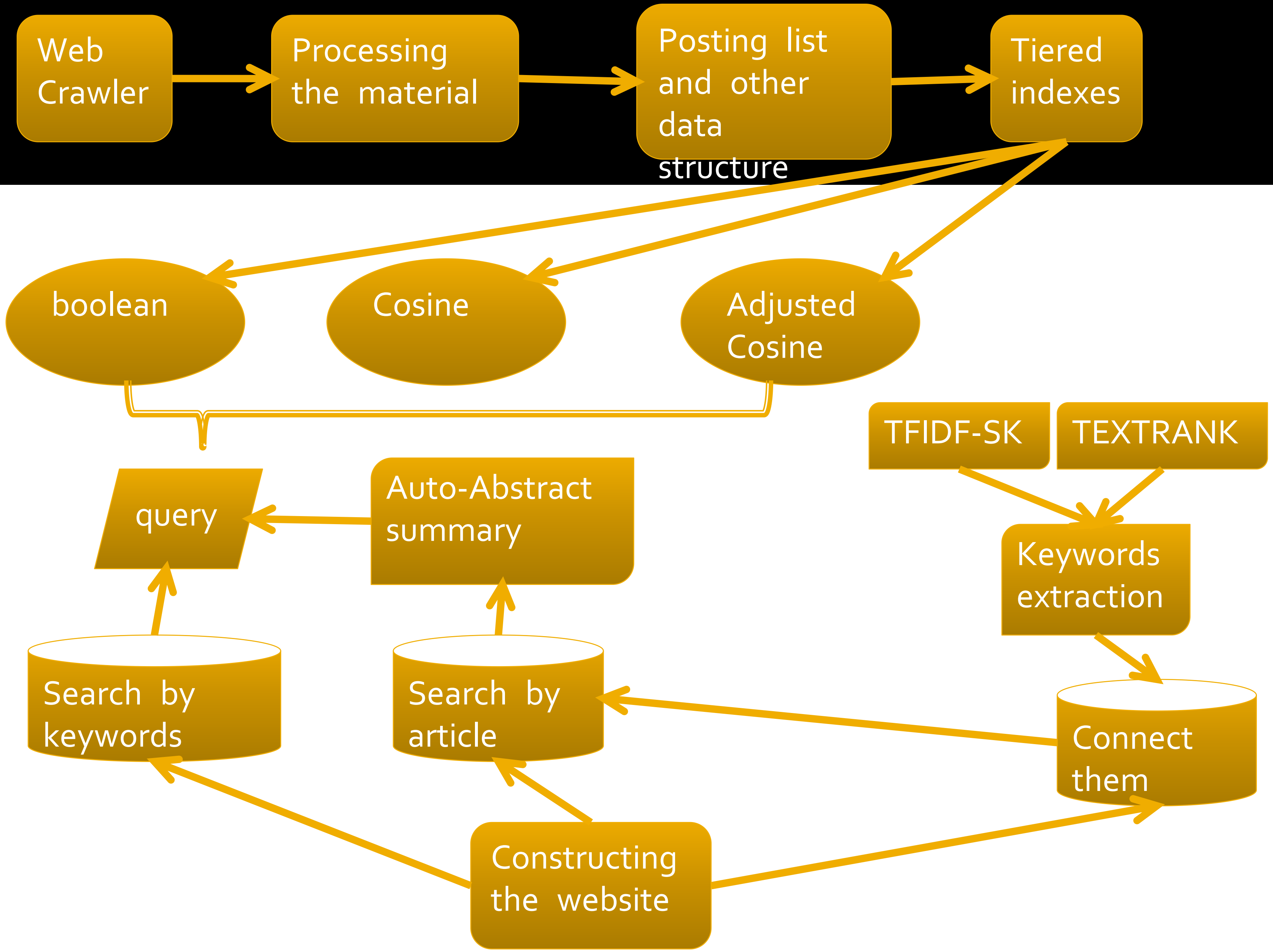
1st cluster: s = 1*1/1 = 1

2nd cluster: s = 3*3/4 = 2.25

- How to score each sentence?
  - Score the cluster first
  - The score of a cluster = $\dfrac{\text{total of significant words in a cluster}^2}{\text{total words in a cluster}}$
  - The score of a sentence = the maximum score among its clusters’ score
- Sentence score = 2.25

Luhn’s Auto-Abstract Algorithm

- How to select the scored sentences?
  - Approach1: Simply select top N sentences with highest scores
    - You can define the length of the summary
  - Approach2: Statistic threshold
    - if score > avg + 0.5 * std (numpy)
      - Avg: average score; Std: standard variance
    - If the score of the sentences are very close to each other, approach2 is better.



Web Crawler

Processing the material

Posting list and other data structure

Tiered indexes

boolean

Cosine

Adjusted Cosine

TFIDF-SK

TEXTRANK

query

Auto-Abstract summary

Keywords extraction

Search by keywords

Search by article

Connect them

Constructing the website

Key Words Extraction - TFIDF–SK

- Base: TF-IDF Algorithm
- Problem
- We can make some improvement.

Key Words Extraction - TFIDF–SK

- Pos_{ij}: Weight of W_i appearing in the document D_i in the first time.
- $$Pos_{ij} = \begin{cases} 1 & title\ or\ summary \\ 0.6 & first\ or\ last\ paragraph \\ 0.2 & others \end{cases}$$

Key Words Extraction

- Noise term: terms which have little connection with the theme
- High tf and high df
- Coefficient of dispersion (CV)
- $CV_i = \frac{SD_i(TFDf_{ij})}{AVE_i(TFDf_{ij})}$
- SD: Standard deviation
- AVE: Average
- Lower CV means higher possibility of noise term

Key Words Extraction

- Term co-occurrence possibility
- If two terms appear in one sentence, there term co-occurrence add 1.
- Eg:

	A	B	C	D	E	Sum
A	-	30	26	19	18	93
B	30	-	5	50	6	154
C	26	5	-	4	23	93
D	19	50	4	-	3	89
E	18	6	23	3	-	89

- $\{x_{a1}, x_{a2}, x_{a3}, x_{a4}\}$ = {30/93,26/93,19/93,18/93}

Key Words Extraction - TFIDF–SK

- Measure of skewness
- To measure the asymmetric degree in statistical data.
- $SK_i = \frac{(N-1)\sum_j(x_{ij}-\text{avg}(x_i))^3}{(N-2)(N-3)SD_i{}^3}$ (N>=4)
- x_{ij}: term co –occurrence possibility of i,j

Key Words Extraction - TFIDF–SK

- Importance measuring function:
- TFIDF-SK$_i$= $\alpha\sum_j (Pos_{ij} * TFIDF_{ij}) + \beta\, SK_i$
- α,β are modifiable parameters

Key Words Extraction-Textrank

- Pagerank:

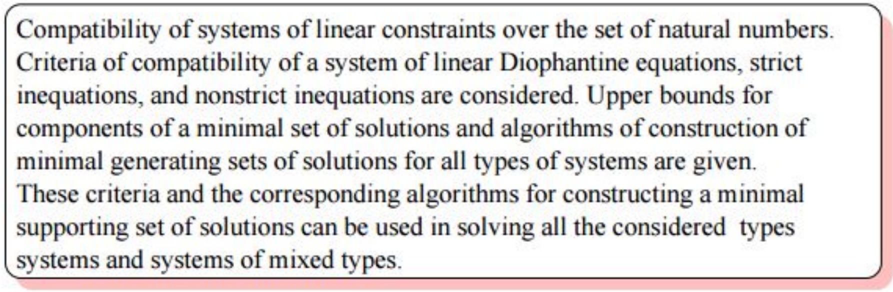
$$S(V_i) = (1 - d) + d * \sum_{j \in In(V_i)} \frac{1}{|Out(V_j)|} S(V_j)$$

- d is a daming factor that can be set between 0 and 1, which is usually 0.85.

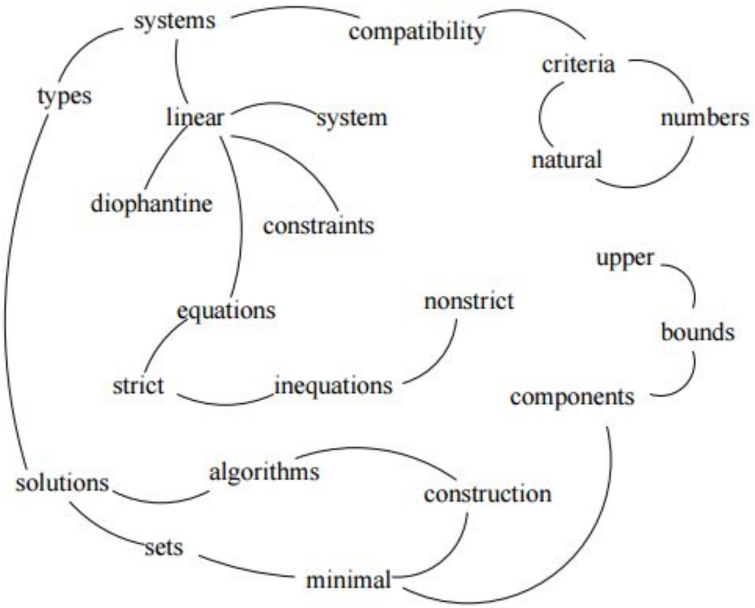
Key Words Extraction -Textrank

- Text rank:
- For every sentence, we can connect the words using the parameter window k:
- Sentence: w1,w2,w3,w4,w5,...,wn
- {w1,w2,...,wk}, {w2,w3,...,wk+1}, {w3,w4,w5,...,k+2} are all a window, two terms in a window can be connected in the graph.

Key Words Extraction -Textrank



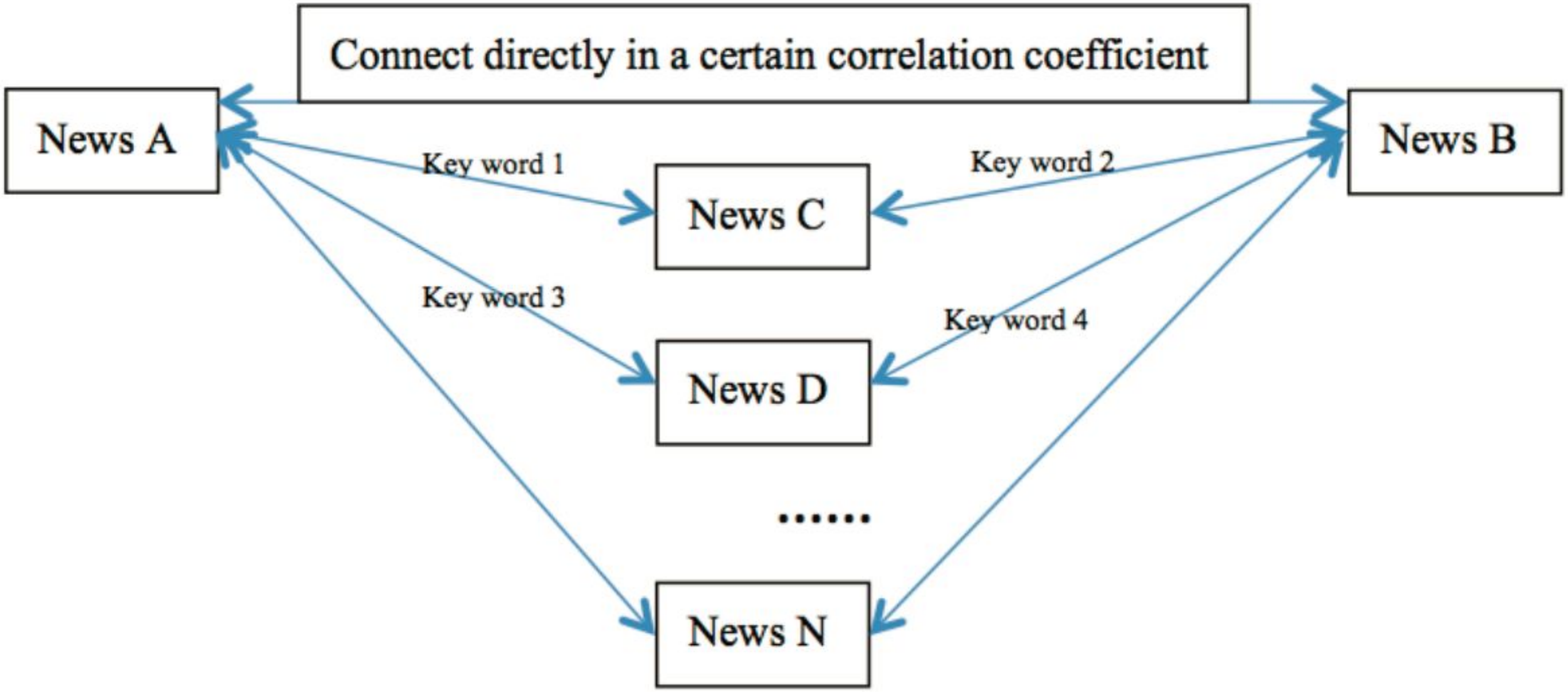
Compatibility of systems of linear constraints over the set of natural numbers. Criteria of compatibility of a system of linear Diophantine equations, strict inequations, and nonstrict inequations are considered. Upper bounds for components of a minimal set of solutions and algorithms of construction of minimal generating sets of solutions for all types of systems are given. These criteria and the corresponding algorithms for constructing a minimal supporting set of solutions can be used in solving all the considered types systems and systems of mixed types.



Key Words Extraction -Final

- $Importance_{\,i}$ =TFIDF–SK$_i{}^{\lambda}$ * S$(V_i)^{u}$
  TFIDF–SK$_i$= α$\sum_j(Pos_{ij}$* $TFIDF_{ij}$) + β SK_i
- $S(V_i) = (1 - d) + d * \sum_{j \in In(V_i)} \frac{1}{|Out(V_j)|} S(V_j)$

Connect them



Connect them

- Connect directly coefficient:
- $importance_i$ in news A * $importance_i$ in news B
- Connect indirectly coefficient:
- $importance_i$ in news A * $importance_i$ in news C * $importance_j$ in news B * $importance_j$ in news C

Expectation

- Better stemming
- Phrase process
- Speed
- LSI LDA
- Testing and adjusting the coefficients
- Search in other aspects

Reference

- The Significance of Normalization Factor of Documents to Enhance the Quality of Search in Information Retrieval Systems. Hossein sadr, Reza Ebrahimi Atani, MohammadReza Yamaghani
- The Automatic Creation of Literature Abstracts, H.P. Luhn
- on the statistical features-based information keyword extraction method in the era of big data, Luo Fanming