JUNAID GIRKAR
6000419005T

29/3/2023

WI ASSIGNMENT - 1

BE COMPS A2

Q3 Explain vector space model clearly with an example

ANS

- Information retrieval is study of helping users to find information that match their information needs. It is about acquisition, organization, storage, retrieval and distribution of information.
- An IR model governs how a document and query are represented and how relevance of document to user query is defined.
- There are 4 main IR models :-
  - → Boolean
  - → Vector space
  - → Language
  - → Probabilistic

- Important terms in IR Model :=

1. Each document and query treated as a bag of words.
2. Each term associated with a weight
3. Given collection of documents D, V (vocabulary) $= \{t_1, t_2, \ldots t_{|V|}\}$ set of distinct terms in collection D.
4. Weight $w_{ij} > 0$ associated with term t_i of document $d_j \in D$; if term does not appear in d_j then $w_{ij} = 0$
5. $d_j = (w_{1j}, w_{2j}, \ldots, w_{|V|j})$, collection of d_j represented as matrix, in different models w_{ij} is computed differently.

__/__/____

⇒ VECTOR SPACE MODEL

- Best known and widely used IR model

• Document Representation.

A document in vector space model is represented as a weight vector, in which weight of each component is computed based on some variation of TF or TF-IDF scheme. Thus w_{ij} can be any number

(i) **TERM FREQUENCY (TF) scheme:**

The weight of term t_i in d_j is number of times t_i appears in document d_j denoted by f_{ij}. Normalization may also be applied.

$$tf_{ij} = \frac{d_{ij}}{\max\{f_{ij}, f_{2j}, \ldots f_{|V|j}\}}$$

The shortcoming of TF scheme is that it doesn't consider a situation where a term appears in many documents of the collection. Such a term may not be discriminative.

(ii) **TF - IDF SCHEME:**

Most well known weighting scheme, TF stands for term frequency & IDF stands for inverse document frequency.

Let N be total number of documents in collection. df_i be number of documents

in which term t_i appears atleast once. f_{ij} be raw count of t_i in d_j, then the normalized term frequency of t_i in d_j is given by

$$tf_{ij} = \frac{f_{ij}}{\max\{f_{ij}, f_{2j}, \ldots f_{ivj}\}}$$

where maximum is computed over all terms that appears in document d_j.

The inverse document frequency of term t_i is given by, $idf_i = \log\left(\frac{N}{df_i}\right)$

The intuition is that if term appears in large number of documents, it is probably not important or that discriminative. The final TF-IDF weight is given by: $w_{ij} = tf_{ij} \times idf_i$

- **QUERIES**

A query q is represented in exactly the same way as document in collection. The term weight w_{iq} of each term t_i in q can also be computed in same way as normal document.

- **DOCUMENT RETRIEVAL & RELEVANCE RANKING**

The documents are ranked according to their degrees of relevance to the query. One way to compute degree of relevance is to calculate similarity of query q to each document d_j in D. There are many similarity measures,

well known one is cosine similarity, which is cosine of angle between q and d_j

$$\text{cosine}(d_j, q) = \frac{\langle d_j \cdot q \rangle}{\|d_j\| \times \|q\|} = \frac{\sum_{i=1}^{|V|} w_{ij} \times w_{iq}}{\sqrt{\sum_{i=1}^{|V|} w_{ij}^2} \times \sqrt{\sum_{i=1}^{|V|} w_{iq}^2}}$$

Ranking of documents is done using similarity values.

Some other methods of computing relevancy scores of document d_j with query q are: simple dot product, okapi method, pivoted normalization weighting.

EXAMPLE

Documents : N = 3

d1 – New York times
d2 – New York Post
d3 – Los Angeles times

Query q = new new times

Documents :

Term frequency matrix $\frac{d_{ij}}{\max\{f_{1j}, f_{2j}, \ldots f_{|V|j}\}}$

	d1	d2	d3
new	1	1	0
york	1	1	0
times	1	0	1
post	0	1	0
los	0	0	1
Angeles	0	0	1

Inverse Document Frequency of terms = $\log\left(\frac{N}{df_i}\right)$

new	log (3/2)	= 0.176
york	log (3/2)	= 0.176
times	log (3/2)	= 0.176
post	log (3/1)	= 0.477
los	log (3/1)	= 0.477
angeles	log (3/1)	= 0.477

TF - IDF Score Matrix ($tf_{ij} \times idf_i$)

	d1	d2	d3
new	0.176	0.176	0
york	0.176	0.176	0
times	0.176	0	0.176
post	0	0.477	0
los	0	0	0.477
angeles	0	0	0.477

Query : TF - IDF score

	q	
new	2/2 x 0.176	= 0.176
york	0	0
times	1/2 x 0.176	= 0088
post	0	0
los	0	0
angeles	0	0

Computing values of

$$\|d1\| = \sqrt{0.176^2 + 0.176^2 + 0.176^2} = 0.305$$

$$\|d2\| = \sqrt{0.176^2 + 0.176^2 + 0.477^2} = 0.538$$

$$\|d3\| = \sqrt{0.176^2 + 0.477^2 + 0.477^2} = 0.697$$

$$\|q\| = \sqrt{0.176^2 + 0.088^2} = 0.196$$

Finding cosine similarities (dj, q).

$$\cos(d1, q) = \frac{\langle d1, q \rangle}{\|d1\| \times \|q\|} = \frac{0.031 + 0.015}{0.305 \times 0.196} = 0.7695$$

$$\cos(d2, q) = \frac{\langle d2, q \rangle}{\|d2\| \times \|q\|} = \frac{0.031}{0.538 \times 0.196} = 0.2940$$

$$\cos(d3, q) = \frac{\langle d3, q \rangle}{\|d3\| \times \|q\|} = \frac{0.015}{0.699 \times 0.196} = 0.1098$$

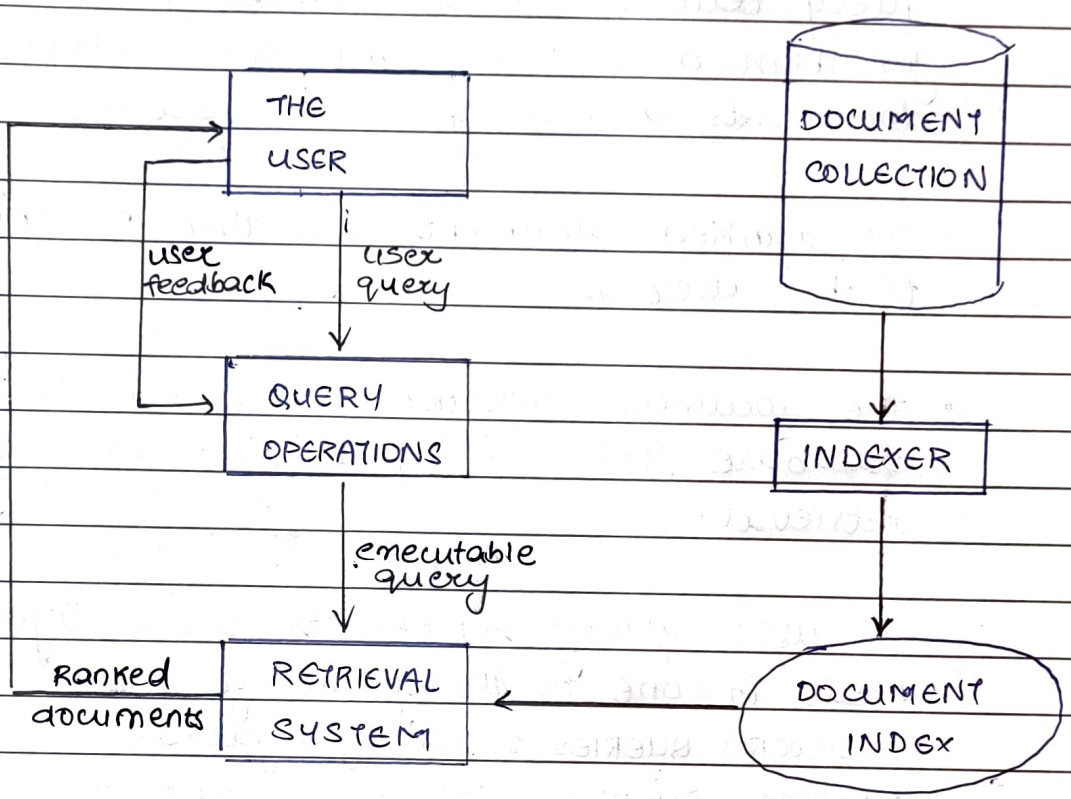
$\therefore$ $0.7695 > 0.2940 > 0.1098$

$\therefore$ Based on query, documents will be presented in the order $d1 > d2 > d3$

__/__/__

Q4 Draw and explain general IR system Architecture.

ANS



GENERAL IR SYSTEM

- Information retrieval is the study of helping users find information that matches their information needs. It studies the acquisition, storage, retrieval and distribution of information
- IR is about document retrieval, emphasizing document as basic unit.
- The user with information neds issues a query (user query) to the retrieval system through query operations module.

__/__/__

- The retrieval module uses document index to retrieve those document that contain some query terms, compute relevancy scores for them and then rank the retrieved documents according to the scores.

- The ranked documents are then presented to the user

- The document collection is called the text database, indexed by indexer for efficient retrieval.

- A user query represents users information needs in one of the following forms:
  (i) KEYWORD QUERIES : List of keywords
  (ii) BOOLEAN QUERIES : Boolean operators to construct complex queries
  (iii) PHRASE QUERIES : sequence of words
  (iv) PROXIMITY QUERIES : Relaxed version of phrase query and can be combination of terms and phrases.
  (v) FULL DOCUMENT QUERIES : Full document
  (vi) NATURAL LANGUAGE QUERIES : human languages. Also known as answering questions.

- Query operation module in simplest case does nothing and pass query to retrieval module after pre-processing (removing stopwords). In complex cases, it transforms natural language

__/__/____

queries to executable queries. It may also accept user feedback and use it to expand & refine original queries called as relevance feedback.

- The indexer indexes raw documents for efficient retrieval, result is document index, most popular indexing scheme is inverted index.

- The retrieval system doesn't compare query with every document in collection which is inefficient. Instead, a small subset of documents containing atleast one query term is first found from index and then relevance scores are computed for this subset of documents with user query.

- An IR model governs how a document & query are represented and how relevance of document to user query is defined.

- There are 4 main IR models :-
  (i) Boolean Model (simplest)
  (ii) Vector space Model (widely used)
  (iii) Language Model (NLP)
  (iv) Probabilistic Model (Stochastic).

WI ASSIGNMENT 2

03/05/2023

JUNAID GIRKAR
60004190057
BE COMPS A2

Q1 Explain web crawling

ANS
- web crawlers are programs that traverse webpages while downloading and indexing.
- Information is ~~retrieved~~ distributed and users traverse via hyperlinks from one page to another.
- Data collected by crawlers can be analysed and mined in central locations.
- The web is a dynamic entity. Thus crawlers are needed to study the current, added and deleted pages overtime.
- Crawlers are thus the main consumers of internet bandwidth

APPLICATIONS OF WEB CRAWLERS:

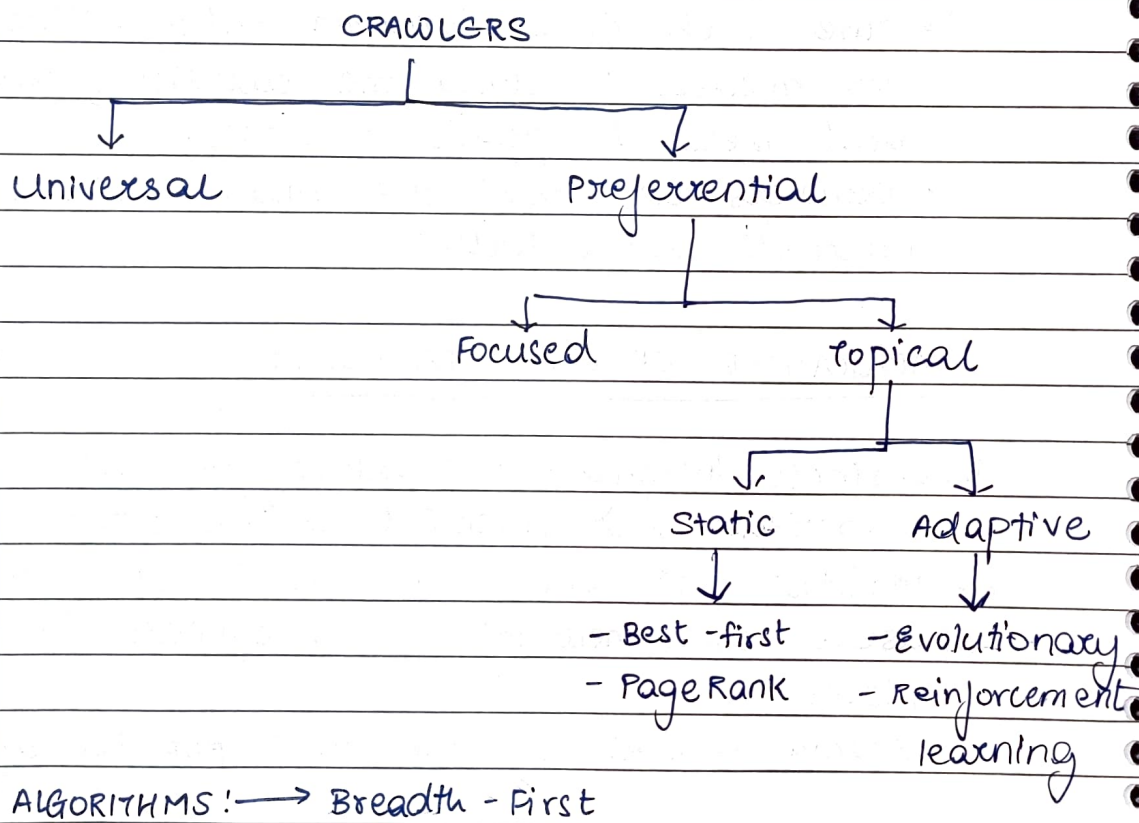
1. Business Intelligence: Collect information about competitors and potential collaboxators.
2. Monitor sites and pages of interest to notify users when new information appears in certain places
3. Search Engines are the most popular application

__/__/____

MALICIOUS APPLICATIONS

1. Harvesting email addresses by spammers
2. Collect personal information using crawlers for phishing
3. web spamming and false search optimization.

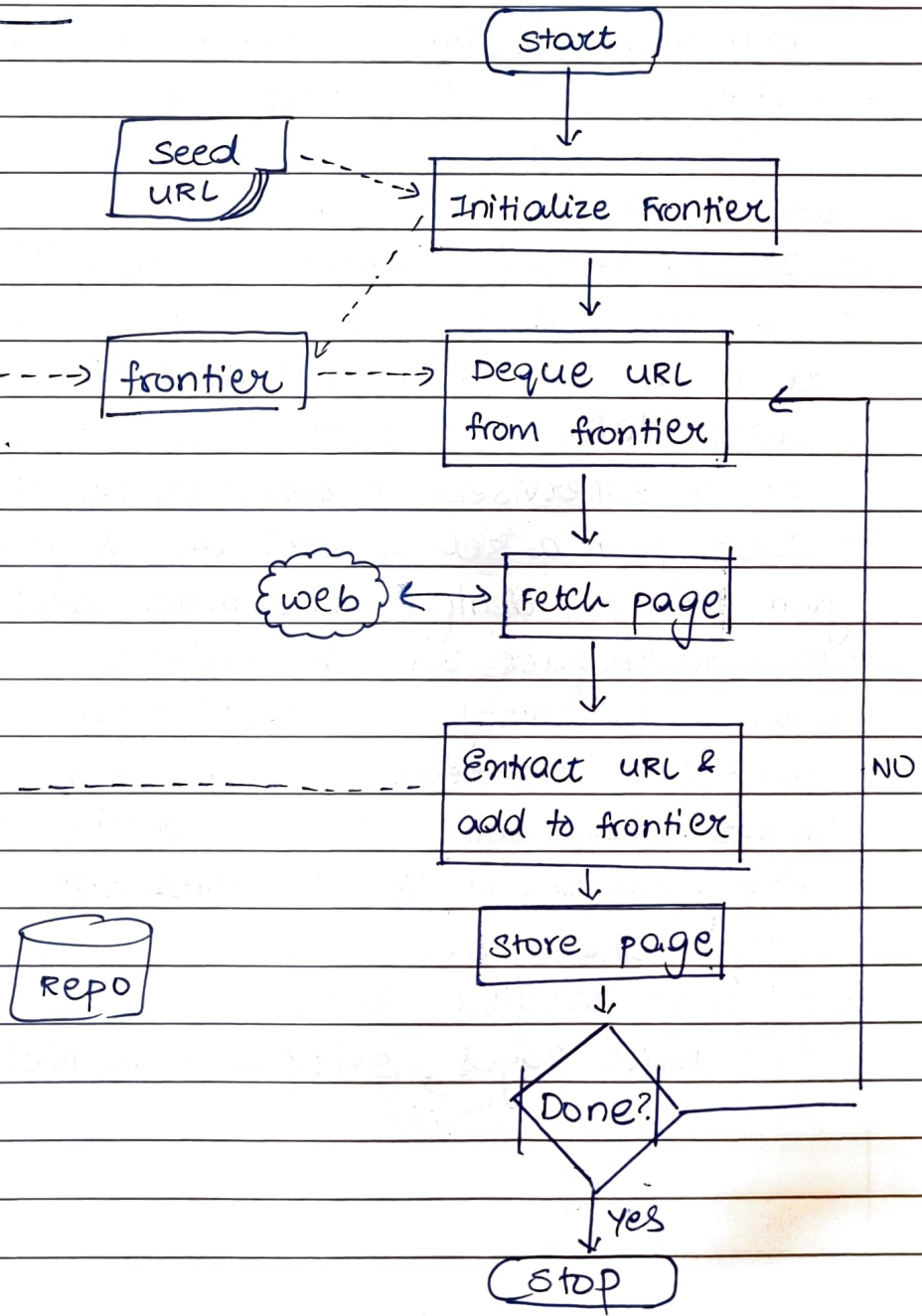
- Crawlers collect pages for search engines to build their indexes.
- Preferential crawlers are more targeted



ALGORITHMS :⟶ Breadth-First
⟶ Best First
⟶ Page Rank
⟶ Shark Search
⟶ InfoSpiders

- The types of crawlers are thus universal, focussed and topical
- Transfer of data due to crawlers is huge
- There are ethical issues regarding the use of crawlers
- They can cause inadvertent denial of service to legitimate users.

WORKING:



__/__/__

Graph traversal for frontier

1. Breadth First Search
   - Along shortest Path
   - Start with good page
   - Give other good pages

2. Depth First Search
   - Stack Based
   - Wanders away

2. FOCUSED CRAWLERS
   - They are selective crawlers that are designed to gather and index web pages based on specific parts of web such as a set of websites.
   - It essentially works on multiple topics.
   - Targets specific domain or a set of domains. It is useful when researchers want to collect data from learned set of web pages
   - It is supervised & based on labelled examples.
   - Starts from a set of seed URLs & follows link's from those to identify other pages related to topic.
   - uses techniques such as keyword & content analysis to identify related pages.
   - used for web research, data mining where target data collection is required. They require more manual config & maintenance as target changes over time.
   - It uses distiller to find hubs
   - Eg: Naive Bayes, SVM, Neural Networks.

3. Topical Crawlers

- It is designed to search and index webpages based on specific topic or set of topics.
- Provides more relevant results on particular topic or domain compared to universal crawler.
- It starts with a set of seed URL's or even a query that highlights the topic.
- Unlike focused crawlers, it does not have a labelled dataset of positive & negetive pages
- They do not have text classifiers to guide crawling
- As pages are visited, they are sorted by user provided metrics like relevancy & score.
- The score can be something as simple as cosine similarity.
- The best advantage is all hits are fresh by definition.
- The downside is its slow speed ~~are~~ due to live crawling speeds.
- It cannot take advantage of global prestige measures. e.g. Rank of PageRank.
- used for information retrieval, SEO, etc.
- Can also be used to monitor changes in particular domain or topic over time.

IMPLEMENTATION ISSUES

1. Need to keep track of visited pages.
2. Frontier size grows fast.
3. Fetcher: must be robust, & should not crash in web errors

4 Parser : HTML has DOM tree structure . It is not enforced & can be syntactically incorrect.

5 Stopwords : Noise words that don't carry meaning should be removed. (A, An, the ...)

6 Conflation & Thesaurus : Improve recall by merging synonyms & similar words using hash tables.

7. URL canonicalization : Convert URL to standard form.

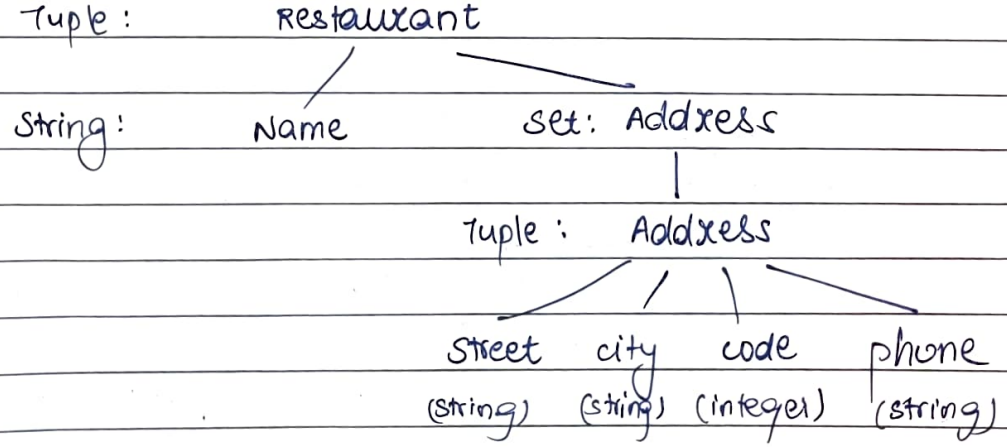
8. Spider traps : Avoid circular loops & multiple URL's pointing to same webpage with arbitrary depth.

9. Page Repository : Each page separately leads to huge store

10. Concurrency : Delays & overlaps in getching multiple pages

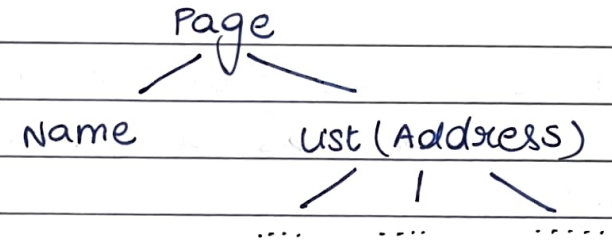
Q2 Wrapper Induction

ANS
- It is the process of mapping the web as nested relations
- It uses tree structure to facilitate extraction rule learning & further data extraction.



Extraction from page

- Web page is sequence of tokens 's'. Extraction is done using EC [Embedded Catalogue] tree which models data embedding in a page.
- Root is whole token sequence 's' & child is subsequence of sequence of parent node.
- To extract node of interest, wrapper uses EC description of page & set of extraction rules



- For each node of interest, wrapper extracts content from parent which contains sequence of tokens of all children.
- It uses 2 rules
  1. Start rule → beginning of node
  2. End rule → end of node.
- It is based on the idea of landmarks
- A landmark is an entity which functions as a token marking the beginning or end of an extraction.
- Rules are applicable to leaf & last nodes
- Applied to each record.
- Given EC tree and rules any node can be extracted by following path from root to node by extracting each node in path from parent.
- E.g: Name : Name of Restaurant

__/__/____

To get name of restaurant :
→ Start rule : SkipTo ()
→ End rule : SkipTo ()

LEARNING EXTRACTION RULES

- To generate start rule for node in EC tree, prefix tokens or wildcards are identified as landmarks.
- To generate end rule for node, suffix tokens or wildcards are identified as landmark
- Rule generation process is almost same and applications are similar except to apply. Start rule - system starts from first to last sequence in parent
- End rule starts from end to first in parent
- It uses labels to target items in few training examples, given set of labelled examples, algorithm should generate rules that extract all target items.
- learning is done using ML methods.
- Sequential covering provides the extraction rules.
- Active learning approach helps in identifying informative unlabelled examples.
- It is applied in case of lack of manual labels for a large dataset

__/__/____

ACTIVE LEARNING

- Given a small subset of labelled examples, it identifies ~~teh~~ unlabelled examples which might be informative to the user.
- To ensure accurate learning, a large number of training examples are needed.
- It thus minimizes manual labelling
- The key is to use 'co-testing' to identify informative examples
- It exploits the fact that there are multiple ways to extract the same information.
- The system thus learns forward & backward rules to locate the same item.
- They are named as such in the order in which they consume tokens.
- Given an example, if both rules agree, the extraction is likely to be correct
- In case of disagreement, it is passed to user.

WRAPPER MAINTAINANCE

- Once a wrapper is generated, it is applicable to other web pages that contain similar data and are formatted in the same ways.

- It introduces new problems:
  1. Wrapper Verification Problem
     + If the size changes, how should it be tracked.

2. If the change is correctly detected, then wrapper repair problem
+ How to repair an inconsistent wrapper.

- One way to deal with this is to learn patterns & rules of extracted items. They can then be checked to verify if execution is correct.

- If the patterns can be used to relocate some data on the page, then it has minor changes this is known as relabelling.

- After relabelling, re-learning is performed to re-learn a new wrapper.

- These methods are still difficult due to contextual and/or semantic information is often needed to find new locations of target items.

__/__/____

JUNAID GIRKAR
G00041900S7
BE COMPS A2

WEB INTELLIGENCE
ASSIGNMENT - 3

Q1 Explain preprocessing for schema matching.

⇒ Issues such as concatenated words, abbreviations, acronyms are dealt with i.e. need to be normalized before using it in matching.

1. TOKENIZATION

- It breaks an item (element or attribute value) into atomic words. Delimeters (-, /, +) and case changes are used to suggest breakdowns.

e.g. fromCity → from city
FirstName → First Name

- A domain dictionary of words maintained to help breakdown (if not present then add to dictionary else use for breakdown)

e.g. deptcity → {dept city} if city in dictionary
Dictionary constructed automatically consisting of words in input of matching (schema, domain & instance values) and dict updated as preprocessing progresses.

- Tokenization has to be done with case.

e.g. Bathroom $\xrightarrow{\times}$ {Bath Room} Different meaning.

2. EXPANSION

- Expands abbreviations & acronyms to full form
e.g. Dept → Department.
- Expansion done is based on information from

02.06.2023

__/__/____

user or other sources. Constraints may be imposed to ensure expansion to likely correct.

E.g: constraint - word to be expanded is not in ENG dictionary.

∴ comp → company (possible)

3. STOPWORD REMOVAL & STEMMING

- It is performed to attribute names & domain values
- A domain specific stopword list may be constructed manually.
- Stemming forms the word root or stem. This stem is used in linguistic based matching methods.

4. STANDARDIZATION OF WORDS:

- Irregular words are standardized to a single form
- E.g: using WordNet
  colour → color
  children → child

Q2 Explain web usage mining - Data collection and pre-processing?

Ans ⇒ Huge volumes of clickstream, transaction data and user profile data are collected and analyzing such data helps in various business activities.

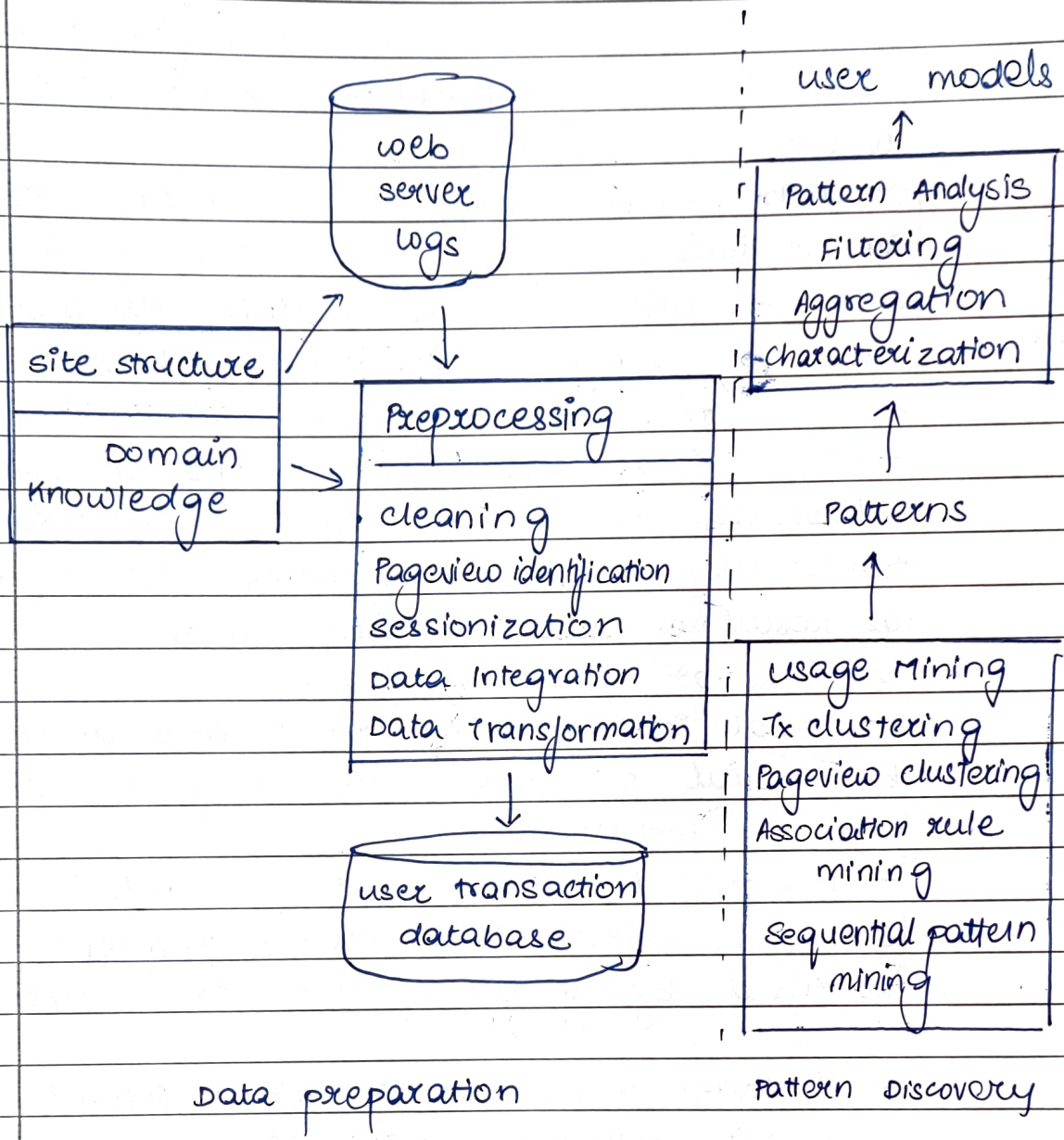
- This type of analysis involves automatic

__/__/___

discovery of patterns and relationships from large collection of semi-structured data often stored in web server logs.

- web usage mining refers to automatic discovery and analysis of patterns in clickstreams & transactions data and other associated data collected as a result of user interaction with web resources on one or more websites.
- Goal is to capture, model and analyze behavorial patterns & profiles in clickstreams, transactions data & other associated data collected as result of users interaction with web resources on one or more web sites.
- Goal is to capture, model and analyze behavorial patterns & profiles of user interacting with the website
- Discovered patterns are represented as collection of pages, objects or resources frequently accessed or used by group of users with common interests.

- 3 stages of web usage mining process :-
  1. Data collection & preprocessing
  2. Pattern Discovery
  3. Pattern Analysis.



1. IN PREPROCESSING : Clickstream data is cleaned and partitioned into set of user transactions representing activities of each user during different visits to sites other sources of knowledge such as site content or structure as well as semantic knowledge from site analysis may be used in preprocessing to enhance data.

__/__/____

2. IN PATTERN DISCOVERY: Statistical database and ML operations are operations performed to obtain hidden patterns reflecting behaviour of users as well as summary statistics on which resources, sessions and users.

3. In final stages of process, discovered patterns and statistics are further processed, filtered, aggregated which can be performed as input to application such as recommendation engines, visualization tools, report generation tools, etc.

SOURCES & TYPES OF DATA

- The primary sources used in web usage mining are the server log files.
- It includes the web server access logs and the application server logs.

1. Usage Data
   - Every HTTP request to the server generates a log record.
   - It identifies fine grained navigational behaviour of users.
   - It includes IP address, url, timestamp, headers, response and if available: client side cookies.
   - It is the primary source of data.

__/__/___

2. Content Data
- It is a combination of textual methods and images.
- It represents the collection of objects conveyed to the user & includes HTML, XML, Multimedia, etc.
- Domain ontologies include hierarchy over sites & page content or categories.
- It has semantic content stored as relationships via ontology languages such as RDF

3. Structure Data
- It represents the designers view of content organization in the site.
- It is captured via interpage linkages through hyperlinks.
- It is available as a 'sitemap' & is stored in XML format.

4. User Data
- It is pseudonymous information about users in the operational transactional database.
- It includes information about various demographics, profile information, reviews, purchases among others.
- Personalization tailored to each user requires storing of user data.