Semantic Approach for Optimal Catch Responses in Web Content Mining

Richlin Selina Jebakumari.A[1] and Dr.Nancy Jasmine Goldena[2]
[1]Research Scholar, Manonmaniam Sundaranar University, Tirunelveli
[2]Assistant Professor, Department of Computer Applications, Sarah Tucker College, Tirunelveli

Abstract
Web surfing generally focuses on query based semantic approach with the expectation of desired results in a speedy way of responses. The role of web content mining is very effective when used in relation to a content database dealing with specific topics. For example searching for a product price in web lead to the online sales domain name, brand of the product, price and offers, which means providing the most specific results of search queries in search engines. This method of allowing only the most relevant information being provided gives a higher quality of results. This increase of productivity is due to direct usage of content mining. Web content mining in education allow for the information provided on their sites to be structured. This allows for a student, faculty, scholar or researcher to access specific information without having to search the entire site. With the use of this type of mining, data remains available through order of relativity to the query, thus providing productive exploration. This paper deals with the semantic approach for providing an optimal catch response system for web mining. In near future the focus will be to implement the neuro fuzzy approaches in real time using neural network conceptual schema.

Keywords: *web mining, catch response, and search engine, semantic, optimality*

I.INTRODUCTION

Web mining is the process of using data mining techniques and algorithms to extract information directly from the Web [1] by extracting it from Web documents and services, Web content, hyperlinks and server logs [3]. The goal of Web mining is to look for patterns in Web data by collecting and analyzing information in order to gain insight into trends, the industry and users in general. Web mining is a branch of data mining concentrating on the World Wide Web [4] as the primary data source, including all of its components from Web content, server [7] logs to everything in between. The contents of data mined from the Web may be a collection of facts that Web pages are meant to contain, and these may consist of text, structured data such as lists and tables, and even images, video and audio[8,10].

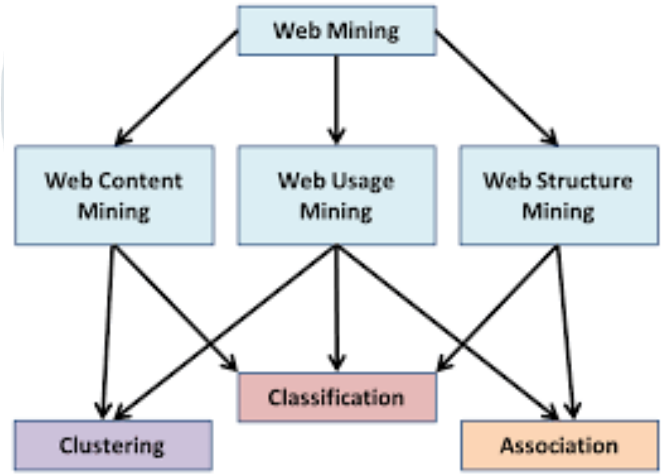

Fig 1: Web Mining Architecture

Categories of Web mining:

- Web content mining — this is the process of mining useful information from the contents of Web pages and Web documents, which are mostly text, images and audio/video files [2]. Techniques used in this discipline have been heavily drawn from natural language processing (NLP) and information retrieval.

- Web structure mining — this is the process of analysing the nodes and connection structure of a website through the use of graph theory [5]. There are two things that can be obtained from this: the structure of a website in terms of how it is connected to other sites and the document structure of the website itself, as to how each page is connected [3, 11].

- Web usage mining — This is the process of extracting patterns and information from server logs to gain insight on user activity including where the users are from, how many clicked what item on the site and the types of activities being done on the site[6,9].

II.PROPOSED METHODOLOGY

The proposed methodology describes the web content of a webpage as the input to this research model emphasizes on the web page resources. Based on the web content category as structured or none structured, the corresponding processing schema will be implemented for the effective extraction approaches. For the non structured content the key-value pair identification is the primary objective for optimal semantic approach towards the speedy response from the server. But for the structured web content, the entire content will be then parsed into several components using the coded programmed and each resultant component is stored in the data mining structure. The text, images and document components are then retrieved for the specified necessity in order to obtain the optimal perfect catch responses in web domain.

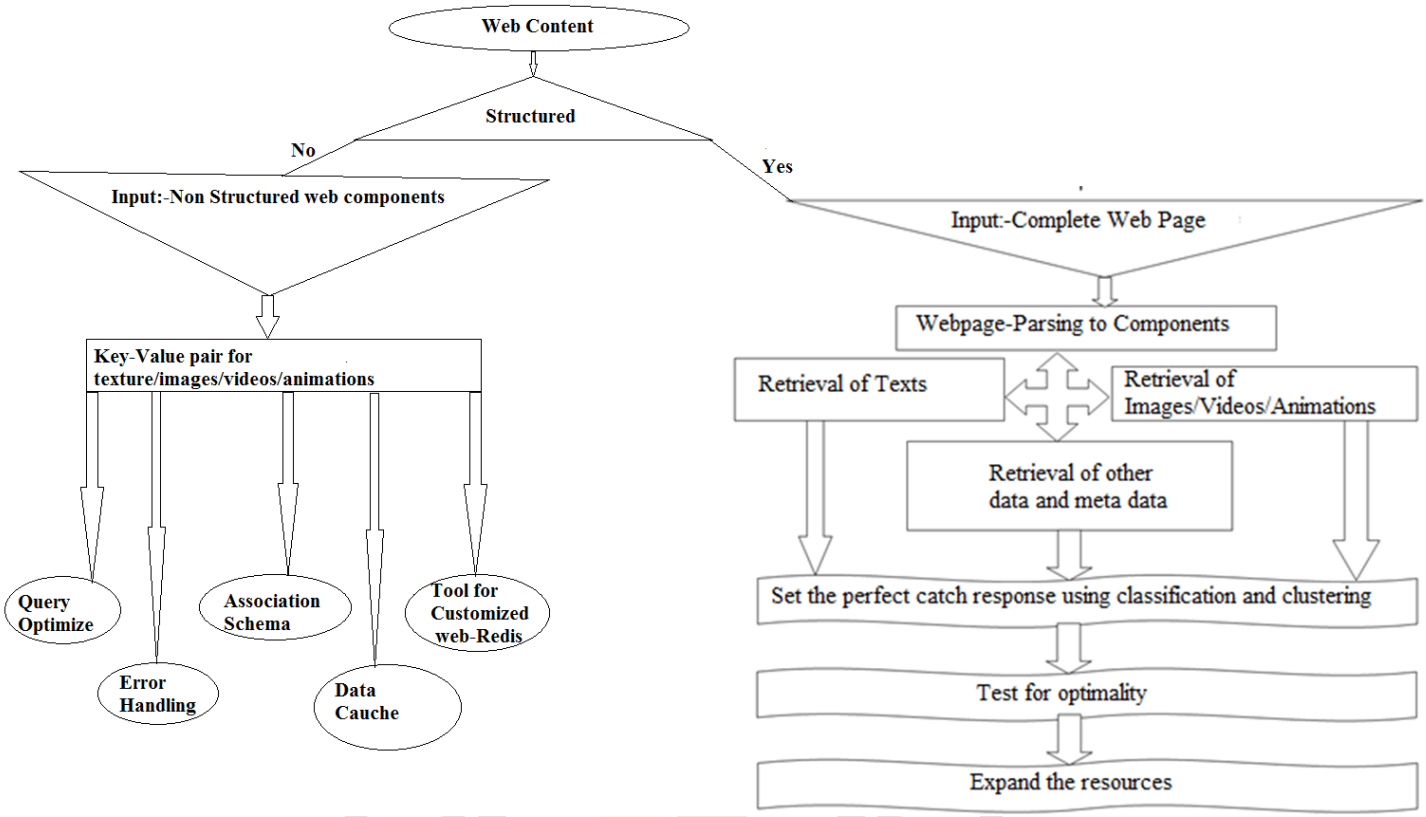

Fig 2: Proposed Catch responses for Web mining

III. IMPLEMENTATION

For the Non structured/unstructured data:

a) Query optimization:

i. Reducing the tuple by selecting the optimal required tuple only.
ii. Reducing the unwanted columns.
iii. Selection operation distributes over the union, intersection, and difference operations
iv. Projection operation distributes over the union operation
v. Union and Intersection are associative
vi. Union and Intersection are commutative
vii. Projection distributes over the Theta Join
viii. Selection operation can be distributed
ix. Join operations are associative.
x. Theta Joins are commutative
xi. Selections on Cartesian Products can be re-written as Theta Joins
xii. All following projections can be omitted, only the first projection is required. This is called a pi-cascade
xiii. Selection is commutative.
xiv. Conjunctive selection operations can be written as a sequence of individual selections. This is called a sigma-cascade.

b) Error Handling:

Application program error checking.
Activity Tracking.

Analyze the traffic.
Handle standard input /output errors
Provide Error/Warning/Info/Verbose/Debug details.

c) Association schema:

Use association rule mining for web data using if-then-rules.

d) Data cache:

Perform data cache for the currently active web databases for faster responses

e) Tools for customized Web-Redis

Redis works to help and improve load performance from relational databases or NoSQL by creating an excellent in-memory cache to reduce access latency. Using Redis one can store cache using SET and GET, besides that Redis also can work with complex type data like Lists, Sets, ordered data structures, and so forth.

The implementation of the web page parsing is done in the basic procedure as follows,
1. Extracting the textual content.
2. Extracting the images in a web page.
3. Extracting the document data and Meta data from the corresponding product pages.

The actual implementation of web content extraction can be utilized by using the following java programming codes.

Steps for Extracting Text content

1. Input the website URL.
2. Use the Java String object for text content.
3. Use Scanner class for identifying the text groups.
4. Collect the texts in a string object by utilizing the append method.
5. Display/store the text collection.

Steps for Extracting Images

1. Using the Octoparse tool to extract the images from website itself.
2. OctoURLextractor is used to extract the URLs of the images

3. Finally using the Google Extension Tab, Save to extract the data required.

Or
1. Using Image downloader extension in Google chrome for downloading all the images in a web site.

For Extracting Content and Meta data from a document:

Using Meta data extractor extension for Google chrome or XPath extractor tool one can extract the Meta data from any website.

Classification and Expansion:
Apply the classification based on the category of textures obtained from the website based on item relation with each other.
Finally expand the website sets with related category of websites using the functionality matching, example all banking sites one by one, all hospital sites one by one, all educational sites one by one and so on, the final result yield the good responses due to the maximum relation of intra dependency and minimum relation of inter dependency among the components matched with the clustering approach.

IV.RESULTS AND DISCUSSION:

In order to implement Redis tool, the initial requirement is to install the open source tool in W7 system with minimum 2 GB of RAM. The commands are user friendly to access the key-value pair for non structured data. Consider the following example for

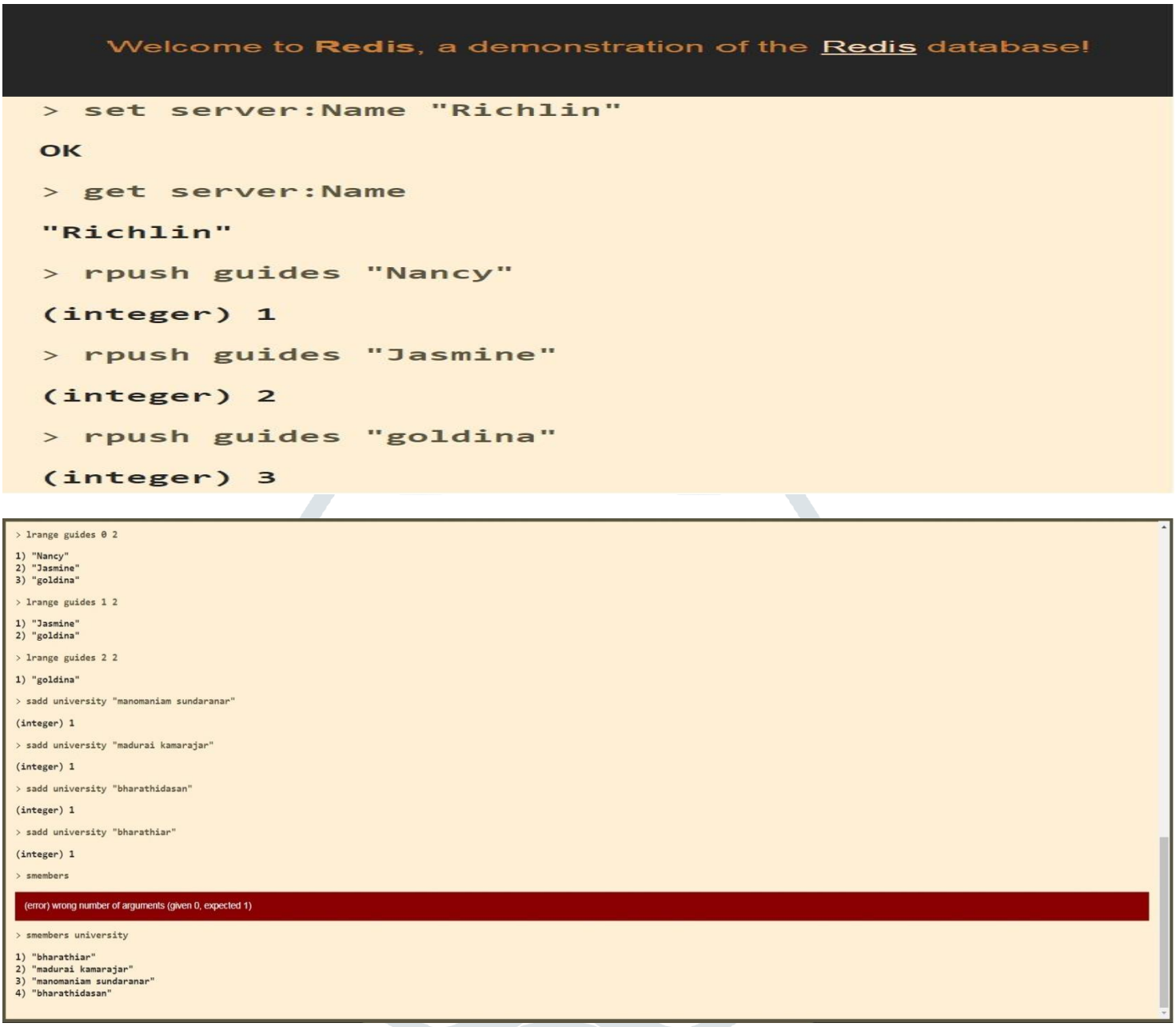

Fig-3: REDIS tool implementation for handling non structured key value pair

The perfect implementation support through Redis for extracting university lists fro wikipedia site is as follows,

```
'use strict';
//Define all dependencies needed
const express = require('express');
const responseTime = require('response-time')
const axios = require('axios');
//Load Express Framework
var app = express();
//Create a middleware that adds a X-Response-Time header to responses.
app.use(responseTime());
const getUniversity = (req, res) => {
  let refno = req.query.refno;
  let url = `https://en.wikipedia.org/wiki/List_of_state_universities_in_India?q=#cite_note-refno:${refno}`;
  axios.get(url)
    .then(response => {
```

```
    let university = response.data.items
    res.send(university);
  })
  .catch(err => {
    res.send('The university you are looking for is not found !!!');
  });
};
app.get('/university', get university);

app.listen(1250, function() {
  console.log('Your node is running on port 1250 !!!')
});
----------WITH REDIS------
'use strict';
//Define all dependencies needed
const express = require('express');
const responseTime = require('response-time')
const axios = require('axios');
const redis = require('redis');
const client = redis.createClient();
//Load Express Framework
var app = express();
//Create a middleware that adds a X-Response-Time header to responses.
app.use(responseTime());
const getuniversity = (req, res) => {
  let refno = req.query.refno;
  let url = ` https://en.wikipedia.org/wiki/List_of_state_universities_in_India?q=#cite_note-refno:${refno}`;
  return axios.get(url)
    .then(response => {
      let university = response.data.items;
      // Set the string-key:refno in the cache. With the contents of the cache : Serial number
      // Set cache expiration to 1 hour (60 minutes)
      client.setex(refno, 1300, JSON.stringify(university));
      res.send(university);
    })
    .catch(err => {
      res.send('The university you are looking for is not found !!!');
    });
};
const getCache = (req, res) => {
  let refno = req.query.refno;
  //Check the cache data from the server redis
  client.get(refno, (err, result) => {
    if (result) {
      res.send(result);
    } else {
      getuniversity(req, res);
    }
  });
}
app.get('/university', getCache);
app.listen(1250, function() {
  console.log('Your node is running on port 1250 !!!')
```

});
The response time using Redis consumes 45 ms whereas non redis implementation produces the results with response time consumes 656 ms which is nearly 16 times faster.

For implementing the Structured content in web data mining, the extraction codes using java plays the vital role for parsing the components as follows,

Consider the State Bank of India Net banking website [13], the implementation of proposed methodology towards this website for text, images and metadata extraction yields the following

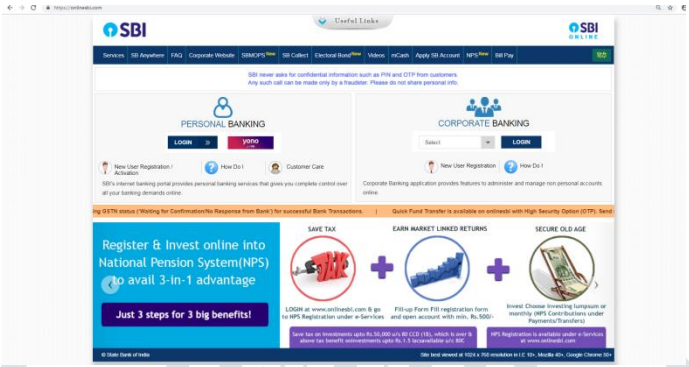

Fig 4: OnlineSBI Website

The following table-1 shows the extracted results for the sample onlinesbi.com web portal.
Table-1:onlineSBI.com web portal extraction

onlineSBI website	Count
Text	47
Images	16
Links	17
Others	8

Sample texts:
Services
SB Anywhere
FAQ
Corporate website
SBMOPS
SB Collect
Electoral Bonds
Videos
MCash
Apply NPS Account
Bill Pay
SBI never asks for confidential INF...

Sample Images

Fig-5: Image samples from onlinesbi.com

The texts based on SB are grouped together; FAQ are grouped together and so on. Similarly the images for logos, information's, buttons etc are grouped together for perfect catch responses under a single set of Keyword associations.

V.CONCLUSION:

Web content is a collection of structured and Non-structured set of websites with its unique implementation. The process of handling and extracting data from non structured data with individual component implementation of query optimization reduces the query complexity, error handling reduces the time complexity, association rule reduces the relational data identification complexity and data cache reduces the replication load complexity and Redis tool reduces the complexity in perfect catch responses, the final result yields 16 times faster than the normal implementation. For structured data sets web data mining is a combination of web mining with the data mining techniques of data filters. Restoring web content is an optimal or an expected form which is a highly technical process to implement in an efficient way. The selection of text, images, document and metadata with the proper utilization of web processing tool is a scientific methodology to implement. This proposed methodology make it as an easy process by the novel view of periodic web data level storage and retrieval combinations, further focusing of their mutual proportion along with variational effects this work achieved an data analysis process with 90 % efficiency. This approach may include many relationships that can be decisive when searching for a query string from the user point of view. The overall method proves to be highly efficient compared to random search theory based approach, dramatically reducing running time and number of features required for the efficient search issues. Moreover, the experimental results revealed that the expressiveness of text, image, and Meta data extraction towards the impact influence cluster representatives is significantly higher than that of normal web mining clustering procedures. In future the focus will be to propose a fuzzy based neural network catch response schema method on web content mining technique.

References

[1] Baraglia, R. Silvestri, F. (2007) "Dynamic personalization of web sites without user intervention", In Communication of the ACM 50(2): 63-67

[2] Cooley, R. Mobasher, B. and Srivastave, J. (2008) “Web Mining: Information and Pattern Discovery on the World Wide Web” In Proceedings of the 9th IEEE International Conference on Tool with Artificial Intelligence

[3] Cooley, R., Mobasher, B. and Srivastava, J. “Data Preparation for Mining World Wide Web Browsing Patterns", Journal of Knowledge and Information System, Vol.1, Issue. 1, pp. 5–32, 2011

[4] Costa, RP and Seco, N. “Hyponymy Extraction and Web Search Behavior Analysis Based On Query Reformulation”, 11th Ibero-American Conference on Artificial Intelligence, 2012 October.

[5] Kohavi, R., Mason, L. and Zheng, Z. (2004) “Lessons and Challenges from Mining Retail E-commerce Data” Machine Learning, Vol 57, pp. 83–113, 2013

[6] Lillian Clark, I-Hsien Ting, Chris Kimble, Peter Wright, Daniel Kudenko (2006)"Combining ethnographic and clickstream data to identify user Web browsing strategies" Journal of Information Research, Vol. 11 No. 2,January 2006

[7] Eirinaki, M., Vazirgiannis, M. (2003) "Web Mining for Web Personalization", ACM Transactions on Internet Technology, Vol.3, No.1, February 2003

[8] Mobasher, B., Cooley, R. and Srivastava, J. (2015) “Automatic Personalization based on web usage Mining "Communications of the ACM, Vol. 43, No.8, pp. 142–151

[9] Mobasher, B., Dai, H., Luo, T. and Nakagawa, M. (2001) “Effective Personalization Based on Association Rule Discover from Web Usage Data” In Proceedings of WIDM 2001, Atlanta, GA, USA, pp. 9–15

[10] Nasraoui O., Petenes C., "Combining Web Usage Mining and Fuzzy Inference for Website Personalization", in Proc. of WebKDD 2003 – KDD Workshop on Web mining as a Premise to Effective and Intelligent Web Applications, Washington DC, August 2013, p. 37

[11] Nasraoui O., Frigui H., Joshi A., and Krishna R., “Mining Web Access Logs Using Relational Competitive Fuzzy Clustering”, Proceedings of the Eighth International Fuzzy Systems Association Congress, Hsinchu, Taiwan, August 2014

[12] Nasraoui O., “World Wide Web Personalization,” Invited chapter in “Encyclopedia of Data Mining and Data Warehousing”, J. Wang, Ed, Idea Group, 2015

[13] http://onlinesbi.com

[14] http://www.redis.io