A semantic preprocessing methodology to improve the recommendations through enhanced preprocessing and feature extraction

Sunitha Cheriyan, PhD Research Scholar,

Madurai Kamaraj University,

Chitra K., PhD

Govt Arts College, Dept. of Computer Science, Melur, Madurai.

Abstract - Today's internet technologies are progressing at rapid speed, resulting in an exponential rise in the number of web pages published. Web page classification is a time-consuming process that is required for identifying and browsing appropriate web pages in response to user searches. The majority of techniques to web page categorization ignore semantic aspects and the context of the page. This research presents a web page classification technique that classifies online pages using semantic features and contextual knowledge. Web sites allow users to collect massive amounts of data. During crawling, browser usage patterns are utilized to identify online consumption resources. By collecting and processing web-related information, log files are used to learn more about user patterns and behavior. The IP address of the visitor, the URL of the website address requested by the end user, and the techniques they used, such as GET or POST, are often saved in web use files on the web server. It also provides details like the protocol version, access date, and time zone utilized to mine the data. It also keeps track of the HTTP status code, the length of each page, and the amount of time a visitor spends on each one.

Index Terms - WUM, Recommendation, Path Completion, Preprocessing, Transaction Identification, Pruning, Web Log Data.

INTRODUCTION

The tremendous increase in the number of web pages in World Wide Web (WWW) have forced e-businesses to become customer or user centric, where the focus is at ensuring user identification and interactions, customization and personalization to increase customer satisfaction, retention and profitability, among other additional business benefits. In today’s competitive environment, the focal point is not only to attract new users but is also on methods to retain existing users. For this purpose, e-businesses analyze details regarding user’s interactions and behavior in order to increase user satisfaction and provide their services in a quick and uncomplicated manner. User details, stored in web log files, have become the most valuable resource for both web masters and designers along with e-concerns.

The WWW has grown very strongly with Internet and broadband users. The evolving nature of web technology has made it possible to capture user interactions with web applications. Process web usage data, extract and convert data stored in web server web log files into knowledge. User activities are stored in a log file that can be used to analyze user movements and understand user intent. Here, pre-processing of received data is unavoidable, as web log data is largely unstructured and may contain redundant files including unwanted noise files. Preprocessing of web log file is mandatory as it contains more irrelevant data which affects analysis of data. Hence analysis of web log file will help prediction mechanisms by serving as a solid back ground to attain precise predictions. It has improved the quality and efficiency of web usage mining. (Dixit, 2010), has reviewed that Data cleansing, data integration, transformation, and data reduction are all part of data preprocessing. Pattern discovery is the process of extracting information from preprocessed data. The few techniques that are applied at this level to preprocess is, data cleaning for removal of unwanted data, filtering of data and integration of data. Several web usage mining activities including detection of the user patterns, analysis of user patterns, cleaning data, identifying sessions, identifying users, completing the navigation path are also discussed here.

LITERATURE REVIEW

Related Work

In (Dixit, 2010), Ms. Dipa Dixit and M Kiruthika (2010) described two ways to data pre-processing, one based on XML and the other on a text file. However, the fundamental algorithm and steps involved in pre-processing are considered same for both the approaches.

In (Ricci, Rokach, & Shapira, 2015) Nehal G. Karelia, Prof. Shweta Shukla 2014, Web usage mining is an essential and difficult research topic. Examining the raw web log files created by the web server is required to determine which user is accessing which sites and with which browser. Data preparation is a crucial step in extracting information from a log.

In (Thakur, Abbas, Beg, & Rizvi, 2015) Bhawesh Kumar Thakur, Syed Qmar Abbas, Mohd Rizwan Beg, Sheenu Rizvi, The improvement of subsequent phases of web usage mining, such as Pattern discovery and Pattern analysis, as well as many data pretreatment techniques such as Data Cleaning, User Identification, Session Identification, and Path Completion, is an important and demanding research area on the web.

In (Rajeswari & Nisha, 2018) , B. Rajeswari, S. Shajun nisha, has conveyed that the web log file contains previous user navigation data or historical data. This web access pattern is used to find the user access behavior.

In (Rajeswari & Sathik, 2020) proposed the possibility of Fuzzy Recommendation by merging case-based totally reasoning (CBR) with ontology, to improve the performance.

In (Sunitha Cheriyan & Chembath, 2017) emphasized the efficiency of ensemble algorithms in terms of accuracy in the prediction and recommendation.

In (S. Cheriyan & Chitra, 2017) analyzed the use of similarity matrix using Markov Model and Bayesian Statistics analyze the users' behavior browsing which can be acquired from web log data.

In (Gaikar Vilas et al., 2021) the study proved the use of statistical methods in the knowledge discovery and improve the user experience in a real life case study.

RESEARCH METHODOLOGY

Web Usage Mining is the process of analyzing web log data to find interesting and useful patterns (WUM). These patterns display user characteristics and are quite useful in the research of how browsers are used and how users interact with the interface. Furthermore, they may be used to investigate users' browsing navigational strategies, which can be utilized to give tailored and relevant information that corresponds to individual users' needs. Apart from helping to raise user interest in a business's offerings, this knowledge may also be utilized to improve the user's surfing experience by anticipating the user's next web page access. These predictions attempt to reduce retrieval time and network bandwidth demand while saving browsing and searching time. Due to the dynamic nature of the WWW and the e-commerce business, Next Web Page Prediction (NWPP) systems are in high demand and deemed tough. This field is undergoing a lot of study, with the goal of developing ways that will increase the accuracy and speed of these systems. Pattern mining, pattern analysis, and preprocessing are all part of the NWPP system. This paper outlines ways for improving prediction accuracy in the future.

The ultimate goal is to create a system that will lead the user to a useful decision based on the logs provided. To support this process, the system will employ data from the collected actions, as well as intrinsic domain and statistical information and any other accessible data. A variety of interconnected research questions arise as a result of this goal:

- How can you know what a user's intent is?
- What features of the dataset may be used to make meaningful predictions?
- How can you figure out what a user prefers?
- Is it true that having a defined goal improves the effectiveness of a recommender system in providing accurate recommendations?

Preprocessing of weblogs

Information about an online user's browsing behavior is stored in web log files. These files are produced on the fly when the browser action occurs, and they come from the web server. Beginning with the request for a web page, the user's activities are logged in the current server log. Finding useful data about how people operate in a timely and exact manner is the most difficult component of web log analysis. By carefully monitoring server logs, we can assess how visitors interact with web sites to arrive at a page where they may obtain all of the data of their search. It will give information on how surfers are progressing toward their targets.

System model

According to (Karelia & Shukla, 2014), the data in the web log contains unnecessary entries, which necessitates processing such data to increase the extraction process' efficiency. export. Data pre-processing's major goal is to minimize the volume of data by removing noisy and irrelevant data, making it simpler to discover user access patterns. Unwanted data, such as images, java scripts, flash animations, and videos, are eliminated during preparation, but the analysis is unaffected. Data elements having similar values that are repeated with isolated record attributes. As described in their paper by (Rajeswari & Nisha, 2018), web log files are processed to eliminate unnecessary data. To make web crawling easy and efficient, non-human access, such as input from search engines (web crawlers and spiders), should be removed.

Cleaning of Weblog Data

The expansion of websites and the information they carry has accompanied the exponential growth of the World Wide Web. All information about how people use the internet is saved in web logs. It records every user click, the path they travelled to get there, and always creates data in the form of photos, failed HTTP status codes, java scripts, applets, audio and video, and more. performed on the web server/log file, which contains all user click streams Unstructured and noisy files predominate in these log files. Preprocessing is, thus, an unavoidable step that must be completed prior to any analysis. With the use of a cleaning or processing algorithm, unrelated and inconsistent files were previously eliminated from the Web log file, creating a strong incentive to examine the aggregated data and generate predictions. The prediction task gets more difficult and complex unless the information to be studied is clear and organized, with just relevant information at hand, which most often leads to incorrect forecasts. Data pretreatment is crucial in web use mining. It's a highly complex operation that accounts for 80% of the entire mining process. Log data is unstructured, redundant, and noisy by nature, according to Bhawesh et al., hence data preparation is essential.

Session identification

A transaction is a group of pages that a user views during a session. Content or backend pages are both possible. Information about the site's content is available on pages dedicated to the user's needs. Add-on pages are sections of a website that help in search navigation. Users often go to the content page through these pages. Based on the efficiency of the backend pages, users concentrate on discovering information. As a result, by identifying these pages, it is required to increase the preprocessing speed. Sessions were determined using MFRA and (Automatic Cutoff Time Estimation Algorithm - ACE Algorithm).

Transaction identification

Users go to websites in order to find the information they're looking for. Users can go through one or more pages at each visit to get the information they want. There may be a requirement for information. The essential information and the algorithm name are provided on a single page. The Transaction Identification Algorithm (TIA) considers the entire session to be a single transaction (Example Purchase of a particular product). It might also be a case of numerous information requirements, in which case more than one page is necessary to deliver the essential information. As a result, a session might be made up of a single page of transactions. A transaction contains some, but not all, of the pages in a session. The "Advanced TIA with pruning algorithm, MFRA, RLA combining completion path," which combines the Transaction Recognition Algorithm (TIA), method reference length approach, and maximum forward reference approach, is used to determine the content and auxiliary pages.

User identification

A user session is a collection of user requests made over the course of a navigation time. In this study, the IP address is employed to identify users. One of the reasons for its popularity is its simplicity. IP addresses are easy to obtain since they are never empty. The technique for identifying the user is completed in the following phases.

- Records with distinct IP addresses are thought to be from two different people.
- In the user agent field, the operating system and navigational browser are investigated for diversity. If they are confirmed to be different, they are acknowledged as two separate users.
- The URL field is checked to make sure it hasn't been used before. If it's a previously unvisited link, it's a valid page.

Transaction Identification Algorithm

A transaction is a group of pages that a user has visited at during a session. These pages might be either content or auxiliary. The focus is on auxiliary pages since people are more concerned with obtaining search information. By finding these pages initially, preprocessing speed can be improved. The solution to the TIA challenges is to create an algorithm that prunes and detects transactions in order to remove irrelevant users.

Transaction Identification Algorithm Boosted with Pruning and path Completion algorithm (TIBP2)

TERMINOLOGIES USED IN THIS ALGORITHM

Let W be the weblog data after cleaning, U1 be the group of people using the session and S1 be the group of sessions,

S1 = {ID, <Address1, ReferAddress1, RDate1>… <Addressi, ReferAddressi, RDatei>}

where ‘i’ denotes the number of transactions in S1 and $1 \leq i \leq I$ (I represents number of valid transactions in the web log data)

A Transaction TS1 consists of a set of pages accessed by the users, that is,

TS1 = {ID, <Address1, RDate1, ReferLen1>,…, <Addressi, RDatei, ReferLeni>}

where ReferLen (reference length) refers to the time spent on page Address.

Let P be the set of content pages of the form <URL1, …, URLi}

Let AP be the set of auxiliary pages of the form <URL1, …, URLi}

whose ReferLen > cutofftime

PRF = Present Transaction’s ReferAddress

FTS = Final Transaction Set

Steps in TIBP2

Step 1: Separate users into relevant and irrelevant users, then choose only the relevant users' sessions.

Step 2: Using the Automatic Cutoff Time Estimation Algorithm - ACE, identify valid pages of individual sessions to fix an estimate of the cutoff time.

Step 3: Using the predicted cutoff time, identify the session.

Step 4: Create transactions using RLA adjusted to use the expected cutoff time in step 2 as well as network transfer rate and data size (Modified Reference Length Approach – MRL).

Step 5: Complete the path for any transactions that aren't complete.

The recommended transaction identification method prunes unnecessary users, then utilizes an automated cutoff methodology to select valid content pages, and then uses a reference length approach and a maximum forward reference algorithm to identify transaction sequences. Because this recommended transaction may be incomplete, a path filling technique is also used to produce optimum transactions.

Grouping users

Based on their level of interest in the web page or site they are viewing, users might be labelled as relevant or non-relevant. The most important users for the company are those who are relevant. Visitors or browsers who have no interest in the site's content are referred to as irrelevant users. The study's users are divided into two groups: relevant and irrelevant users, with the latter group being excluded. There are advantages to removing inactive users. It contributes to the decrease of dataset size and the increase of prediction accuracy. This is accomplished through the use of an integrated K-Means and C4.5 Decision Tree-based Classifier.

SOLUTIONS: Algorithm for rectifying the issues and concerns is given below.

Step 1: Grouping users for Integrated Clustering and Classification System (GIC2) as in Figure 1.

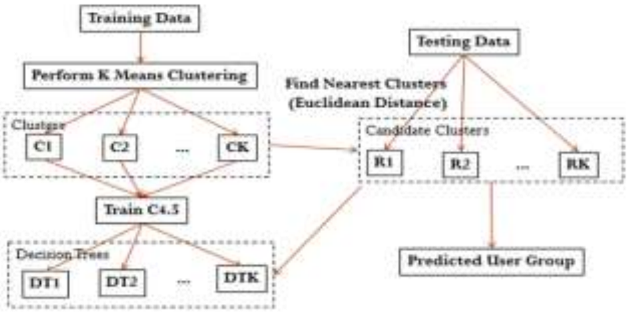


Fig. 1. Grouping users for Integrated Clustering and Classification System

The algorithm suggests combining a clustering algorithm with an existing technique to improve the algorithm's efficacy. Grouping Users for Merged Clustering and Classification System is the name of the algorithm (GIC2). Figure 1 depicts the steps of the suggested method.

Step 2: ACE Algorithm

```
Step 1:
    for i = 1 to n
        if (ReferAddressi == ReferAddressi-1) or ReferAddressi-1 □ C
            Add i to C
    end for
Step 2:  Set CN = size of C
Step 3:  PC (% of Content Pages in S) = (N/CN) *100
Step 4: Set MRF = Mean Reference Length of pages in W (ignore last page as it is content page)
Step 5: CoT = -MRF * ln Pc
```

Step 3: Session Identification

Session identification's objective is to separate the pages that users visit into unique sessions.

Session Identification Algorithm

Let θ1 be the time stamp of the first request, R1.

```
A new session (S) is started at this time (θ1)
Repeat
Let θ2 be the time stamp of the next request, Ri
Add Ri to S
Till θ1 - θ2 < Time Threshold (TT).
```

TT is usually assigned a value of 30 minutes, however for this study, the cutoff time is automatically calculated. Session identification divides the pages that users open into distinct sessions. Session identification is classified as either time-oriented or structure-oriented. The time-oriented method was employed in this investigation.

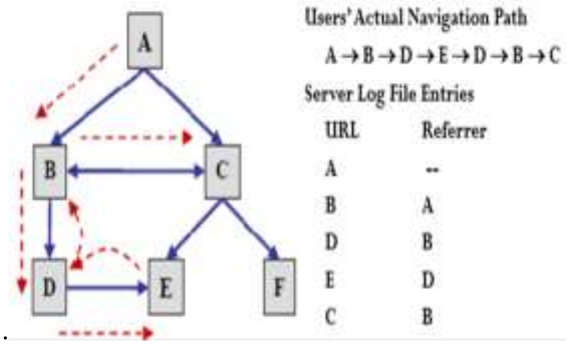
Step 4: MRL Approach

```
Let A be the set of auxiliary pages whose RefLen > CoT
LastSessionID = 0
CRF = NULL; // Current Transaction’s ReferURL
Let FTS = NULL; // FinalTransactionSet
 For i = 1 to n // i □ A
    if SessionID(i) == LastSessionID
            RefLen = -1
            Tr = <SessionID(i), URLi, Date, RefLen>
            Insert Tr into FTS
    end
    CRU = ReferURLi //Current ReferURL
    If CRU == ReferURLi-1 then Insert i to FTS
    if URL(i) == URL(i-1)
            delete i from S
    else
    RefLen = (mod(Date(URL(i))- Date(URL(i-1))) - bytes_sent / c
    Tr = <SessionID(i), URLi, Date, RefLen>
            Insert Tr into TFTS
    end
end for
```

The reference length approach uses the cutoff time obtained in step 3 to identify appropriate transaction page sequences.

Step 5: Path Completion in Incomplete Transactions

After the web log data is classified into sessions, a route completion phase is performed on the discovered transactions to obtain the whole transaction sequence for each session. This made it easier to get all of the user's access information. Path filler strategies are employed when the number of URLs is less than the definite number of URLs visited by the user. Because user access data was absent, Path Completion Algorithms were necessary. Important data was misplaced, resulting in poor prediction results. Incomplete transactions occur when page requests are not recorded (Figure 4.2). Path completion is necessary when the number of URLs is much smaller than the actual number. User access information that is only partially given makes it more difficult to detect browsing patterns, resulting in poor prediction performance. There's a chance that web pages may go missing after making transactions in weblog files. It's critical to find the missing pages in order to learn about users' navigation and browsing habits. A path completion technique is utilized to obtain the entire users' access path in order to increase the system's performance, as shown in Figure 2.



Users' Actual Navigation Path

A→B→D→E→D→B→C

Server Log File Entries

URL	Referrer
A	--
B	A
D	B
E	D
C	B

Fig. 2. Incomplete Transaction

Method of Path Filling

Following the path identification:

1. If any of the URL addresses in the referrer URL address differ from the prior record's URL address.
2. To complete the path, the referrer URL address field of the recent record is added to this session.
3. The length of the references on new attached pages is then decided.
4. Reference lengths of neighbours are modified
5. The lengths of neighbouring pages' references are thus managed.

PERFORMANCE METRICS

The performance of the algorithms was assessed in terms of memory and size. The accuracy, coverage, and F1 measure are used to assess the algorithm's performance. We'll look at how these performance measures are calculated in the next section. The four datasets used to evaluate the TIBP2's performance are listed in Table 1.

TABLE I. DATA SETS USED PERFORMANCE EVALUATION OF TIBP2 ALGORITHM.

Dataset	**TIBP2 performance**			
	Code	***Period***	***Size (MB)***	***No. of Records***
NASA Kennedy Center Space http://ita.ee.lbl.gov/html/contrib/NASA-HTTP.html)	NASA	01-07-1995 to 31-08-1995	205.2	34,61,612
University of Saskatchewan's http://ita.ee.lbl.gov/html/contrib/Sask-HTTP.html	SASK	01-06-1995 to 31-12-1995	233.4	24,08,625
ClarkNet Internet Service Provider http://ita.ee.lbl.gov/html/contrib/ClarkNet-HTTP.html	CN	24-08-1995 to 10-09-1995	171	33,28,587
University of Calgary's, department of Computer Science http://ita.ee.lbl.gov/html/contrib/Calgary-HTTP.html	CL	24-10-1994 to 11-10-1994	52.3	7,26,739

EVALUATION METHOD USED

I. *Effect on the Size of the file*
   i. In terms of memory usage
   ii. In terms of number of entries

II. *Effect of preprocessing on Prediction*
   i. Accuracy
      - Ratio between relevant pages and the summative number of pages in the weblog dataset.

- Accuracy measures the actual degree of accurate pages recommended by a prediction model.

ii. Coverage

- Ratio between relevant pages and the summative number of pages in the user session
- Measures the capability of a prediction algorithm to produce complete page view that the user likely visits.

iii. F1 Measure

- Weighted average of Precision and Recall
- F1 Measure is more than accuracy if distribution of the class is uneven
- The F1 measure reaches its maximum value when both accuracy and coverage are maximized, refer equation 1.

$$Cove \quad (1) \quad \frac{2 x Accuracy \text{ x } \mathrm{Coverage}}{Accuracy + \mathrm{Coverage}}$$

EXPERIMENTAL RESULTS

I. Effect *on the Size of the file*

Effect of cleaning transactions

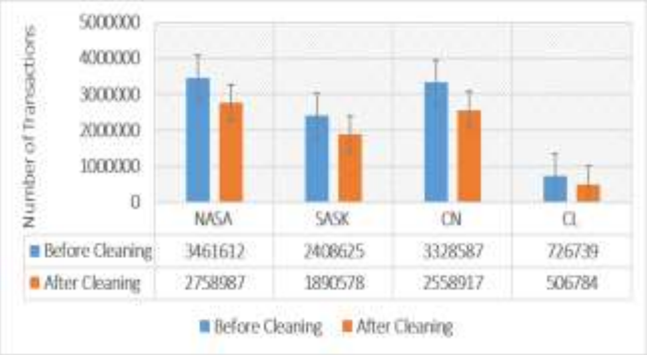


Fig. 3. Effect of cleaning transactions

Figure 3 shows how well the cleaning algorithm works. According to the results, the cleaning approach decreased the number of entries in the raw web log data in all four datasets.

Effect of cleaning memory size

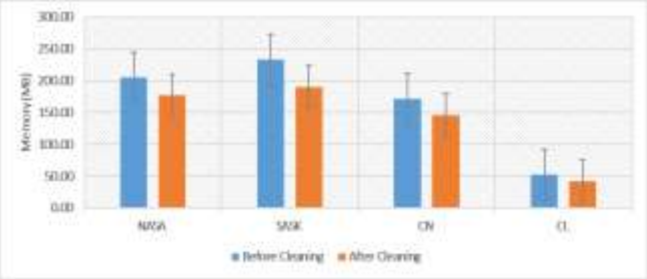


Fig. 4. Effect of cleaning memory size

As shown in Figure 4, the cleaning operation reduced the web log file's memory size.

Effect of removing irrelevant users in terms of number of transactions

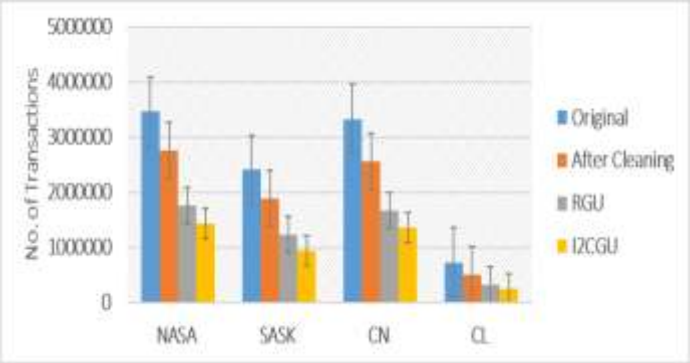


Fig. 5. Effect of removing irrelevant users in terms of number of transactions

By eliminating items that were not relevant input to the prediction system, the number of transactions linked with the prediction system was decreased, as shown in Figure 5. The original file, RGU (rule-based technique for grouping users), and the file after cleaning had all performed worse than I2CGU.

Effect of removing irrelevant users in terms of storage size

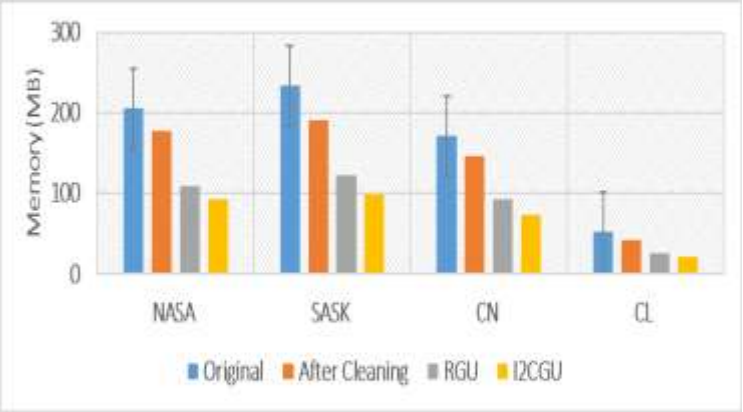


Fig. 6. Effect of removing irrelevant users in terms of storage size

As illustrated in Figure 6, the suggested integrated algorithm I2CGU improved the efficiency of recognizing relevant and irrelevant users in terms of storage space by requiring the least amount of time. This benefited the prediction mechanism. By comparing it to the prior RGU algorithm, the result was established. All four datasets showed the same tendency.

II. Effect of preprocessing on Prediction

Accuracy

i. Effect of user grouping algorithms in terms of Accuracy(%)

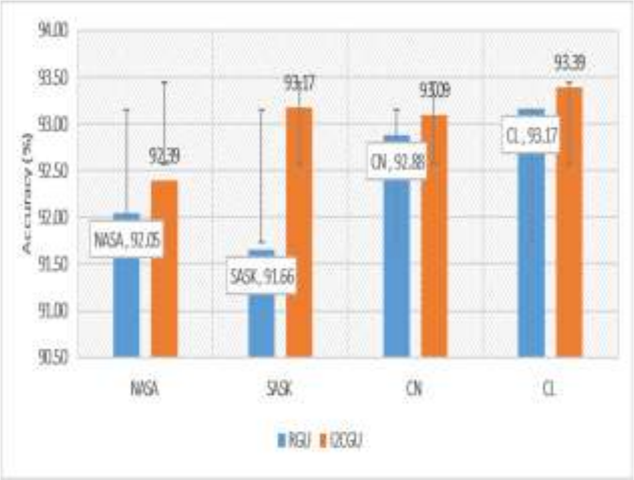


Fig. 7. Effect of user grouping algorithms

In comparison to the previous algorithm RGU, the suggested integrated algorithm boosted the accuracy aspect when users were grouped for detecting relevant and irrelevant users, as shown in Figure 7.

ii. Effect of preprocessing algorithms on prediction

Jalali et al. (2010) developed a model to assess the suggested preprocessing method. The Longest Common Subsequence was employed in this algorithm's prediction. The LPA algorithm has the following phases when it comes to prediction.

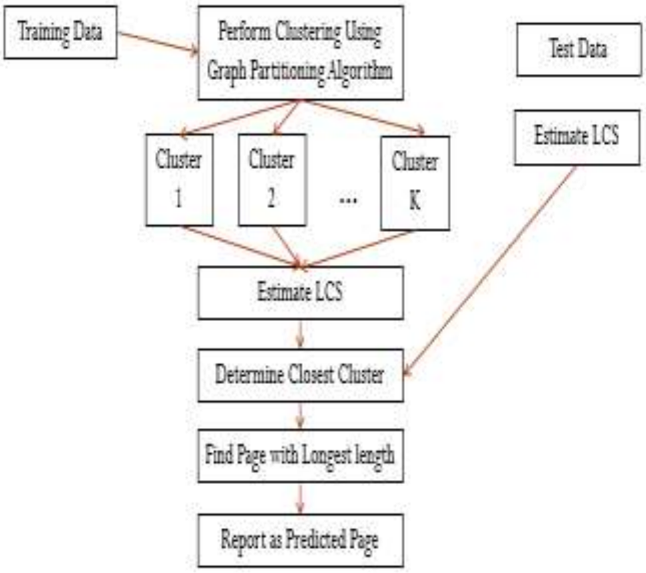

Training Data
Perform Clustering Using Graph Partitioning Algorithm
Test Data
Cluster 1
Cluster 2
...
Cluster K
Estimate LCS
Estimate LCS
Determine Closest Cluster
Find Page with Longest length
Report as Predicted Page

Fig. 8. Effect of preprocessing algorithms on prediction

The model presented by Jalali et al.(Jalali, Mustapha, Sulaiman, & Mamat, 2010) was utilised to assess the influence of the proposed preprocessing methods on page prediction. During prediction, this algorithm used the LCS (Longest Common Subsequence). As shown in Figure 8, the LPA preprocessing method comprises of the steps listed below.

TABLE II. CODING SCHEME

Code	Description
LPA	LCS-BASED PREDICTION ALGORITHM
LPA-C	LPA WITH CLEANING
LPA-RLA	LPA WITH TIA BASED ON RLA
LPA-MFRA	LPA WITH TIA BASED ON MFRA
TIBP2 –WOPF	TIBP2 WITHOUT PATH FILLING
TIBP2	TIBP2 WITH PATH FILLING

TABLE III. EFFECT OF PREPROCESSING ALGORITHM ON PREDICTION IN TERMS OF ACCURACY

Prediction Model	NASA	SASK	CN	CL
LPA	85.62	84.79	86.25	86.94
LPA-C	86.23	84.94	86.77	87.02
LPA-RLA	87.58	86.28	87.48	87.94
LPA-MFRA	86.72	85.69	87.40	87.76
TIBP2 – WOPF	88.63	87.32	88.76	89.08
TIBP2	**90.35**	**88.26**	**89.45**	**90.49**

From the table III, it is obvious that the preprocessing operations has also increased the accuracy of prediction.

Coverage

TABLE IV. EFFECT OF PREPROCESSING ALGORITHM ON PREDICTION IN TERMS OF COVERAGE

Prediction Model	NASA	SASK	CN	CL
LPA	1.6325	1.6518	1.6163	1.6068
LPA-C	1.4899	1.5048	1.4904	1.4557
LPA-RLA	1.3743	1.3684	1.3478	1.3089
LPA-MFRA	1.3959	1.4432	1.4310	1.3871
TIBP2 – WOPF	1.3049	1.2996	1.2617	1.2172
TIBP2	**1.2184**	**1.2169**	**1.1897**	**1.1302**

This results in table IV further confirms the effect of preprocessing algorithms in terms of coverage performance metric also.

F1 Measure

TABLE V. Effect of Preprocessing Algorithm on Prediction - F1 Measure

Prediction Model	NASA	SASK	CN	CL
LPA	0.8241	0.8340	0.8158	0.8109
LPA-C	0.7514	0.7591	0.7517	0.7340
LPA-RLA	0.6926	0.6897	0.6791	0.6593
LPA-MFRA	0.7036	0.7277	0.7214	0.6991
TIBP2 – WOPF	0.6573	0.6547	0.6354	0.6128
TIBP2	**0.6133**	**0.6127**	**0.5988**	**0.5686**

It can be deduced from the F1 measure values in Table V that the preprocessing operations enhanced the overall prediction performance.

SUMMARY AND CONCLUSION

The preprocessing of data is an important phase in the web mining process. Web log analysis is critical for finding and forecasting user movements. It is possible to make subsequent prediction procedures easy and effective after evaluating web log files. When the clustering stage is completed before the prediction, sessions can be divided into a number of comparable groups. The creation of competitive forecasting models is aided at this stage. In order to increase forecast reliability, it helped to decrease the complexity of decision-making and to lessen the scalability issue. Data mining is a fascinating career that entails extracting information from large amounts of data.

REFERENCES

1. Cheriyan, S., & Chitra, K. (2017). Web page prediction using Markov model and Bayesian statistics. *Proceedings of the 2017 2nd IEEE International Conference on Electrical, Computer and Communication Technologies, ICECCT 2017*. https://doi.org/10.1109/ICECCT.2017.8117864
2. Cheriyan, Sunitha, & Chembath, J. (2017). Comprehensible Predictive System Model using Parameter less Fast K-Means (EPFK-Means) for web usage data. *Procedia Computer Science*, *115*, 243–250. https://doi.org/10.1016/j.procs.2017.09.131
3. Dixit, M. D. (2010). Preprocessing of web logs. *International Journal*, *02*(07), 2447–2452.
4. Gaikar Vilas, B., Joshi Bharat, M., Jaywant, B., Mhatre, N. S., Chitra, K., Cheriyan, S., & Rane Caroleena, G. (2021). An Impact Of Covid-19 On Virtual Learning: The Innovative Study On Undergraduate Students Of Mumbai Metropolitan Region. *Academy of Strategic Management Journal*, *20*(SpecialIssue2), 1–19.
5. Jalali, M., Mustapha, N., Sulaiman, M. N., & Mamat, A. (2010). WebPUM: A Web-based recommendation system to predict user future movements. *Expert Systems with Applications*, *37*(9), 6201–6212. https://doi.org/10.1016/j.eswa.2010.02.105
6. Karelia, N. G., & Shukla, P. S. (2014). *Data Preprocessing: A Pre requisite for Web Log Files*. *3*(4), 1571–1574.
7. Rajeswari, B., & Nisha, S. S. (2018). *Web Page Prediction Using Web Mining*. (x), 4234–4237.
8. Rajeswari, B., & Sathik, M. M. (2020). Web Recommendation System: A Systematic Survey. *International Journal of Engineering Research and Technology (IJERT)*, *8*(3), 1–3. Retrieved from www.ijert.org
9. Ricci, F., Rokach, L., & Shapira, B. (2015). *Recommender systems handbook*. https://doi.org/10.1007/978-1-4899-7637-6
10. Thakur, B. K., Abbas, S. Q., Beg, M. R., & Rizvi, S. (2015). *Automated Tool For Web User Identification*. *3*(4), 75–80.