International Journal of Computer Sciences and Engineering
Vol.**12**, Issue.**6**, pp.**21-29**, June **2024**
ISSN: 2347-2693 (Online)
Available online at: www.ijcseonline.org

ISSN: 2347-2693 (E)

Research Article

Extraction of Sequential Patterns Using PREFIXSPAN

Elliot S.J.[1*], **Bennett E.O.**[2]

[1,2]Department of Computer Science, Rivers State University, Port Harcourt, Nigeria

**Corresponding Author: sobestman2@gmail.com*

Received: 16/Apr/2024; **Accepted**: 18/May/2024; **Published**: 30/Jun/2024. **DOI**: https://doi.org/10.26438/ijcse/v12i6.2129

Abstract: A great number of individuals are anxious to exploit the internet's wealth of information. It can be employed to further enhance the existing data. However, the primary challenge lies in uncovering the valuable information that is concealed within HTML elements. This study proposes a framework for web usage mining that examines web server log files using sequential pattern mining approaches. Web log patterns reveal information about user behavior, preferences, and website interactions. Preprocessing of the web data was carried out. The primary objective of preprocessing is to enhance data integrity while decreasing the volume of information that requires evaluation. Prior to inputting the data into the pattern discovery phase, it is necessary to eliminate noise by resolving the challenge of distinguishing between different users and sessions. To identify frequent sequential access in large, low-support data sets, a method for mining sequential patterns is developed. A sequential pattern mining technique identifies recurring sequential patterns in multidimensional web log files with minimum support provided. Multidimensional sequential pattern mining is primarily concerned with enhancing the standard of the patterns the user received back. PrefixSpan algorithm has been used to extract tabular as well as unstructured data from HTML tag. Prefix prunes some web info by calculating the support value at different nodes in the represented projected sub-database and snipe away huge portions of the representation that are guaranteed not to create any outcomes. The system is implemented in Matlab programming language. In the domain of web mining, Matlab has been employed to extract valuable information from the web, including user records and content. When mining extensive sequences containing numerous records, in particular, the method substantially reduces execution time and eliminates enormous memory access costs. The PrefixSpan algorithm enhanced with the starting position and innertagcount parameters has better performance than Markov model and GSP algorithm with execution time of 2.35seconds.

Keywords: Web Usage Mining, Sequential Patterns, Web Access Pattern, Prefixspan, Web Server logs, Preprocessing.

1. Introduction

It is impossible to ignore the exponential growth of data on a worldwide level as time passes. There is a growing dependence in our daily existence on the 2.5 quintillion bytes of data that are generated. As of 2018, 90 percent of the global data volume had already been produced, and projections indicated that by 2020, the average individual would generate 1.7 gigabytes of data per second. The amount of data produced has increased dramatically as a result of society's digitization and the quick advancement of technologies for data collection and storage.Descriptive of our era thus far, the data and information age, or simply the digital age, is appropriate [1].

As the largest repository of information, the World Wide Web significantly affects the daily lives of the majority of people. It facilitates the dissemination of knowledge and improves the accessibility of critical information. The World Wide Web has evolved from an esoteric academic concept to a critical resource for business, marketing, and communication, in addition to serving as a virtual community, in less than two decades [2]. The utilization of web mining is increasingly becoming essential due to the proliferation of unprocessed data and the pressing requirement to extract valuable insights from it.

As the volume of information expands at an exponential rate, ensuring the quality and precision of data becomes a critical challenge. The increased demand for specialized data collection and administration platforms that can facilitate advanced research has resulted from this development. Web mining is rapidly emerging as the preferred technique for extracting enormous quantities of data from the internet. The considerable amount of data that is currently accessible has garnered considerable interest from both the information industry and the general public due to its critical nature. The applications for which web mining yields insights are diverse and encompass a broad spectrum of fields. As an illustration, they can be implemented in enterprise context-aware advertising, database development, business intelligence, and social web applications [3], which may involve the extraction of data from online social platforms.

In situations where objects outside of a cluster exhibit dissimilarities while objects within the cluster share similarities, clustering algorithms can identify these groups and subsequently group the objects. However, heuristics are required because the development of clustering algorithms can be difficult due to the potential for substantial memory or processing requirements. Sequential pattern mining is a popular technique for web usage mining [4] due to its ability to process crawling-induced profiles.

The main focus of the study is to extract web content using the PrefixSpan algorithm in order to conduct sequential pattern analysis. To identify patterns present in the expected data, PrefixSpan performs pattern mining on the complete collection, which significantly reduces the effort and time required to generate candidate subsequences. Additionally, The size of projected databases is decreased using prefix-projection, resulting in more efficient processing. The algorithm selected for large item sets was PrefixSpan, due to its innovative strategy that enhances mining speed and efficiency. By examining a web-derived dataset, this research endeavors to resolve the methodological enigmas associated with the conception, retrieval, and interpretation of website data for the purpose of understanding enterprise research and development initiatives and business innovation strategies. The paper is organised into 5 sections. Section one is the introduction of web mining. Section 2 is related literature on different algorithms of Web Usage Mining . In Section 3 methodology and architecture of the system was discussed. Section 4 is the implementation and result. Finally, in section 6 is the conclusion.

2. Related Literature

In order to expedite the process of recognizing sequential patterns, [5] introduced the SPADE algorithm. Present methods for addressing this concern depend on intricate hash structures that exhibit limited locality and necessitate frequent database scans. In order to solve the original problem, employing effective lattice search methods and simple join procedures in main memory, SPADE divides it into smaller sub-problems using combinatorial features. Three database searches were sufficient to locate every sequence.

The FreeSpan method [6] utilizes annotations produced by projected databases in order to accelerate the mining procedure and identify prevalent patterns. Its anticipated database has a considerably lower contraction factor than PrefixSpan's.

PrefixSpan, as delineated in reference [7], operates by scanning original sequence database once in order to produce the projected and original databases. The lexicographic order is maintained and is of considerable importance. Split-and-Project, despite its usefulness, requires a significant amount of sufficient memory to hold each projections, particularly when implementing recursion. Pseudo-projections were proposed as a potential resolution to this dilemma. Later, algorithms including SPARSE [8] and LAPIN_Suffix [9] were developed based on this. In this instance, the sequence database is maintained in memory, and instead of storing the projected database in memory, the locations of various projections are specified using a pointer and an offset. Utilizing bi-level projection, as exemplified by FreeSpan and PrefixSpan, offers an additional expedited alternative subsequent to the establishment of the projected database.

By placing particular emphasis on the preprocessing stage, [10] implemented WUM in the field of e-learning. When considering e-learning, they reconsidered the notion of a visit to this particular location. Their method permits a learning session to span many days if that duration relates to a particular learning time. A sequence of procedures implemented to accomplish a task could potentially be linked to an educational session. The authors identified comparable episodes to [11] by categorizing web pages into three distinct groups—resource, auxiliary and content —using information already known about the website.

In addition, [12] established a data warehouse for the storage of web traffic files. Their methodology does not incorporate structured data pertaining to usage (including sessions and visits), users, or aggregated variables. They exerted effort in pursuit of the objective of web archiving. Every individual request documented in the web archives is of the utmost importance and must be retained by web caching systems. The authors utilize the Internet Protocol (IP) heuristic to identify a user during a session.

In order to assist enterprises in discovering useful knowledge inside network information resources, [13] integrates online data mining technologies with an e-service platform, drawing upon an analysis of web data mining ideas and applications. E-commerce businesses can acquire new clients and retain existing ones by using this predictive power to identify customer trends. Correct decision-making increases an enterprise's ability to compete.

2.1 Sequential Pattern

Sequential patterns are frequent patterns wherein a significant number of input sequences from a single transaction or more concurrent transactions coexist. SPM is utilized to partition sequential patterns whose assistance surpasses a predetermined minimal assistance threshold. Sequential pattern mining additionally aids in the elimination of configurations that mirror the most prevalent practices within the succession database. Such configurations may, for certain rationales, be regarded as space information.

Only a limited number of fields employ Sequential Pattern Mining (SPM). SPM is frequently implemented by business organizations when considering consumer behavior. SPM is additionally employed in computational biology to investigate patterns of corrosion of amino acids. Web usage mining also employs SPM to extract information from a limited quantity of web records transmitted by multiple hosts. SPM algorithms may be categorized as Apriori-based, Pattern-development, Early-pruning, or a hybrid approach combining all three. Apriori-based methodologies commonly

incorporate breadth-first search, create-and-test, and diverse database outputs. These approaches often present difficulties in terms of testing and cause disruptions to the algorithms' visual representation. Pattern-development algorithms have demonstrated their speed through rigorous testing in the domain of web log mining. Early-pruning algorithms have even managed to succeed when confronted with protein arrangements stored in dense databases. In contrast, apriori-based algorithms have been identified as exceedingly sluggish and requiring an extensive search space. By utilizing the Java Hash Map data structure, the strategy performs a breadth-first search.

In the contemporary era characterized by technological advancements, each organization possesses its own website, facilitated by either proprietary or corporate web servers, and employs an intelligent correspondence cycle. The information provided by a user who interacts with this website in order to locate correspondence pertaining to specific assets is recorded. Furthermore, this feature offers the most straightforward approach for the website administrator to analyze the information derived from the web access records concerning the client's navigational patterns. Additionally, it can be advantageous to analyze the client's access behavior in order to ascertain the most suitable page to display to them according to those observed patterns. The sequential pattern technique is advantageous for this reason. By employing the sequential pattern method, patterns in sequential databases are identified. Comparing the designed patterns to the baseline assistance measure, the researchers in this study proposed an algorithm for identifying sequential patterns that may be useful for recommendation generation [14].

2.2 Web Usage Mining (WUM)

WUM is a subfield of web mining that is responsible for the administration of knowledge extraction from log data produced by web servers, as stated in reference [15]. The utilization of diverse data mining methodologies to analyze online usage data enables the segregation of client access patterns, a process that yields numerous benefits including business intelligence, website customization, system optimization, and more. In order to discern patterns in web usage extraction, data reflection is fundamental. The process of data preparation is what enables this data reflection to occur. They gained an understanding of data preprocessing duties, such as data cleansing and reduction, as well as the algorithms involved.

Although web analysis tools are available to report server user activity and filter data in various ways, their functionality is primarily designed for servers experiencing moderate to low traffic. Furthermore, these tools do not provide extensive analysis capabilities regarding the relationships between files and directories accessed via the web. Conversely, emerging techniques that are more advanced in nature are poised to detect and assess patterns. In essence, two categories of these instruments exist:

a) Pattern Discovery Tool: In order to extract insights from collected data, new user pattern discovery tools employ sophisticated techniques such as information theory, automation, data mining, and data mining.

b) Tools for Pattern Analysis: To effectively analyze, visualize, and interpret identified access patterns, analysts necessitate appropriate tools and methodologies.

2.2.1 Process of Web-Usage-Mining

Utilizing web mining for data mining purposes involves the use of various data mining methodologies. It analyzes patterns in web data usage to gain a deeper understanding of and meet the demands of internet users. Web utilization mining follows a three-stage process, just like any other data mining task:

(a) “Pre-processing”
(b) “Pattern Discovery”
(c) “Pattern Analysis”.

The preprocessing procedure comprises three discrete stages: data cleansing, user identification, and session identification. Pattern discovery in the following study, involves the implementation of recently developed techniques to identify sequential patterns in log data. As the transformed data is required as input for the algorithms, data conversion is a crucial component of the pre-processing phase. Deciphering and deriving conclusions from the algorithmic output constitutes pattern analysis.The phase of pattern discovery involves the identification of principles or patterns. The pattern analysis objective is to eliminate those that are uninspiring. The purpose of web mining generally dictates the analysis procedure. The majority of pattern analysis is conducted using visualization methods. We can assign colors to values or graph patterns in this manner. This will frequently reveal overarching patterns or trends within the data.

In Exploring the Landscape of Web Data Mining. Through the presentation of the most recent advancement and the provision of web mining techniques on web data, both practitioners and researchers can get insight to current state of the field and pinpoint possible area requiring more investigation.

An image search engine called iFind was created by the researchers, and it works better than conventional methods. Numerous search possibilities are supported by iFind, such as log mining, relevance feedback, query by example, and keyword-based search. [16]. Refer [17] suggested an algorithm for web usage mining involving data pre-processing and personalisation. The recommended algorithm concentrated on gathering data fields, cleaning data, and finding patterns in log files.For data processing, the field extractor technique was employed, and for data cleaning, parameters like file_ext, status code, and method was used. As proposed, these algorithms would improve webpage performance in terms of speed and accuracy, reduce memory usage, and personalize information based on user requirements.

2.3. **PrefixSpan Algorithm**
The anticipated prefix employs the sequential pattern mining technique known as PrefixSpan, it becomes feasible to identify the frequent items by scanning the sequence database. This strategy employs a pattern-growth methodology. The primary database is partitioned into multiple smaller databases based on the commonly accessed objects. Finally, each projected database encompasses the complete assortment of sequential patterns, acquired through the iterative expansion of subsequence pieces. The PrefixSpan technique employs a divide-and-conquer approach to identify patterns. However, this method may generate and manage numerous projected sub-databases, resulting in a substantial memory overhead.
The PrefixSpan method is responsible for identifying sequential patterns in sequence databases.

Here are few characteristics of the PrefixSpan Algorithm:
i. Candidate Sequence Pruning
ii. Sampling and/or Compression
iii. Tree Projection
iv. Suffix/Prefix Growth
v. Search Space Partitioning
vi. Memory-Only
vii. Depth-First Traversal

As PrefixSpan scans the original sequence database, it generates the projected databases concurrently. The lexicographic order is maintained and is of considerable importance. Split-and-Project, despite its usefulness, requires a significant amount of memory to store all the projections, especially when implementing recursion. Pseudo-projections were proposed as a potential resolution to this dilemma. Later, algorithms including SPARSE and LAPIN_Suffix were derived from this. In this instance, the sequence database is maintained in memory, and instead of storing the projected database in memory, the locations of various projections are specified using a pointer and an offset. Utilizing bi-level projection, as exemplified by FreeSpan and PrefixSpan, offers an additional expedited alternative subsequent to the establishment of the projected database.

3. Methodology

Constructive research and Object Oriented Design Analysis (OODA) were used in the new construct. The Prefixspan Algorithm, representing the pattern growth process, was employed. The sequential pattern mining was implemented in the Matlab programming language.

System Design
The system utilizes Prefix techniques to identify sequential patterns. Figure 1 illustrates the various components of the system.

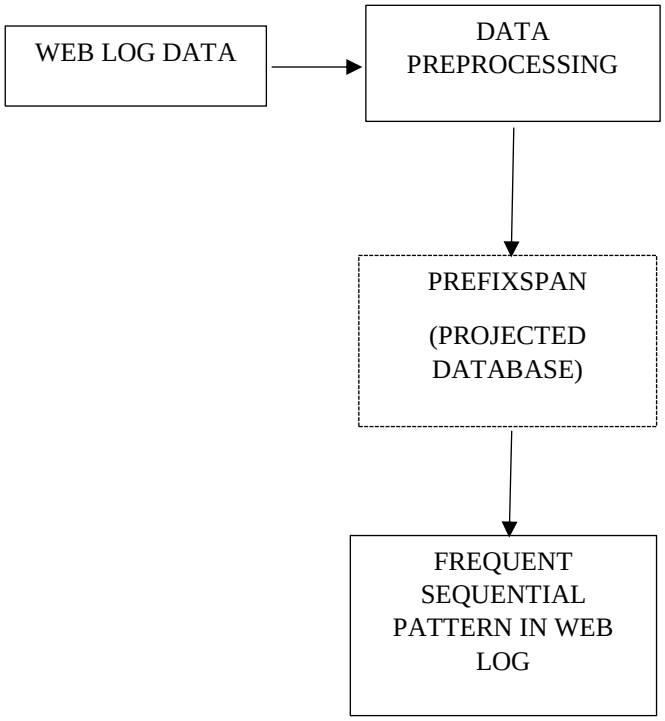

Figure 1: Architecture of the System

3.1. Extraction of Sequential Patterns Using Prefixspan
The PrefixSpan algorithm is an SPM algorithm used in mining of data and pattern recognition. It is particularly useful for finding frequent sequential patterns in sequences or sequences of sets.

The Prefixscan Algorithm Steps:
1. Initialize with an empty prefix and the entire sequence database.
2. For each item frequent in the current sequence database:
3. Extend the current prefix with the items that are frequent in-order to create a new prefix.
4. Project the database onto the new prefix to create a projected database.
5. Recursively mine the projected database for frequent patterns.
6. Continue till no other frequent patterns are found.

The PrefixSpan algorithm operates as follows:
1. Understanding the concept of prefixes in sequences forms the basis of the algorithm. Prefixes are subsequences that are found at the beginning of a sequence. For instance, sequence {A, B, C, D} includes the prefixes {A} and {A, B}.

2. Mining Sequential Patterns: Through a systematic exploration of the dataset in search of frequently occurring prefixes, PrefixSpan is capable of extracting recurring sequential patterns.

3. Frequent Pattern Growth: PrefixSpan, as opposed to alternative depth-first search sequential pattern mining algorithms, integrates both approaches into a unified frequent pattern growth strategy. By focusing on patterns that are

anticipated to manifest frequently, this approach effectively eliminates superfluous queries.
To illustrate this,

Suppose we have the following sequence database:
1. “Sequence 1”: {A, B, C}
2. “Sequence 2”: {A, B, D, E}
3. “Sequence 3”: {A, C}
4. “Sequence 4”: {B, C, D}

Let's assume we want to extract patterns that occur in multiple sequences and have a minimum support threshold of 2, just for the sake of discussion.

I. Initialization:
Start with an empty prefix.
Initialize the sequence database with all sequences.

II. Mining Process:
Start with an empty prefix and the entire sequence database.
Find frequent items in the current database:
{A}: Appearing in sequences 1, 2, and 3.
{B}: Appearing in sequences 1, 2, and 4.
{C}: Appearing in sequences 1, 3, and 4.
{D}: Appearing in sequences 2 and 4.
Mine frequent patterns:
Prefix {A}:
Project database onto {A} to get:
“Sequence 1”: {B, C}
“Sequence 2”: {B, D, E}
“Sequence 3”: {C}
Recursively mine projected databases:
Frequent patterns:{B}, {C} (both have support >= 2)
Prefix {B}:
Project database onto {B} to get:
Sequence 1'': {C}
Sequence 2'': {D, E}
Recursively mine projected databases:
Frequent patterns:{C},{D}
(both have support >= 2)
Prefix {C}:
Project database onto {C} to get:
Sequence 1'': {B}
Sequence 3'': {}
Recursively mine projected databases:
Frequent patterns: {B} (support >= 2)
Prefix {B}:
Project database onto {B} to get:
“Sequence 1”: {C}
“Sequence 2”: {D, E}
“Sequence 4”: {C, D}
Recursively mine projected databases:
Frequent patterns: {C}, {D}(both have support >= 2)
Prefix {C}:
Project database onto {C} to get:
Sequence 1': {B}
Sequence 3': {}
Sequence 4': {B, D}
Recursively mine projected databases:
Frequent patterns: {B} (support >= 2)
Prefix {D}:
Project database onto {D} to get:
Sequence 2': {E}
Sequence 4': {C}
Recursively mine projected databases:
Frequent patterns: {E}, {C} (both have support >= 2)
The mining procedure keeps going until frequent patterns are no longer detected.

III. Result:
Frequent sequential patterns with support >= 2:
{A}, {B}, {C}, {D}, {A, B}, {A, C}, {B, C}, {B, D}, {C, D}
The PrefixSpan algorithm efficiently mines frequent sequential patterns within the given sequence database based on the specified minimum support threshold.

Input Specification of the System
By the analysis of web page content, web archives can be extracted. A collection of web logs extracted from the papers presented at the NIPS 2015 conference will be utilized for this investigation. The input variables are arranged in the following manner, as shown in Table 1: host IP address, papers, authors, affiliates, and coauthors; request date and time; and Uniform Resource Locators (URL) status bytes.

Table 1: Input Variables

Input Variables	Description
10.2.1.40—(30/aug/2022:9:15:12+0530)	Date and Time
http://WWW.nipapers.com/favicon.icoHTTP?!>! 302460	URL
TCP_MISS:FIRST_UP_PARENT	Host IP
Article Name	Deep learning, Reinforcement learning, Neural network
Authors	Samy Bengio, Danilo Rezende, Bill Dally
Coauthors	Nihar Shah, Dengyong Zhou, Jonathan Vacher
Affiliation	NICTA, UC Berkeley, MSR

Processing Design

1. Data Preprocessing
Preparing unprocessed data entails a process that aims to produce a pristine data set. The algorithm performs preprocessing on the dataset to remove any missing values, noise, or other inconsistencies. Utilizing unprocessed web log data for pattern mining presents challenges due to its frequently diverse, fragmented, and poorly organized nature. Its processing requires adherence to the procedures delineated in Figure 2.

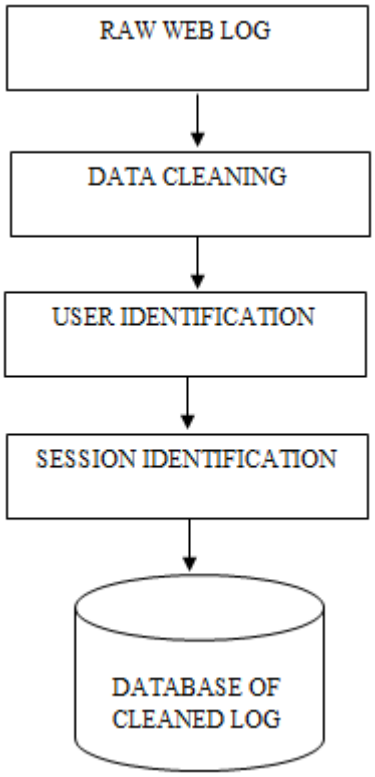

Figure 2: Data Preprocessing Sequence

Raw Web Log is data collected and structured from the Internet.

Data Cleaning: The initial step in data preprocessing is the removal of superfluous requests made from the log files. This process typically eliminates requests associated with HTTP failures, crawler logs, and non-analyzed resources such as images and multimedia.

User Identification: User identification algorithm contributes to a unique user database containing information such as the total number of users, user agents, IP addresses and browsers. Proxy servers, corporate firewalls, and local caches significantly complicate user identification. A set of recommendations regarding user identification has been put forth in an effort to resolve these concerns.

i. 'If there is a new IP address there is a new user'
ii. When distinct operating systems or browser software coexist with an identical IP address, we can reasonably assume that each agent type associated with that IP address represents a unique user. The implementation of these regulations within the user identification algorithm produces a database containing comprehensive details about every user, such as their user agent, browser type, and IP address.

Session Identification: A user session refers to the compilation of webpages that an individual user views on a website during a singular session. We shall categorize the actions of a given user into a single session, provided that the time between page requests does not surpass a specific threshold. We configure the system with two 12-hour sessions, one in the morning and one in the afternoon. The rules for session identification are as follows:

i. 'Every time a new user joins, a new session begins'.
ii. If the reference page remains blank following the conclusion of a single user's session, it is possible to deduce that a new session has commenced.3
iii. When the time between page requests surpasses a predefined threshold, we assume a new session is starting.

2. PrefixSpan

Using an extended iteration of the prefixspan algorithm, the proposed solution extracts recurrent sequential patterns from a server log file. Represent the collection of elements as I = i1, i2,... ir. A one-dimensional sequence, denoted as <s1, s2,... sl >1, is an ordered list of elements. For this instance, si ϵ I where (1 < i < 1). Sequences of n dimensions (where n>1) can be formed by sorting lists of sequences with (n-1) dimensions. The sequence si may be represented as <s1, s2,... sl>n, where n represents the number of dimensions, for each value of i ranging from 1 to l. Let B represent the i-dimensional component of an n-dimensional sequence. A includes both A(k) and A(j) as separate elements, for any two items A(k) and A(j) in A. None of the u-dimensional elements in A, where u is smaller than i, have this feature. Hence, the notation DS(A,k,j)=i signifies that the elements A(k) and A(j) has a scope of dimension i.

3. Sequential Pattern in Web Log

Web servers generate a log entry, including the requested URL, IP address, and timestamp, in response to each request. Figure 3 is Sample Data Sequences in Weblog.

<160by2.com, 160by2.com,160by2.com>

<jaxtr.com, mail.google, mail.google, mail.google ,>

<home.microsoft,home.microsoft,tce.edu443HTT,infinity.ic icibank,broker.ad

obe,microsoft.com, msn.com, microsoft.com,>

<yahoo.com, us.a2,us.il,>

<iris.tcenet, iris.tcenet, iris.tcenet,>

Figure 3: Sample Data Sequences in Weblog

A WAP is a pattern that appears repeatedly in a sizeable collection of Web logs and is searched for by users. An event and user ID sequence can be thought of as a web log. Web log files are split up for mining purposes.To get portions of web logs, preprocessing can be applied to the original web log files. Each entry in the Web log is an event sequence from the user or session, arranged chronologically (earliest event goes first). Web logs in bits and pieces as a series of events, then mine the sequence of patterns over a predetermined support threshold.

160by2.com

mail.google

iris.tcenet

jaxtr.com

us.i1

Figure 4: Frequent Prefixes

Output Design

Mining web logs within web pages determines the outcome. Its regular pattern guides the extraction of the objects. After being fed the web logs, the prefixSpan system mines the data and generates the output that is displayed in Table 2.

Table 2: Output resulting from the processing of Input

Output Variables	Descriptions
10.2.1.34 30/aug/2022 FN microsoft.com	IP address/date/URL
10.2.1.35 30/aug/2022 FN home.microsoft.com	IP address/date/URL
10.2.1.51 30/aug/2022 FN iris.tcenet	IP address/date/URL
10.2.1.52 30/aug/2022 FN iris.tcenet	IP address/date/URL
10.2.1.53 30/aug/2022 FN fxfeeds.mozilla	IP address/date/URL
10.2.1.54 30/aug/2022 FN newsrss.bbc	IP address/date/URL
Mohammad Norouzi/University of Toronto	Author/affiliation

4. Implementation and Result

The programming languages used in the implementation of the system:

Matlab R2017a and MLX file which contains a live script or function in the MATLAB Live Code format

System Setup

The steps to run application include:

1. Do a double clicking on Matlab on the desktop to open
2. Click open file
3. Locate the file with the extension mlx
4. Click open
5. The file runs while opening.

Results

An input file containing a web log from the NIPS website feeds the proposed algorithm. The papers presented at the annual conference on machine learning and computational neuroscience, now known as NeurIPS (Neural Information Processing Systems), detail the most recent advancements and discoveries in the field. Table 3 presents the authors of the extracted publications. Authors of articles are individuals who have made substantial contributions to the content.

Table 3: Authors

ID	Names
178	Yoshua Bengio
200	Yann LeCun
205	Avrim Blum
347	Jonathan D. Cohen
350	Samy Bengio
521	Alon Orlitsky
549	Wulfram Gerstner
575	Robert C. Williamson
583	Sanmi Koyejo
590	Danilo Rezende
592	Bill Dally
596	Yoshua Bengio
600	Yann LeCun
601	Samy Bengio

Using the most frequently visited web pages as reference material, the system generates patterns of results.

Minimum support is 50%. Support can be obtained as follows:

$$Support = \frac{minimum\ spport}{100} * number\ of\ affiliation \quad (1)$$

$$Support = \frac{50}{100} * 4 = 2$$

Table 4: Frequent Sequential Accessed Affiliation

Affiliation	Support
NICTA	3.5
Stanford University	2
Google Research	1

Table 5: Number of Accepted Papers from NIPS Website

Year	Papers
2015	400
2016	550
2017	700
2018	1000
Total	**2650**

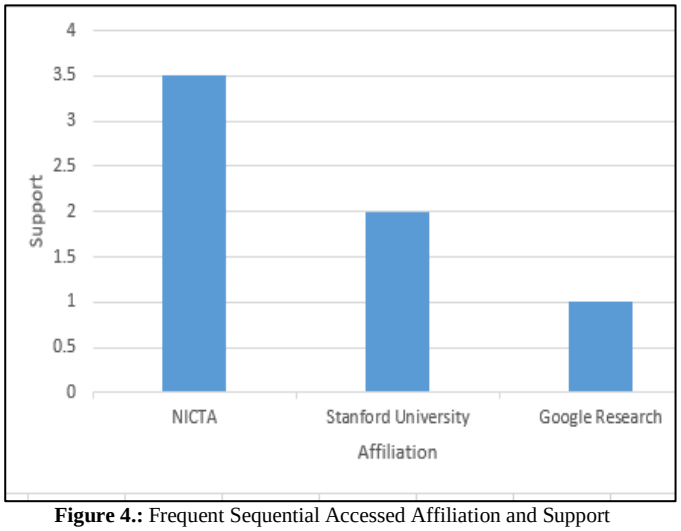

Figure 4.: Frequent Sequential Accessed Affiliation and Support

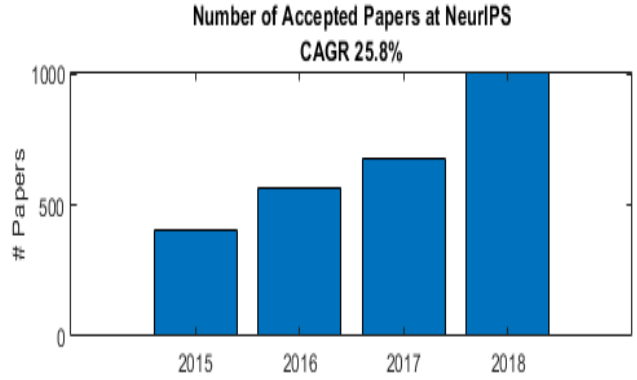

Figure 5: Frequent Sequential Accessed Affiliation and Support

Table 6: 'The average extraction time of three different ways for each category'

Parameters	**All Extractions**	**Improvement**	**AETR**	**AETIT**	**Improvement**
Number of extraction tasks	7410	N/A	5020	6580	N/A
Extraction	0.130	N/A	0.088	0.139	N/A
InnerTag Count	0.128	1.43%	0.085	N/A	3.08%
Starting position	0.096	27.01%	N/A	N/A	N/A
Only repetition	0.094	28.57%	N/A	0.097	30.31%
Abbreviation: AETR: 'Appropriate extraction tasks for repetition' AETI: 'Appropriate extraction tasks for innerTag count'					

5. Discussion

From the result of the extraction from NIP website, Table 4 and figure 4 shows the tabular and graphical result of frequently sequential accessed affiliation. NICTA had the highest support with 3.5 followed by Stanford University with 2 as support and then Google Research with 1 as support. Table 5 and figure 5, shows number of accepted papers from NIPS website from 2015 to 2018. The least papers were in 2015 with 400 accepted papers but as the years progressed there was an increase in the list of acceptable paper which is seen in 2018 with 1000 accepted papers. The data mentioned in table 6 have an impact on all extractions, including appropriate extractions, as illustrated above. The experiments comprised a compilation of 30 web pages and 247 extraction patterns in total. To be more specific, we have effectively completed 7410 extraction procedures without encountering any negative incidents. By using the beginning location information, the mean extraction time is reduced dramatically from 0.130 milliseconds to 0.096 milliseconds. By only using the initial position data, it is feasible to get a notable enhancement of roughly 27.01%.
The innerTagCount resulted in a drop in the average extraction time from 0.130 ms to 0.128 ms. This indicator shows a little improvement of around 1.43 percent.

By using the isolated repetition information throughout the extraction job, the average extraction time lowers from 0.130 ms to 0.094 ms. We have observed an approximate 28.57% improvement.

For innerTagCount data collection, there are 5020 suitable extraction tasks, and 6580 for repetition data collection. This information could prove to be quite beneficial in terms of extraction procedures. Incorporating the innerTagCount data results in a marginal enhancement of approximately 3.08%.
Repetition information inclusion yields a substantial enhancement of approximately 30.31 percent across all 6580 extraction tasks. Consequently, the utilization of the starting position and repetition data results in a substantial reduction in extraction time when compared to the innerTagCount data.

6. Conclusion

The internet disseminates a substantial volume of information, and numerous individuals are enthusiastic about harnessing its potential. One potential application of this technology is to enhance the size and caliber of the existing data by incorporating enrichment into the datasets themselves. However, there are situations where HTML elements conceal valuable information, complicating the extraction process. Web mining has implemented Matlab to extract valuable information from the internet, including user profiles and content.

To facilitate the automated identification of patterns in user behavior on the web, this study presents a web mining framework that employs frequent sequential pattern mining and web log analysis. The described method can derive web user patterns from massive datasets with minimal assistance. This methodology's mean duration, a straightforward strategy, results in enhanced efficiency. PrefixSpan algorithm has been used to extract tabular as well as unstructured data from HTML tag. The prefixspan algorithm generates frequently accessed webpage sequenced. The comparative analysis of mining time in different extraction patterns has been demonstrated using innerTag count, starting position, and repetition. The Matlab programming language has been used for extracting valuable data from the internet, including user logs and content. The utilisation of the proposed method has been implemented in order to optimise both efficiency and accuracy inside the process. This method provides additional data, such as the initial location, the quantity of nested tags, and the frequency of their occurrence. In comparison with another system, the PrefixSpan algorithm has used less memory in comparison to GSP and markov's model. The PrefixSpan algorithm has also more performance than Markov model and GSP algorithm with execution time 2.35seconds. It is clear that GSP algorithm is efficient. However, PrefixSpan Algorithm is more efficient with respect to running time and space utilization.

The optimisation of web usage mining may be enhanced by establishing a correlation between user behaviour and the specific content of web pages. Potential topics for future research encompass distributed mining using the presented

methodology, as well as the application of these approaches to the incremental mining of web logs.

Conflict of Interest
There is no conflict of interest among the authors.

Funding Source
Non exist.

References

[1] B. J. Daher, "Sequential Pattern Generalization for Mining Multi-source Data," *Computer Science [cs]. Université de Lorraine*, **2020.**
[2] M. Berthold, & D. J. Hand, "Intelligent Data Analysis: An Introduction*" Springer-Verlag New York, Inc., Secaucus, NJ, USA*, **1999.**
[3] S. A Catanese, P. De-Meo, E. Ferrara, G. Fiumara & A. Provetti, "Crawling facebook for social network analysis purposes." *In Proc, International Conference on Web Intelligence, Mining and Semantics, Sogndal, Norway, ACM*, 52 pp.**1–8, 2011.** https://doi.org/10.1145/1988688.1988749, 2011
[4] R. Bhaumik, R. Burke, & B. Mobasher, "Effectiveness of Crawling Attacks Against Web-based Recommender Systems". *In: Proceedings of the 5th workshop on intelligent techniques for web personalization (ITWP)*, **2007.**
[5] M. J. Zaki, "SPADE; An Efficient Algorithm for Frequent Sequences". *Machine Learning*, 42. pp.**31-60, 2021.**
[6] J. Han, J. Pei, B. Mortazavi-Asl, Q. Chen, U. Dayal, and M. C. Hsu.. "FreeSpan: Frequent Pattern-Projected Sequential Pattern Mining". *In Proceedings International Conference Knowledge Discovery and Data Mining (KDD)*, pp.**355-359, 2000**
[7] J. Pei, J. Han, H. Pinto, Q. Chen., U. Dayal & Hsu, M. C. "PrefixSpan: mining sequential patterns efficiently by prefix-projected pattern growth". *Proceedings of 12th International Conference on Data Engineering, Heidelberg, Germany*, pp.**215-224, 2001.**
[8] C. Antunes & A. L. Oliveira. "Sequential pattern mining algorithms: trade-offs between speed and memory". *In Workshop on Mining Graphs, Trees and Sequences (MGTS-ECML/PKDD), Pisa, Italy*, pp.**213-216, 2004.**
[9] Z. Yang & M. Kitsuregawa. "LAPIN-SPAM: An improved algorithm for mining sequential pattern". *In Proceedings of the 21st International Conference on Data Engineering Workshops, Tokyo, Japan*, pp.**1222-1229, 2013.**
[10] C. Marquardt, K. Becker & D. Ruiz. "A Preprocessing Tool for Web Usage Mining in the Distance Education Domain". *In Proceedings of the International Database Engineering and Application Symposium (IDEAS)*, pp.**78-87, 2004.**
[11] R. W. Cooley. "Web Usage Mining Discovery and Application of Interesting Pattern from Web Data". *PhD Thesis, University of Minnesota*, **2000.**
[12] F. Bonchi, C. Giannotti, G. Gozzi, M. Manco, D. Nanni, C. R. Pedreschi & S. Ruggieri. "Web Log Data Warehousing and Mining for Intelligent Web Caching". *Data Knowledge Engineering*, 39(2), pp.**165-189, 2011.**
[13] Doja, M. N. "Web data mining in E-services–concepts and applications." *Indian J. Comput. Sci. Eng*, 8 pp.**313-318, 2017.**
[14] S. K. Girish. "Web Usage Mining for Comparing User Access Behaviour using Sequential Pattern," .**2015.**
[15] N. K. Tyagi, A. K. Solanki & S. Sanjay Tyagi. "An Algorithmic Approach to Data Preprocessing in Web Usage Mining", **2010.**
[16] L. Choudhary, L & S. Swami. Exploring the Landscape of Web Data Mining: An In-depth Research Analysis. *Current Journal of Applied Science and Technology*, *42*(24), pp.**32-42, 2023.**
[17] Rathi, Preeti, and Nipur Singh. "An efficient algorithm for data preprocessing and personalization in Web usage mining." *International Journal of Computer Sciences and Engineering* 7.5, pp.**160-164, 2019.**

AUTHORS PROFILE

Elliot Soyemi Jane earned her B. Sc in Computer Science from the University of Uyo, M. Sc in Information Technology from National Open University of Nigeria and M.Sc in Computer Science in River State University in 2002, 2015, 2020 respectively. She is currently a research scholar who is working on her Ph.D. in Computer Science. She is a member of Computer Professional of Nigeria (CPN).

Dr. E. O. Bennett graduated with a Bachelor's degree in Computer Science from Rivers State University, Port Harcourt, Nigeria in 1998, MSc and PhD in Computer Science from University of Port Harcourt in 2008 and 2014 respectively. He is currently an Associate Professor & Lecturer in the Department of Computer Science, Rivers State University, Port Harcourt. He is a member of Computer Professionals of Nigeria (CPN). He has published over 50 research papers in reputed international journals. His research works focus on Algorithms, parallel, distributed & Intelligent computing.

