Implementation of Web Page Prediction Using Web Usage Mining by Markov Tree Algorithm and Longest Common Subsequence (LCS)

Su Wine Htut, Daw Wai Wai Lwin
University of Computer Studies, Yangon
ms.suwinehtut@gmail.com

Abstract

A web prediction model helps to predict user requests ahead of time, making web servers more responsive. Web usage mining based on users' clickstream data and it is the main subject in web prediction systems. Prediction the near future web pages based on user's current clickstream data. This paper presents web page prediction through web usage mining and it also presents classification of users' navigation pattern to predict users' future intentions. Markov Tree algorithm is used in web access pattern generation. Markov tree model behaves an All-Kth-order Markov model because of its ability to recognize different order models according to the height of the tree. A Markov tree gives a complete description on the frequency with which a particular state occurs, and the number of times a path to a particular state is used, to access its child nodes. Longest Common Subsequence algorithm is used in web page prediction based on user access patterns.

1. Introduction

The rapid expansion of the World Wide Web has created an unprecedented opportunity to disseminate and gather information online. There is an increasing need to study web-user behavior to better serve the web users and increase the value of enterprises. Web server log files show the user's web browsing actions. The web log data consist of sequences of URLs requested by different clients bearing different IP Addresses. Association rules can be used to decide the next likely web page requests based on significant statistical correlations. Large number of users access web sites from all over the world. When the users accessa web site, large volumes of data are collected and the log is maintained in log files. This series of accessed web pages can be considered as the browsing pattern of user which can be use for predicting next web pages. With the help of user future request prediction the browsing time can be reduced and server load can be decreased as well. Web prediction systems are very helpful in directing the users to the target pages in particular web sites.

Markov model is a machine learning technique and is different from the approach that data mining does with web logs. Data mining approach identifies the classes of users using their attributes and predicting future actions without considering interactivity and immediate implications. There are other techniques like prediction by partial matching and information retrieval that may be used in conjunction with Markov modeling, to enhance performance and accuracy. The main motivation of this paper is to know how to implementation of web page prediction by using web usage mining.

2. Related Work

Web usage mining is processed through information collected from server log files. A log file provides a list of page requests made to a given web server, where a request is characterized by, at least, the IP address of the machine placing the request, the date and time of the request, and the URL of the page requested. From this information it is possible to reconstruct the user navigation sessions within the web site [1], where a session consists of a sequence of web pages viewed by a user within a given time window. Sarukkai [10] presents a study showing that Markov models have prediction power and on this basis proposes a system based on such a model for predicting the next page accessed by the user. In [2], Markov models are utilized for classifying browsing sessions into different categories. A model-based clustering approach is used in which users with similar navigation patterns are grouped into the same cluster (each cluster is represented by a Markov

model), and a method to visualize the data on each cluster is also presented.

Deshpande et al. [3] propose techniques for combining different order Markov models to obtain low state complexity and improved accuracy. The method starts by building the All – K^{th} – Order Markov model, that uses the highest order model out of K that covers each state, and makes use of three different techniques to eliminate states in order to reduce the model complexity. Zhu et al. [11] proposes to use a Markov model inferred from user navigation data to measure page co-citation and coupling similarity, based on in-link and out-link similarity. A clustering algorithm is then used to construct a conceptual hierarchy and the Markov model is used to estimate the probability of visiting another cluster of pages in the concept hierarchy. Jespersen et al. [10] study the quality of a fixed-order Markov model in representing a collection of navigation sessions and, according to two proposed measures for the quality of a set of patterns, they conclude that a fixed-order Markov model has some limitations in the accuracy achieved.

Jalali et al. [4, 5] proposed a recommender system for navigation pattern mining through Web usage mining to predict user future movements. The approach is based on the graph partitioning clustering algorithm to model user navigation patterns for the navigation patterns mining phase. Furthermore, in the recommender phase, longest common subsequence algorithm is utilized to classify current user activities to foresee user next movement.

Mobasher et al., present WebPersonalizer a system which provides dynamic recommendations, as a list of hypertext links, to users [8, 9]. The analysis is based on anonymous usage data combined with the structure formed by the hyperlinks of the site. Data mining techniques (i.e. clustering, association rules and sequential pattern discovery) are used in the preprocessing phase in order to obtain aggregate usage profiles. In this phase Web server logs are converted in clusters made up of sequences of visited pages, and cluster made up of set of pages with common usage characteristics. The online phase considers the active user session in order to find matches among the user's activities and the discovered usage profiles. Matching entries are then used to compute a set of recommendations which will be inserted into the last requested page as a list of hypertext links. WebPersonalizer is a good example of two-tier architecture for Personalization systems.

In this paper, Markov Tree approach is presented. It is a fourth order model, but behaves like an All-K^{th}-order Markov model because of its ability to recognize different order models according to the height of the tree. It has dual characteristics of good applicability and predictive accuracy. A Markov tree gives a complete description on the frequency with which a particular state occurs, and the number of times a path to a particular state is used, to access its child nodes. In the prediction phase, longest common subsequence algorithm is used to predict user intention in the near future request.

3. Web Mining

Web mining is the extraction of interesting and useful knowledge and implicit information from artifacts or activity related to the WWW. Web Mining is the extraction of interesting and potentially useful patterns and implicit information from artifacts or activity related to the Worldwide Web. There are roughly three knowledge discovery domains that pertain to web mining:

- **Web Content Mining**: Web content mining is the process of extracting knowledge from the content of documents or their descriptions. Web document text mining, resource discovery based on concepts indexing or agent based technology may also fall in this category.
- **Web Structure Mining**: Web structure mining is the process of inferring knowledge from the Worldwide Web organization and links between references and referents in the Web.
- **Web Usage Mining**: Web usage mining, also known as Web Log Mining, is the process of extracting interesting patterns in web access logs. Web servers record and accumulate data about user interactions whenever requests for resources are received.

4. Web Usage Mining

Web Usage Mining is the application of data mining techniques to usage logs of large Web data repositories in order to produce results that can be used in the design tasks mentioned above. However, there are several preprocessing tasks that must be performed prior to applying data mining algorithms to the data collected from server logs. Web usage mining has become very critical for effective Web site management, creating adaptive Web sites, business and support services, personalization, and

network traffic flow analysis and so on. Web usage mining tries to discover the useful information from the secondary data derived from the interactions of the users while surfing on the Web. Web usage mining consists of following steps:

- Preprocessing,
- Pattern discovery and
- Pattern Analysis

4.1. Web Log

The web usage data includes the data from web server logs, proxy server logs, browser logs, and user profiles. (The usage data can also be split into 3 different kinds on the basis of the source of its collection: on the server side, the client side (while on the client side there is complete picture of usage of all services by a particular client), and the proxy side (with the proxy side being somewhere in the middle). Web Server logs are plain text (ASCII) files, that is Independent from the server platform. Web server log consists of valuable information such as the client IP, browser, access time, status, agent and so on. Example web log is shown in following Figure 1.

```
#Software: Microsoft Internet Information Services 5.0
#Version: 1.0
#Date: 2011-07-01 01:47:04
#Fields: date time c-ipcs-username s-ip s-port cs-method
cs-uri-stem cs-uri-query sc-status cs(User-Agent)
2011-07-01 01:47:04 216.221.62.142 - 203.81.71.66 80
GET http://www.smsync.com/docs/browner/adminbio.html
- 200 Mozilla)
2011-07-01 01:51:05 151.48.123.70 - 203.81.71.66 80
GET
http://www.smsync.com/docs/oppe/spatial.html?req=d -
200 Mozilla)
2011-07-01 02:52:18  64.124.85.78 - 203.81.71.66 80 GET
http://www.smsync.com/logos/us-flag.gif - 200 Mozilla)
2011-07-01 02:52:18 64.124.85.78 - 203.81.71.66 80 GET
http://www.smsync.com/oswrcra/general/hotline.html  -
200 Mozilla)
2011-07-01 02:52:49 64.124.85.78 - 203.81.71.66 80 GET
http://www.smsync.com/logos/small_gopher.gif - 404
Mozilla)
2011-07-01 02:52:50 64.124.85.78 - 203.81.71.66 80 GET
http://www.smsync.com/smartsyncpro/whatsnew.html
1057114368118 200 Mozilla)
```

Figure 1. Web Usage Log File

4.2. Preprocessing

Web log data are preprocessed in order to clean the data – it includes following processes:

- Cleaning: The principle of data cleaning is to reduce extraneous items. According to the purposes of different mining applications, extraneous records in web access log will be eliminated during data cleaning. Since the target of Web Usage Mining is to get the user's access patterns, following two kinds of records are unnecessary and should be removed:
  - The records of graphics, videos and the format information. The records have filename suffixes of .gif, .png, .css, .jpg and so on.
  - The records with the failed HTTP status code. By grouping the Status field of every record in the web access log.
- User Identification: User's identification is, to categorize who access web site and which pages are accessed. The different IP addresses distinguish different users; if the IP addresses are same, the different browsers and operation systems indicate different users. This can be done in various ways like using IP addresses, cookies and so on.
- Session Identification: A user session can be defined as a set of pages visited by the same user within the duration of one particular visit to a web-site. A user may have a single or multiple sessions during a period. The goal of session identification is to divide the page accesses of each user at a time into individual sessions. In this system session identification is processed by time-oriented session identification of 30 minutes is used. Example session database after session identification is shown in Table 1.

Table 1. Session Identification

Session ID	Web access pages
S1	P9
S2	P3
S3	P6,P9,P5,P2
S4	P2,P12
S5	P10
S6	P9
S7	P2
S8	P8,P9,P5,P2
S9	P1
S10	P2
S11	P8

S12	P1,P12,P3
S13	P3,P4,P7,P1
S14	P3
S15	P1
S16	P1
S17	P6
S18	P8,P2

4.3. Pattern Discovery

Pattern discovery process is performed by the data mining algorithm. In this system, Markov tree algorithm is applied to perform the pattern discovery process. It generates all user access patterns from web server log files.

4.4. Pattern Analysis

Patterns generated from the pattern discovery are filtered by the interestingness measure in this step. Patterns after filtering can regard as the frequent patterns, and can later be used in other analysis and enhancement tools.

5. Proposed System

This system presents web page prediction based on user's current navigation pattern. Web prediction by Web usage mining system consists of two main phases:

- Mining of user navigation patterns by Markov tree algorithm
- Prediction of web pages base on user's navigation pattern by LCS algorithm

Classifying of user's current activities based on navigation patterns in the particular web site is the main objective of the prediction engine. System overview design is shown in Figure 2.

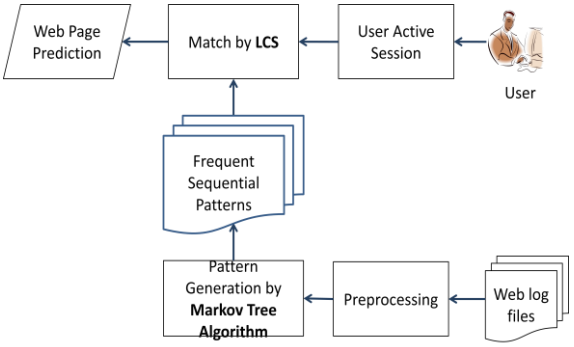

Figure 2. Proposed System Design

5.1. Markov Tree Model

Markov tree model is building tree data structure that implicitly captured Markov property. Markov tree implies that, given the present state of the model, future states are independent of the past states. Thus the description of the present state fully captures all the information that could influence the future evolution of the process. Future states will be reached through a probabilistic process instead of a deterministic one. Markov property can be used to maintain a tree that stores state information. It is simpler and very effective. Trees have the added advantage of being easier to maintain when it comes to pruning. Pruning requires releasing the children and grand children of a particular node, based on minimum support.

Markov tree has the property: while holding the description of the present state, it has the ability to hold all the information that could influence the evolution of the tree and thus the model.

Building Markov Tree

In the Markov tree, each node has the following set of information: self-count, number of children and child-count. These are the minimum set of information that each node should hold. The root node corresponds to a sequence without context. The transition probability of any node with children sums to one.

Algorithm :BuildingMarkovTree,
Input :User Sessions-sessiondb,
Output :Markov Tree
Begin
for each set of sequences associated with sessiondb, Let t as pointer
sub-sequence s = first request
letss be the subsequence of s
if not-exist-node(t(s))

```
        create a new node t(s)
        else t(s) = node (t(s))
        if  ss.count==0
        increment t(s).selfcount
        else
                for j from 0 to sscount
                increment t(s).selfcount
        increment t(s).childcount
        if not-exist-child (t(s), j)
        add a new node for j to the list of t(s)'s
    children
        end if
        let t point to child j
        if  j=last(ss)
        increment t(s).selfcount
        end if
        end for
        end if
        end if
    end for
End
```

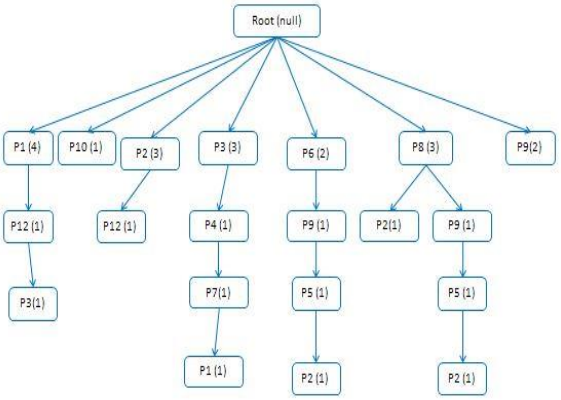

Figure 3. Example Markov Tree

Figure 3 presents example Markov tree. With minimum support = 2, P10 will be pruned since it node count is 1. Traversing the Markov tree produces frequent patterns as in Table 2.

Table 2. Frequent Pattern with minimum support = 2

Frequent Pattern	Support Count
P5,P2	2
P9,P5	2
P9,P5,P2	2

Advantages of Markov Tree

- Markov models are the most effective techniques for Web page access prediction and to improve the Web server access efficiency.
- It is used for the identification of next page to be accessed by the user based on the sequence of previously accessed pages.
- Markov models have been found to be effective in generating sequential patterns of web logs.
- It gains high accuracy in web page predictions of web usage mining.
- High order Markov Model– high predictive accuracy, but extremely high complexity
- Low order Markov Model– insufficient coverage and low predictive accuracy
- **Markov Tree**– behaves like high order Markov model because of its ability to recognize different order models according to the height of the tree. But reduce complexity.

5.2. Longest Common Subsequence

Longest Common Subsequence is the process of comparing two sequences X and Y to determine their similarity is one of the fundamental problems in pattern matching. One of the basic forms of the problem is to determine the longest common subsequence (LCS) of X and Y. The LCS string comparison metric measures the subsequence of maximal length common to both sequences.

- Let sequence X = $x_1, x_2, \ldots, x_n$ and sequence Y = $y_1, y_2, \ldots, y_n$.
- Y is a subsequence of X if there exists a strictly increasing sequence $j_1, j_2, \ldots, j_n$ of indices of x such that for all i = 1, 2, .., n.
- For two sequences X and Y, Z is a common sequence of X and Y if Z is a subsequence of both X and Y.
- Maximum length or longest common subsequence (LCS) gives two paths or sequence of page visits X = $x_1, x_2, \ldots, x_n$ and Y= $y_1, y_2, \ldots, y_n$.

Input: Two sequences $X=x_1x_2...x_m$, and $Y=y_1y_2...y_n$.
Output: Longest common subsequence of X and Y
Similarity of two lists:

- **Given two lists:** X: 1, 2, 3, 4, 5 and Y:1, 3, 2, 4, 5
- LCS = 1, 3, 4, 5 and 1, 2, 4, 5
- Length of LCS=4 indicating the similarity of the two lists.

Algorithm LCSMatching

```
public string LCS(string a, string b)
{
stringaSub = a.Substring(0, (a.Length - 1 < 0) ? 0
        : a.Length - 1);
stringbSub = b.Substring(0, (b.Length - 1 < 0) ? 0
        : b.Length - 1);

if (a.Length == 0 || b.Length == 0)
return "";
else if (a[a.Length - 1] == b[b.Length - 1])
return LCS(aSub, bSub) + a[a.Length - 1];
else
        {
string x = LCS(a, bSub);
string y = LCS(aSub, b);
return (x.Length>y.Length) ? x : y;
        }
    }
```

Example computation of LCS for active user session is shown below:

For the input sequence P9, X = {P9}
If Y1 = {P9,P5,P2}, common sequence Z1= {P9}, LCS=1
If Y2 = {P9,P5}, common sequence Z2= {P9}, LCS=1

- Hence Y1 = {P9,P5,P2} and Y2={P9,P5} has longest common sequence and **P5** is predicted for user.

For the next input sequence P6, X = {P9,P5}
If Y1 = {P9,P5,P2}, common sequence Z1= {P9,P5}, LCS=2
If Y2 = {P5,P2}, common sequence Z2 = {P5},LCS=1

- Hence Y 1= {P9,P5,P2} has longest common sequence and **P2** is predicted for user.

6. System Implementation

This system is implemented using Microsoft Visual Studio .Net 2008.Web log files used in this system are downloaded from http://www.smsync.com. The size of log file is 7 MB and there are total 55 log entries. There are 3 others example log files collected from internet café. They have following size and number of records.

Log File	Size	No. of Records
shuttle	1148 KB	10974 records
avata	2155 KB	13684 records
xpolog	2494 KB	9978 records

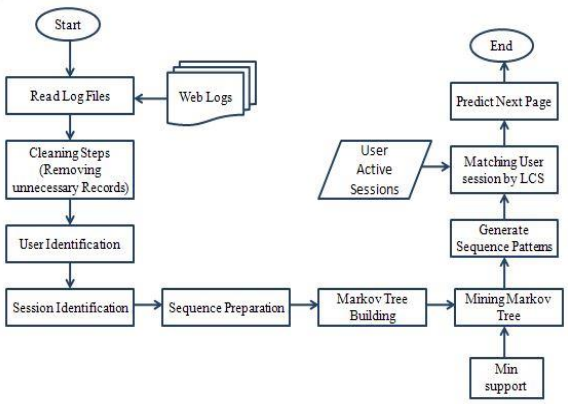

Figure 3. Process Flow of System

Web Prediction by LCS

- Find the longest match of user sessions in web access patterns (generated by Markov Tree)
- Navigation patterns (patterns from web log) np and active session window (user's current navigation pattern) S.
- Find navigation pattern by utilizing longest common subsequences algorithm.
- Navigation pattern with the highest degree of similarity is found according to the LCS algorithm to predict next user's activities and create a recommendation list.

6.1. System Evaluation

Measuring the accuracy of the predictions needs to characterize the quality of the results obtained. There are a lot of performance measures for measuring system correctness. For the classifier system, cross validation for measuring classifier accuracy. It computes for measurements true positive (tp), true negative (tn), false positive (fp) and false negative (fn). Precision, recall and accuracy is computed from those four measurements. For the information retrieval system, precision and recall measurements are computed based on number of retrieved documents, number of total relevant documents and number of relevant documents retrieved. For the recommender system, we use following methods to measure the system performance.

To measure the quality of prediction, we use second half of the dataset after the dataset divided into

two halves; training set and evaluation set. Each navigational pattern *npi* (a session in the dataset) in the evaluation set is divided into two parts. The first *n* page views in *npi* are used for generating predictions, whereas, the remaining part of *npi* is used to evaluate the generated predictions. The active session window is the part of the user's navigational patterns used by the prediction engine in order to produce a prediction set. We call this part of the navigational pattern *np* the active session with respect to np, denoted by as_{np}. The prediction engine takes as_{np} and a prediction threshold τ (minimum support) as inputs and produces a set of page views as a prediction list P (as_{np}, τ). The set of page views P (as_{np}, τ) can now be compared with the remaining |np| − n, page views in *np*.

Three different metrics namely, accuracy, coverage and F1 measures are used to measure the performance of the system.The accuracy of prediction set is defined as:

$$Accuracy\left(P\left(as_{np},\tau\right)\right)=\frac{\left|P\left(as_{np},\tau\right)\cap eval_{np}\right|}{P\left(as_{np},\tau\right)}$$

Where $|P(as_{np}, \tau) \cap eval_{np}|$ is number of web pages that these are common in the prediction list and evaluation set.

Accuracy is Number of relevant web pages retrieved divide by the total number of web pages in recommendations set. Another evaluation parameter is coverage that is defined as

$$Coverage\left(P\left(as_{np},\tau\right)\right)=\frac{\left|P\left(as_{np},\tau\right)\cap eval_{np}\right|}{eval_{np}}$$

Coverage is number of relevant web pages retrieved divide by the total number of web pages that actually belong to the user sessions. In the other hand, coverage measures the ability of the prediction engine to produce all of the pageviews that are likely to be visited by the user.The F1 measure attains its maximum value when both accuracy and coverage are maximized. Finally, for a given prediction threshold τ, the mean over all navigational pattern in the evaluation set is computed as the overall evaluation score for each measure.

$$F1=\frac{2*Accuracy(P(as_{np},\tau))*Coverage(P(as_{np},\tau))}{Accuracy(P(as_{np},\tau))+Coverage(P(as_{np},\tau))}$$

This system is tested with different log files; and system evaluation results are as follows: they are tested with minimum support 3

Log File Size	Accuracy	Coverage	F1
7 KB	66.67%	100%	80%
1,148 KB	87.44%	47.74%	61.74%
2,155 KB	93.33%	72.92%	81.87%
2,494 KB	93.23%	38.08%	54.07%

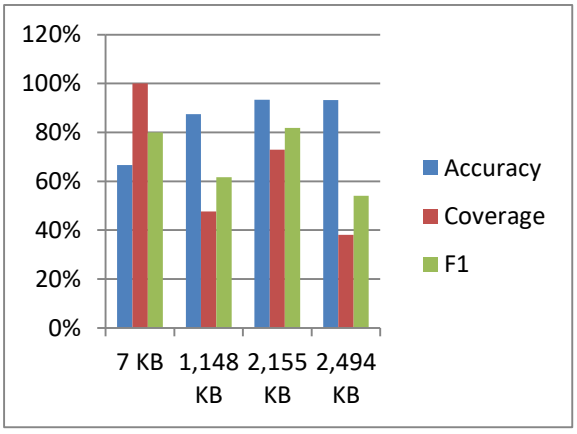

Figure 4. Performance of the System for Different Web Log files

7. Conclusion

This system presents Web page prediction system for conference papers which predict the near future web pages based on user's current clickstream data. Web page prediction system is very helpful in directing the users to the target pages in particular web sites.Hence it produces more help to user can be easily access and user's future request prediction of browsing time can be reduced and sever load can be decreased as well. So the performance of particular website is can be improving the experience of the users on the web.

References

[1] Berent, B., Mobasher, B., Spiliopoulou, M., and Wiltshire, J. (2001). Measuring the accuracy of sessionizers for web usage analysis.In Proceedings of the Web Mining Workshop at the First SIAM International Conference on Data Mining, Chicago.

[2] Cadez, I., Heckerman, D., Meek, C., Smyth, P., and White, S. (2000). Visualization of navigation patterns on a web site using model based clustering. In Proceedings of the Sixth International KDD conference.

[3] Deshpande, M. and Karypis, G. (2001). Selective markov models for predicting web-page accesses. In Proceedings of the First SIAM International Conference on Data Mining.

[4] Jalali, M., Mustapha, N.,Sulaiman, M. N. B. and Mamat, A. "OPWUMP: An Architecture for Online Predicting in WUM-Based Personalization System," Communications in Computer and Information Science,Advances in Computer Science and Engineering,Springer Berlin Heidelberg, vol. 6, pp. 838–841, 2008.

[5] Jalali, M., Mustapha, N.,Sulaiman, M. N. B. and Mamat, A., "A Web Usage Mining Approach Based on LCS Algorithm in Online Predicting Recommendation Systems," in 12th International on Information Visualisation,IV'08, London, UK, 2008, pp. 302-307.

[6] Jespersen, S., Pedersen, T. B., and Thorhauge, J. (2003). Evaluating the markov assumption for web usage mining.In Proceeding of the Fifth International Workshop on Web Information and Data Management (WIDM'03), pages 82{89, New Orleans - Lousiana, USA.

[7] Kurian, H, "A Markov Model For Web Request Prediction", B.TECH. Computer, Dr. BabasahebAmbedkar Technological University, 2005.

[8] Mobasher,B. Cooley, R. and Srivastava, J. "Automatic personalization based on Web usage mining," Communications of the ACM, vol. 43, pp. 142-151, 2000.

[9] Nakagawa M. and B. Mobasher, "A hybrid web personalization model based on site connectivity," 2003, pp. 59–70.

[10] Sarukkai, R. R. (2000). Link prediction and path analysis using markov chains. Computer Networks, 33(1-6):377{386.

[11] Zhu, J., Hong, J., and Hughes, J. G. (2002). Using markov models for web site link prediction. In Proceedings of the 13th ACM Conference on Hypertext and Hypermedia.