Indian Journal of Science and Technology, *Vol 8(27), DOI:10.17485/ijst/2015/v8i27/72455, October 2015*
*ISSN (Print) : 0974-6846*
*ISSN (Online) : 0974-5645*

# Applying Supervised Learning Techniques for Constructing Predictive Models

**Nirali Honest[1], Bankim Patel[2] and Atul Patel[1]**

[1]Smt. Chandaben Mohanbhai Patel Institute of Computer Applications, Charotar University of Science and Technology, (CHARUSAT), Changa - 388421, Gujarat, India; niralihonest.mca@charusat.ac.in, atulpatel.mca@charusat.ac.in
[2]Shrimad Rajchandra Institute of Management and Computer Application, Uka Tarsadia University, Tarsadi - 394350, Gujarat, India; bankim_patel@srimca.edu.in

## Abstract

**Background/Objectives:** The website is composed of permanent and temporary pages. Deriving a prediction model which considers the dynamic pages generated on the website requires to consider new aspects. **Methods/Statistical Analysis:** We adopt supervised learning models as they give better prediction results for new input data. After reading the log files and applying preprocessing, we build the user navigation patterns and then apply the prediction of pages. The main parameter on which we have modified is the time stamp. In earlier approaches the time stamp was divided into day, month and year and based on the timestamp granule selected the prediction model was formed. In our work we consider the same granule with introduction to new timestamp namely event. **Findings:** Markov model are very good in predicting the pages for n length, but the model doesn't focus on temporal aspect for prediction. Temporal n-gram model covers the temporal aspect of prediction by forming the granules of time. This model gives good accuracy in predicting pages that are permanent for any given website, but doesn't tend to be good for pages that are temporary in nature. Our model focuses on temporal aspect for both types of pages by creating an event based temporal n-gram model. Event means creating a special named interval for which the pages are made available on the website. This means that after the interval specified in the event the page will be no more visible on the website. The pages are predicted based on the nature of pages, we form broadly two types of nature of pages 1. Regular for permanent pages and 2. Event for temporary pages. By introducing this temporal aspect the prediction algorithm considers the specified interval only for the event specific pages, after the interval is over the pages are not considered for prediction. **Application/Improvements:** Specifying events help to derive better accuracy in prediction when we consider permanent and temporary pages, as we predict the pages based on the condition whether they are regular or event based pages.

**Keywords:** Classification Algorithms, Conditional Probability, Event Based Granule Model, Naive Baysian, Prediction Model

## 1. Introduction

The supervised learning can be applied to construct a predictor model that generates sensible predictions for the reply to the new data as shown in Figure 1. A test data set can be used to validate the model, if size of training data is larger than a better predictive model can be build for the new data set.

Supervised learning includes two categories of algorithms

- **Classification:** For categorical response values, where the data can be separated into specific "classes".
- **Regression:** For continuous-response values.

In Supervised learning the data, observations, measurements, etc. are labeled with pre-defined classes, whereas in unsupervised learning (clustering) class labels of the data are unknown. In most of the supervised

**Author for correspondence*

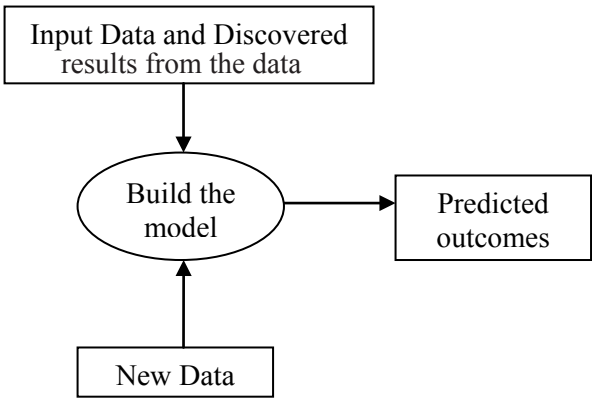


**Figure 1.** Use of supervised learning in prediction model.

learning models, the models are divided into two parts, learning and testing.

- **Learning or Training:** Train a model using the historical data.
- **Testing:** Test the model using new test data to assess the model accuracy, accuracy can be calculated by number of correct classification/total number of records.

## 2. Classification Algorithms

Common classification algorithms include, Decision trees, Support Vector Machine, Nearest Neighbor, Naive Baysian, etc.

- Decision tree learning is one of the most widely used techniques for classification. Its classification accuracy is competitive with other methods and it is very efficient. However, finding rules from trees, managing missing values, attribute formation is difficult.
- Support Vector Machines (SVM) are linear classifiers that find a hyperplane to separate two class of data, positive and negative. SVM not only has a rigorous theoretical foundation, but also performs classification more accurately than most other methods in applications, especially for high dimensional data. It is perhaps the best classifier for text classification. However SVM works only in a real-valued space. For a categorical attribute, we need to convert its categorical values to numeric values. SVM does only two-class classification. For multi-class problems, some strategies can be applied, e.g., one-against-rest and error-correcting output coding. The hyperplane produced by SVM is hard to understand by human users. The matter is made worse by kernels. Thus, SVM is commonly used in applications that do not require human understanding.
- In Nearest Neighbor method it does not build model from the training data. No training is needed. Classification time is linear in training set size for each test case. Nearest Neighbor can deal with complex and arbitrary decision boundaries. Despite its simplicity, researchers have shown that the classification accuracy of Nearest Neighbor can be quite strong and in many cases as accurate as those elaborated methods. Nearest Neighbor is slow at the classification time. Nearest Neighbor does not produce an understandable model.
- In Naive Baysian method Supervised learning is applied from a probabilistic point of view. It is easy to implement, very efficient, good results obtained in many applications. However poor assumptions in forming the class conditions may lead to lack of accuracy.

General characteristics of any supervised learning algorithm can be given as,

- Speed of training.
- Usage of Memory.
- Accuracy of Prediction on new data.
- Interpretability, ease of understanding about how algorithm makes its predictions.

We list the behavior of Supervised Learning (SL) methods discussed above with the characteristics in Table 1.

## 3. Compute Page Prediction

Various approaches have been proposed for characterizing user behavior and predicting the user's next page request. According to[1], association rules, sequential pattern discovery, clustering and classification are most popular methods for web usage mining. Association rules were proposed to capture the related pages by forming rules[3], use association rules. Association rules were mainly used to form the buying patterns in a super market shopping[4]. Authors in[1] apply sequential association rules for predicting web pages. Dependency graph[5] is also used to know the user behavior, it forms the pattern for user page request, every page that is visited by a user is represented as a node in the graph, in this method the consecution of requests is not considered. Markov model[6] is used to model the user navigation sessions. Lower order Markov models are not

**Table 1.** General characteristics of SL methods

| Sl. No. | Algorithm | Characteristics |
|---|---|---|
| 1 | Tree | Accuracy in prediction is Average.<br>Speed of prediction is Fast.<br>Usage of memory is Low.<br>Interpretation is easier.<br>Fitting speed is Fast. |
| 2 | SVM | Accuracy in prediction is good.<br>Prediction speed and memory usage are good for few support vectors, but can be poor for many support vectors.<br>Fiiting speed is medium. |
| 3 | Nearest Neighbor | Accuracy in prediction depends on scope and size, it is has good predictions in low proportions, but can have poor predictions in high proportions.<br>Speed of prediction is Medium.<br>Usage of memory is High.<br>Interpretation is not easier.<br>For linear search it does not fit, it fits for kd trees. |
| 4 | Naive Bayes | Accuracy in prediction is medium.<br>Speed of prediction and usage of memory is good for simple distributions but poor for kernel distributions.<br>Interpretation is easier. |

so accurate in predicting the user's browsing behavior whereas higher order markov models give better coverage. In[2] pruning criteria like support, confidence and error where covered to improve the efficiency. Increased number of states in markov models, pattern in sequential pattern and association rules result in more requirement of memory and computation power. In[7] authors consider the future page visits on the based on the current visits and the global query log. It is important to consider that different users have different browsing pattern over different times, so milestone based approach is considered to know the point in change of browsing[8]. Authors in[9] consider the maximum utility measure, to calculate the subsequences in mining. In[10] parameters like frequency, utility, down loads and selection are considered in each node of this optimal prefix tree. The prediction accuracy of patterns is improved by considering the pattern in total number of patterns extracted and also the time spent on page[11]. In[12] we consider the current search pattern for the querying the log by considering the order, adjacency, recency and event based temporality. In this paper we discuss the types of pages of a website which can fall in one of the two categories 1. Regular and 2. Event based. We use a Naive Bayes algorithm to incorporate conditional probability for the pages selectively based on the type of page. We predict the pages by applying probability calculation as below,

Probability of Page P1, P(P1) =

$$\frac{\text{Number of occurrence of P1}}{\text{Total number of Pages Pages Condition1}}$$

Probability of Page for given length and given

even Probability of P1 for length = 1 and event = Regular

$$P(Page = P1 \mid Length = 1 \mid Event = Regular) = \frac{\text{No. of occurrence of Page P1 and Length1}}{\text{Total number of Pages P1}}$$

Consider the Table 2. Which consists data of pages accessed by users for given length.

**Table 2.** List of pages accessed by users of length 1, 2 and 3 for a given date

| | Users | Length | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| Pages accessed for a given date | U1 | P1 | P5\|P3 | P2\|P4\|P5 |
| | U2 | P4 | P2\|P3 | P3\|P5\|P2 |
| | U3 | P8 | P1\|P4 | P2\|P4\|P5 |
| | U4 | P6 | P2\|P3 | P3\|P5\|P2 |
| | U5 | P6 | P4\|P8 | P4\|P7\|P8 |
| | U6 | P3 | P2\|P3 | P2\|P4\|P5 |
| | U7 | P6 | P8\|P6 | P8\|P3\|P9 |
| | U8 | P2 | P3\|P10 | P4\|P6\|P7 |
| | U9 | P14 | P5\|P3 | P3\|P8\|P6 |
| | U10 | P2 | P7\|P9 | P5\|P8\|P10 |
| | U11 | P15 | P15\|P14 | P8\|P11\|P13 |
| | U12 | P14 | P12\|P14 | P3\|P7\|P10 |
| | U13 | P8 | P3\|P10 | P4\|P6\|P12 |
| | U14 | P6 | P2\|P4 | P10\|P11\|P14 |
| | U15 | P4 | P14\|P9 | P2\|P8\|P9 |

From the given Table 2, we calculate the probability of each page from the page count, for a given length as shown in Table 3.

**Table 3.** List of page probability for a given length

| Pages | P(page)count | P(page)count_Avoiding_Zero_Probability | $P(page)_{Length}$ | |
|---|---|---|---|---|
| P1 | 1 | 2 | 0.133333333 | |
| P2 | 2 | 3 | 0.2 | |
| P3 | 1 | 2 | 0.133333333 | |
| P4 | 2 | 3 | 0.2 | |
| P5 | 0 | 1 | 0.066666667 | |
| P6 | 4 | 5 | 0.333333333 | |
| P7 | 0 | 1 | 0.066666667 | |
| P8 | 2 | 3 | 0.2 | Length=1 |
| P9 | 0 | 1 | 0.066666667 | |
| P10 | 0 | 1 | 0.066666667 | |
| P11 | 0 | 1 | 0.066666667 | |
| P12 | 0 | 1 | 0.066666667 | |
| P13 | 0 | 1 | 0.066666667 | |
| P14 | 2 | 3 | 0.2 | |
| P15 | 1 | 2 | 0.133333333 | |
| P5\|P3 | 2 | 3 | 0.2 | |
| P2\|P3 | 3 | 4 | 0.266666667 | |
| P1\|P4 | 1 | 2 | 0.133333333 | |
| P4\|P8 | 1 | 2 | 0.133333333 | |
| P8\|P6 | 1 | 2 | 0.133333333 | |
| P3\|P10 | 1 | 2 | 0.133333333 | Length=2 |
| P7\|P9 | 1 | 2 | 0.133333333 | |
| P15\|P14 | 1 | 2 | 0.133333333 | |
| P12\|P14 | 1 | 2 | 0.133333333 | |
| P3\|P10 | 1 | 2 | 0.133333333 | |
| P2\|P4 | 1 | 2 | 0.133333333 | |
| P14\|P9 | 1 | 2 | 0.133333333 | |
| P2\|P4\|P5 | 3 | 4 | 0.266666667 | |
| P3\|P5\|P2 | 2 | 3 | 0.2 | |
| P4\|P7\|P8 | 1 | 2 | 0.133333333 | |
| P8\|P3\|P9 | 1 | 2 | 0.133333333 | |
| P4\|P6\|P7 | 1 | 2 | 0.133333333 | |
| P3\|P8\|P6 | 2 | 3 | 0.2 | Length=3 |
| P5\|P8\|P10 | 1 | 2 | 0.133333333 | |
| P8\|P11\|P13 | 1 | 2 | 0.133333333 | |
| P3\|P7\|P10 | 1 | 2 | 0.133333333 | |
| P4\|P6\|P12 | 1 | 2 | 0.133333333 | |
| P10\|P11\|P14 | 1 | 2 | 0.133333333 | |
| P2\|P8\|P9 | 1 | 2 | 0.133333333 | |

From the calculated probability we find the maximum probability as shown in Table 4.

**Table 4.** Prediction of the next page for given length

| Length | Maximum Probability | Predicted Page |
|---|---|---|
| 1 | 0.333333333 | P6 |
| 2 | 0.266666667 | P2\|P3 |
| 3 | 0.266666667 | P2\|P4\|P5 |

Likewise we calculate the prediction of every page from the session based on length and maximum frequency as shown in Figure 2.

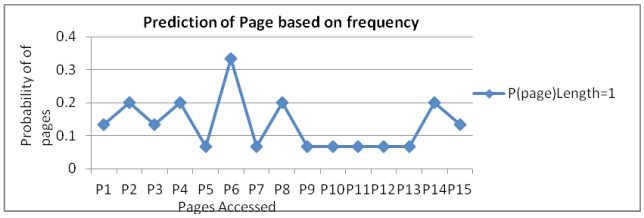


**Figure 2.** Prediction of pages with length 1.

Consider Table 5, to apply the conditional probability we add the event apart from the timestamp when the page is accessed.

**Table 5.** Prediction of page based on sequence

| Event | User | Length | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| Regular | U1 | P1 | P5\|P3 | P2\|P4\|P5 |
| Regular | U2 | P4 | P2\|P3 | P3\|P5\|P2 |
| Ignite | U3 | P8 | P1\|P4 | P1\|P3\|P4 |
| Regular | U4 | P6 | P2\|P6 | P5\|P7\|P9 |
| Regular | U5 | P2 | P4\|P8 | P4\|P7\|P8 |
| Regular | U6 | P3 | P3\|P2 | P2\|P6\|P3 |
| Regular | U7 | P1 | P8\|P6 | P8\|P3\|P9 |
| Ignite | U8 | P2 | P3\|P10 | P4\|P6\|P7 |
| Ignite | U9 | P3 | P5\|P6 | P3\|P8\|P6 |
| Ignite | U10 | P1 | P7\|P9 | P5\|P8\|P10 |
| Ignite | U11 | P10 | P15\|P14 | P8\|P11\|P13 |
| Ignite | U12 | P6 | P12\|P14 | P3\|P7\|P11 |
| Regular | U13 | P8 | P3\|P12 | P4\|P6\|P12 |
| Regular | U14 | P1 | P2\|P4 | P12\|P11\|P14 |
| Regular | U15 | P3 | P14\|P9 | P2\|P8\|P9 |

We can apply conditional probability of Regular and Event pages as shown in Table 6.

**Table 6.** Probability calculation for pages accessed during the event or on regular basis

| P (PageType) | Probality Value |
|---|---|
| P(RegularPages) | 0.6 |
| P(IgnitePages) | 0.4 |

We calculate the probability of predicting the page P1 for the length 1 and event as regular and ignite as shown in Table 7.

**Table 7.** Conditional probability calculation for page p1 of length 1 and event type as regular and ignite

| Conditional Probability | Probability Value |
|---|---|
| P(P1\|L=1\|E=R) | 0.75 |
| P(P1\|L=1\|E=I) | 0.25 |

*R – Indicates Regular
I – Indicates Ignite

Likewise we calculate the probability of every page for predicting the next page as shown in Table 8.

**Table 8.** Conditional probability calculation for pages of length 1 and event type as regular and ignite

| Pages | P(Page\|L=1\|E=R) | P(Page\|L=1\|E=I) |
|---|---|---|
| P1 | 0.75 | 0.25 |
| P2 | 0.5 | 0.5 |
| P3 | 0.666666667 | 0.333333333 |
| P4 | 0 | 0 |
| P5 | 0 | 0 |
| P6 | 0.5 | 0.5 |
| P7 | 0 | 0 |
| P8 | 0.5 | 0.5 |
| P9 | 0 | 0 |
| P10 | 0 | 0 |
| P11 | 0 | 0 |
| P12 | 0 | 0 |
| P13 | 0 | 0 |
| P14 | 0 | 0 |
| P15 | 0 | 0 |

Based on this table we derive the prediction of page for a given length and given event as shown in Figure 3.

To avoid zero probability we can add 1 to the probability value of every page.

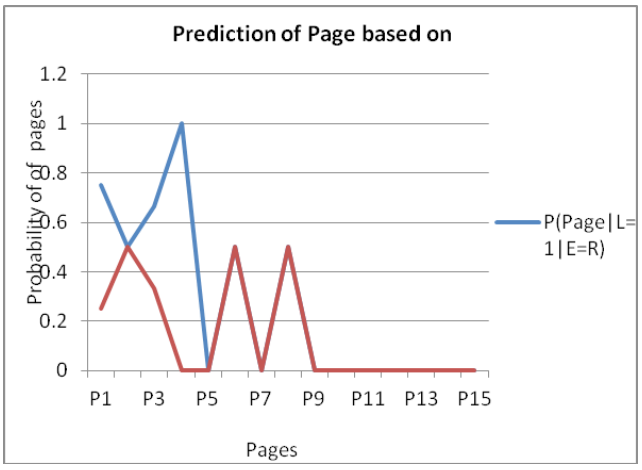


**Figure 3.** Prediction of pages with length 1 based on event.

## 4. Conclusion

In this paper we try to state that based on the type of website the approach should be selected to predict the pages. Selectively applying the conditional probability and forming the rules reduce in number of pages for a given rule and helps in increasing the accuracy with reduction in computational resources.

## 5. Acknowledgement

The authors would like to thank Charotar University of Science and Technology (CHARUSAT) for providing the necessary resources for accomplishing the work.

## 6. References

1. Yang Q, Li T, Wang K. Building association-rule based sequential classifiers for web-document prediction. Springer. 2004 May; 8(3):253–73.
2. Deshpande M, Karypis G. Selective Markov models for predicting Web page accesses. New York. NY USA: ACM Press. 2004 May; 4(2):163–84.
3. Ban Z, Gu Z, Jin Y. An online PPM prediction model for web prefetching. New York. NY USA: ACM; 2007 p. 89–96.
4. Hipp J, Guntzer U, Nakhaeizadeh G. Algorithms for association rule mining—A general survey and comparison. New York. NY USA: ACM Press. 2000 Jun; 2(1):58–64.
5. Padmanabhan VN, Mogul JC. Using predictive prefetching to improve World Wide Web latency. 1996 Jul; 26(3):22–36.
6. Borges J, Levene M. Data mining of user navigation patterns. London UK: Springer-Verlag. 2000; 1836:92–112.
7. Lin K, Wang C, Chen H. Predicting next search actions with search engine query logs. Proceedings of the 2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology; 2011. p. 227–34.
8. Thilagu M, Nadarajan R. Investigating significant changes in users interest on web traversal patterns. Int J Cybern Informatics. 2013 Aug; 2(4):39–55.
9. Lan GC, Hong TP, Tseng VS, Wang SL. Applying the maximum utility measure in high utility sequential pattern mining. Expert Syst Appl. 2014 Sep; 41(11):5071–81.
10. Grace LKJ, Maheswari V. Efficiency calculation of mined web navigational patterns. Indian Journal of Science and Technology. 2014 Sep; 7(9):1350–4.
11. Mehta P, Jadhav SB, Joshi RB. Web usage mining for discovery and evaluation of online navigation pattern prediction. Int J Comput Appl. 2014 Apr; 91(4):23–6.
12. Honest N, Patel B, Patel A. A study of user navigation patterns for web usage mining. International Journal of Advent Research in Computer and Electronics. 2015 Feb; 2(1):5–8.