A Data Driven Approach For Forecasting Click Through Rates

Ashutosh Chaturvedi[1] , Prof. Pankaj Raghuwanshi[2]
[1, 2] Dept of CSE
[1, 2] AIT, Ujjain

Abstract- *There has been a metamorphosis in the advertising realm with the conventional techniques such as billboards, print media and television media facing extremely large competition from the online advertising platform by dint of the fact that a multitude of users purchase online which has increased with the increase in the infrastructure and reliability of online marketing. This has resulted in a necessary requirement to churn out ads specific and apt to the queries entered. The outcome may be a possible click or not based on the experience of the customer. An estimate of click through prior to fetching an add for a query is important for the accurate decision in the context. In this work a recursive binary partitioning algorithm is used along with support vector regression (SVR) to predict click through rates (CTR). It is shown that the proposed work attains a higher accuracy of estimates compared to the benchmark techniques.*

Keywords- Click Through Rates (CTR), ranged data structuring, Binary Partitioning, Wavelet Tree, Time Bidding, Support Vector Regression.

I. INTRODUCTION

The advertising industry has undergone a paradigm shift in terms of its functioning. While earlier version of advertising relied on print media, television and billboards, new age advertising has targeted the online audience to increase its sales and probability of brand fixations [1]. The new age advertising models try and leverage the much larger audience which is constantly on the internet and new branding and advertising methods have some up such as:

- Sponsored search advertising
- Contextual advertising
- Display advertising
- Affiliate marketing
- Online brand influencing
- Real-time bidding auctions etc.

It is extremely important to choose the correct or apt ads for a quarry to maximize the probability of clicks. Is the estimates of click through can be made accurately; they may materialize into staggeringly large profits. For instance an accuracy increase of 0.1% may increase the chances of increasing the profits by a million dollars depending on the diaspora of the audience the add is catering to. In some instances, click baits are also employed to increase the click through rate, which is a bottom-line for the pay per click model [2].

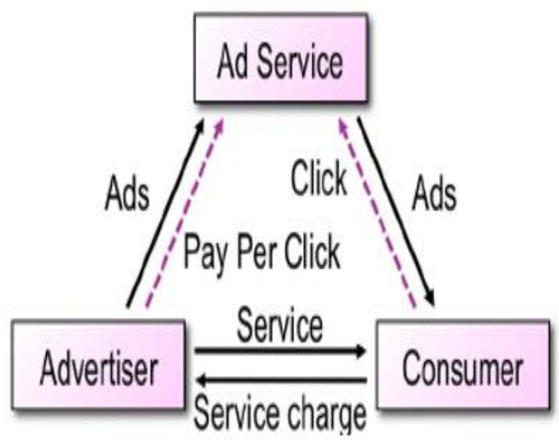

Figure.1 The pay per click model

Typically, the pay per click is measured as:

$$PPC = \frac{Cost_{tot}}{N} \qquad \textbf{(1)}$$

Here,
PPC is the pay per click
$Cost_{tot}$ corresponds to the total add cost.
N is the number of measured clicks

From a business point of view, lesser pay per click is profitable for companies while add designers try to increase the number of clicks using the users online trails, search trends, cookies etc. from the add server [3]. The multitude of data to be managed in this case is staggeringly large and hence effective techniques to manage the same is challenging. The tasks are generally computation heavy and hence machine learning based approaches are needed to accomplish the same [4]

II. RECURSIVE BINARY PARTITIONING

The estimates of click through rates are extremely challenging in nature due to the bipolar and discrete nature of the decision that the user makes. On the contrary a smoothly changing to continuous data set is easier to analyze [9]-[11].The two-way polarity makes the estimates more prone to errors. Hence data preparation and pre-processing is fundamentally important for the prediction problem. Several metrics can be used to augment or bolster the pattern recognition process among which the persistent segment trees (PST) based data structure can be effective [12]. This may help in partitioning the arrays to strings of user data to analyze some important features such as [5]:

1) Maximum clicks in a range
2) Least clicks in a range
3) Frequency of clicks in a range etc.

The concept of a persistent segment tree (PST) is depicted in figure 2.

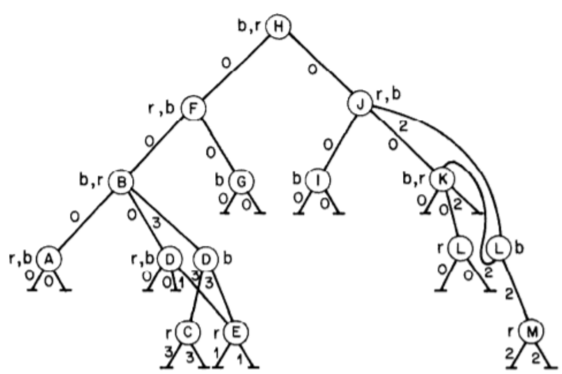

Fig.2 A Persistent Segment Tree [6]

The major problem with the PST based approach is the time complexity which is relatively high given by [6]:

$$T_{PST} = O(nlogn) \quad \textbf{(2)}$$

As the elements keep increasing, the steepness of the complexity increasing making the process computation heavy and slow [15].

An alternative approach is the recursive binary partitioning using the Wavelet Tree. The complexity of such an approach is lesser and is given by [7]:

$$T_{wavelet\ tree} = O(logn) \quad \textbf{(3)}$$

The recursive binary partition approach using the wavelet tree is depicted in figure 3.

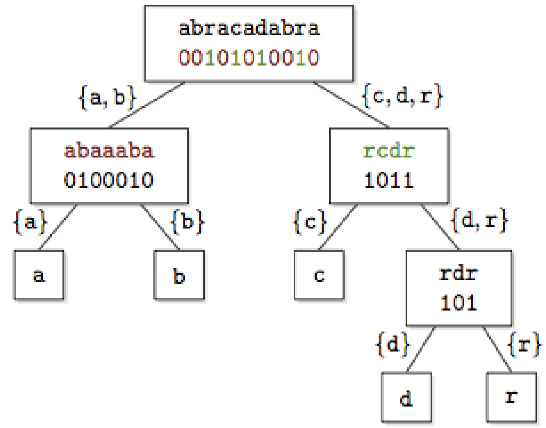

Fig.3 The wavelet Tree [7]

The partitioning of a string or array 'S' is done in the following manner:

1. Consider an array with 'n' elements denoted by S[n].
2. Find max(S) and min(S)
3. Compute the pivot point P as :

$$Pivot = \frac{lower+upper}{2} \quad \textbf{(4)}$$

4. Partition the unsorted string into two sub-strings based on the pivot values. The ones greater to the pivot go to one side of the partition and the ones smaller or equal go to the other side.
5. Recursively partition (without sorting) till you hit leaf node, (when all the elements are same in the decomposed array)

It is necessary to note that the pivot value P may or may not be an integer. The mean based partitioning is generally more common compared to the median based partitioning. The most common operations on the trees are the rank and the quantile. Rank of an element q is the frequency of the element q in the range (I,j) and is given by [7]:

$$R_q = f_{q(i,j)} = f_{q(1,j)} - f_{q(1,i)} \quad \textbf{(5)}$$

Considering the first element to be 0 and considering the limit up to the value I,

$$R_q = f_{q(i)} = f_{q(j)} - f_{q(i-1)} \quad \textbf{(6)}$$

Quantile of k: kth largest element in the range (I,j), this helps us to avoid the persistent segment trees (PST) to solve m kth number problem.

To obtain the quantile, we can use the approach:

$$Q_{k\,(i,j)} \xrightarrow[partition]{} Q_{k(j)} : Q_{k(i)} \quad \textbf{(7)}$$

The quantile and the rank allow the additional features of the data set to be fed to the pattern recognition model so as to increase the accuracy of pattern analysis. The metrics often used are:

1) Initial tree
2) Best Tree
3) Best Level
4) Residuals
5) Denoised Tree

The structuring and data preparation plays a very critical role in a data centric model.

III. THE SUPPORT VECOR REGRESSION (SVR) AND THE PROPOSED MODEL

The support vector regression model is a modified version of the support vector regression with a modification in the objective to loss function. The classification using the support vector machine (SVM) is depicted in figure 4.

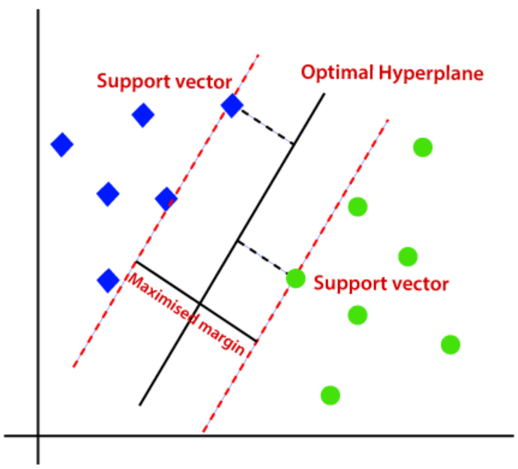

Fig.4The SVM [8]

The support vector regression can be designed as a least squares optimization (LS optimization) as:

for (i=1 : n)
{
Update weights and bias
And

$$Minimize\{\frac{e_1^2+e_2^2+\cdots\ldots\ldots e_n^2}{n} \quad \textbf{(8)}$$

}

The least squares minimization approach is the fastest and most stable approach to convergence. The iterative update of the support vectors keeps changing the bias and weights to minimize the least squares objective function. The parameters to be evaluated for the CTR prediction are:

1) Date and time
2) Product
3) Campaign
4) Product category
5) Webpage
6) User group
7) Gender
8) Age level
9) City (location)

The dependent variable is chosen as the occurrence of click (1) or non-click (0). The evaluation parameters are the accuracy and percentage errors given by:

$$error\% = \frac{false\ classifications}{total\ classifications} * 100 \quad \textbf{(9)}$$

$$Ac = 100 - error\% \quad \textbf{(10)}$$

IV. RESULTS

The results of the proposed system are evaluated in terms of the error % and the classification accuracy.

The training vector is designed as the training parameters along with the rank and quantile values of the outcomes to feed the SVR model. Once the system is trained, the testing is done based on the testing data. The data division has been done in the ratio of 70:30 based for training and testing. The add sample and click polarity are recorded for the computations. The tokenization (target formation) of the clicks are done as:

1) 1: Expected Click
2) 0: No Click
3) -1: Diverted click

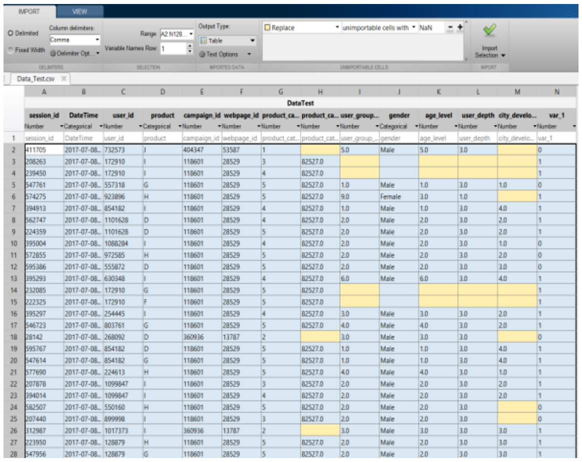

Fig.5 Raw Data

Figure 5 depicts the importing of the raw data

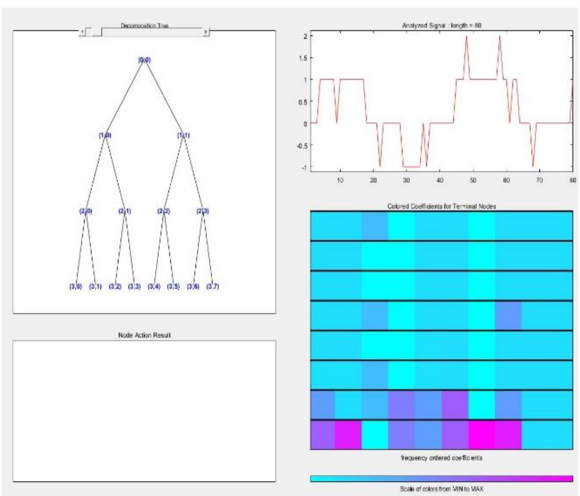

Fig.6 Initial Tree

Figure 6 depicts the initial tree for binary classification. The subsequent steps are to find the best level and the best tree for the given dataset.

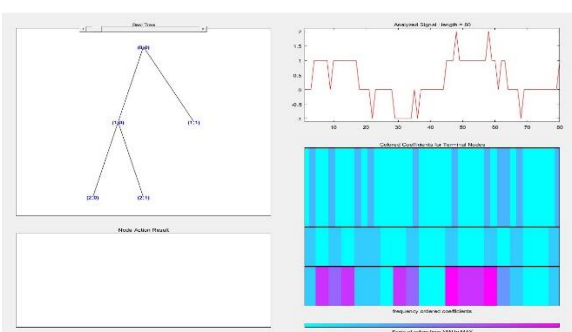

Fig.7 Best Tree

Figure 7 depicts the best tree among the possible bifurcations.

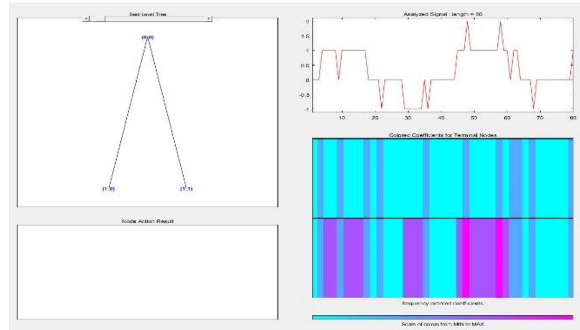

Fig.8 Best Level

Figure 8 depicts the best levelamong the possible bifurcations.

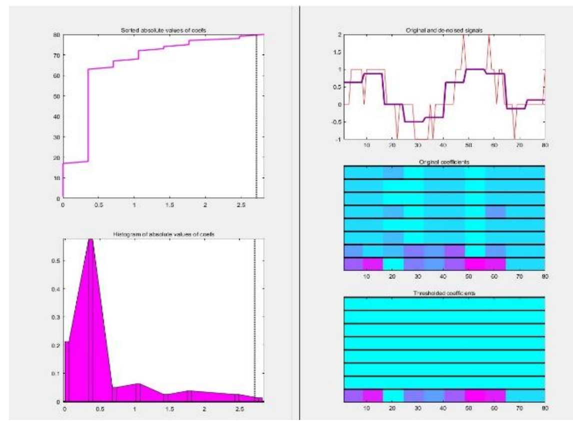

Fig.9 Denoised data using Shannon Entropy

Figure 9. depicts the denoised version of the data based on the Shannon entropy wherein the entropy is considered for smoothening operation.

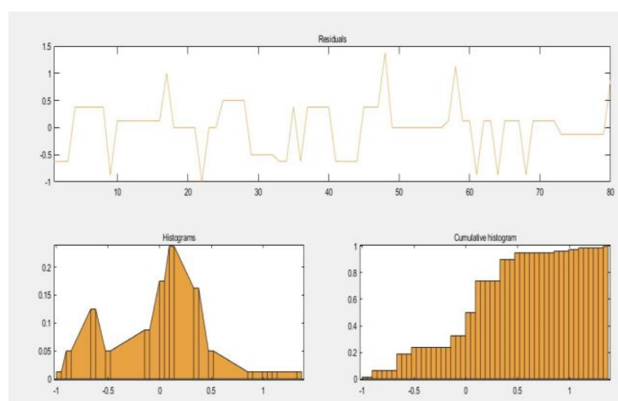

Fig.10 Histogram Analysis of Residuals

Figure 10 depicts the normal and cumulative histogram for the residuals of the decomposition.

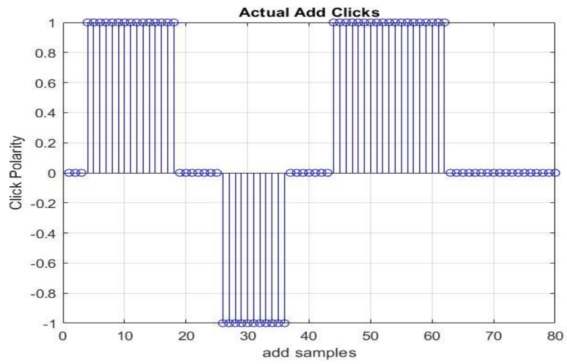

Fig.11 Actual Add Clicks

Figure 11 depicts the actual add click with three polarities of 1,-1 and 0.

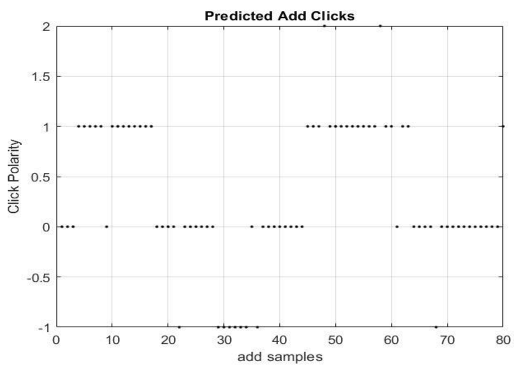

Fig.12 Predicted Add Clicks

Figure 12 depicts the predicted add clicks with three polarities of 1,-1 and 0.

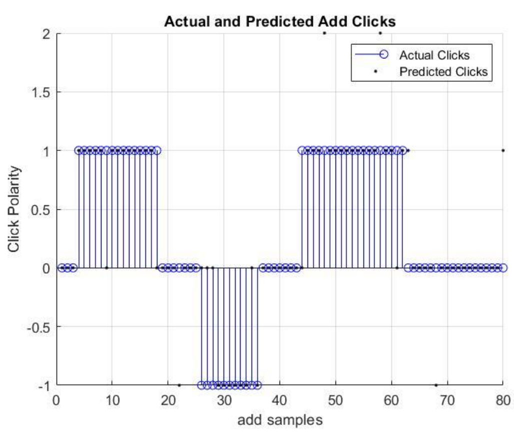

Fig.12 Predicted and Actual Clicks

Figure 12 depicts the predicted and actual clicks.

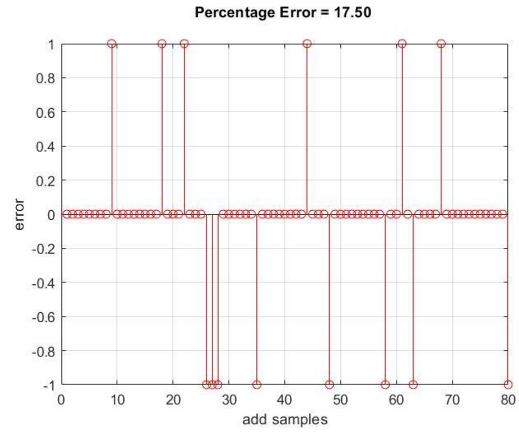

Fig.13 Errors and Percentage Error

Figure 13 depicts the sample wise errors and percentage errors. The errors are estimated with the condition of:

$$actual\ click \neq predicted\ click$$

The percentage error obtained in this approach is 17.50% and hence the accuracy of the system is 82.50%. This is significantly higher compared to the average accuracy of previous benchmark approach of 77%.

V. CONCLUSION

This paper presents a recursive binary tree partition algorithm employing wavelet trees for data preparation. Subsequently the data is fed to a support vector regression model to estimate add click through rate (CTR). Previous discussion have emphasized upon the CTR and its estimation for online advertising models. The performance of the designed system has been evaluated in terms of the error% and classification accuracy. It has been shown that the error% of the system is 17.5% and the classification accuracy is 82.5% which is higher compared to the existing benchmark approaches [1].

REFERENCES

[1] JA Choi, K Lim, “Identifying machine learning techniques for classification of target advertising”, ICT Express, Elsevier 2022, vol. 6, no. 3, pp. 175-180.
[2] M. Gan and K. Xiao, "R-RNN: Extracting User Recent Behavior Sequence for Click-Through Rate Prediction," in IEEE Access, 2021, vol. 7, pp. 111767-111777.
[3] Q. Wang, F. Liu, P. Huang, S. Xing and X. Zhao, "A Hierarchical Attention Model for CTR Prediction Based on User Interest," in IEEE Systems Journal, 2019., vol. 14, no. 3, pp. 4015-4024.

[4] L. Y. Akella, "Ad-Blockers — Rising threat to digital content: Business analytics study," 2017 International Conference on Big Data Analytics and Computational Intelligence (ICBDAC), 2017, pp. 324-32.

[5] G. Chauhan and D. V. Mishra, "Evaluating deep learning based models for predicting click through rate," 2019 4th International Conference on Computational Systems and Information Technology for Sustainable Solution (CSITSS), 2019, pp. 1-5.

[6] X Wang, G Hu, H Lin, J Sun, "A novel ensemble approach for click-through rate prediction based on factorization machines and gradient boosting decision trees", APWeb-WAIM 2019: Web and Big Data, Springer 2019, pp 152–162.

[7] Z Xiao, L Yang, W Jiang, Y Wei, Y Hu, "Deep multi-interest network for click-through rate prediction", CIKM '20: Proceedings of the 29th ACM International Conference on Information & Knowledge Management, ACM, 2021, pp.2265-2268.

[8] J Gligorijevic, J .Gligorijevic., D. Stojkovic, "Deeply supervised model for click-through rate prediction in sponsored search" Data Min Knowledge Discovery, Springer 2019, vol. 33, pp: 1446–1467.

[9] L. Zhang, W. Shen, J. Huang, S. Li and G. Pan, "Field-Aware Neural Factorization Machine for Click-Through Rate Prediction," in IEEE Access, 2019, vol. 7, pp. 75032-75040.

[10] X. Qu, L. Li, X. Liu, R. Chen, Y. Ge and S. -H. Choi, "A Dynamic Neural Network Model for Click-Through Rate Prediction in Real-Time Bidding," 2019 IEEE International Conference on Big Data (Big Data), 2019, pp. 1887-1896.

[11] S. Zhang, Z. Liu and W. Xiao, "A Hierarchical Extreme Learning Machine Algorithm for Advertisement Click-Through Rate Prediction," in IEEE Access, 2018, vol. 6, pp. 50641-50647.

[12] J Dhanani, K Rana, "Logistic Regression with Stochastic Gradient Ascent to Estimate Click Through Rate", Information and Communication Technology for Sustainable Development, Springer 2018, pp.319-326.

[13] X. She and S. Wang, "Research on Advertising Click-Through Rate Prediction Based on CNN-FM Hybrid Model," 2018 10th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC), 2018, pp. 56-59.

[14] H. Liu, X. Zhu, K. Kalish and J. Kayne, "ULTR-CTR: Fast Page Grouping Using URL Truncation for Real-Time Click Through Rate Estimation," 2017 IEEE International Conference on Information Reuse and Integration (IRI), 2017, pp. 444-451.

[15] C. Jie-Hao, L. Xue-Yi, Z. Zi-Qian, S. Ji-Yun and Z. Qiu-Hong, "A CTR prediction method based on feature engineering and online learning," 2017 17th International Symposium on Communications and Information Technologies (ISCIT), 2017, pp. 1-6.

[16] W. -Y. Zhu, C. -H. Wang, W. -Y. Shih, W. -C. Peng and J. -L. Huang, "SEM: A Softmax-based Ensemble Model for CTR estimation in Real-Time Bidding advertising," 2017 IEEE International Conference on Big Data and Smart Computing (BigComp), 2017, pp. 5-12.

[17] B Edizel, A Mantrach, X Bai, "Deep character-level click-through rate prediction for sponsored search", SIGIR '17: Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, 2017 pp. 305–314.

[18] K. Ren, W. Zhang, K. Chang, Y. Rong, Y. Yu and J. Wang, "Bidding Machine: Learning to Bid for Directly Optimizing Profits in Display Advertising," in IEEE Transactions on Knowledge and Data Engineering, 2017, vol. 30, no. 4, pp. 645-659.