Global Journal of Pure and Applied Mathematics.
ISSN 0973-1768 Volume 13, Number 6 (2017), pp. 2693-2702

http://www.ripublication.com

Fuzzy Logic Based Lateral Unruffled Treatment Time Prediction Algorithm by Using Big Data

[1]Prabhakaran V and [2]Nagarajan E

[1]P.G Scholar, [2]Assistant Professor

Department of Computer Science and Engineering
Sathyabama University, Chennai. India.

Abstract

Nowadays in hospitals, facing patient queue and overcrowding as a major issue. To solve this problem, many approaches are proposed but still issue is not completely resolve. Waiting for extended period in the queue is giving disturbance to the patient. To overcome this, we have proposed a system Treatment time prediction TTP supermodel for patient based on Random forest algorithm, to predict average waiting time for each treatment task for a patient. We use streaming data from various hospitals to obtain Treatment time supermodel for patient each task. Based on Streaming dataset or live data, the treatment time for each patient in the current line of each task is predicted. Based on Predicted average time, we developed Queue-recommendation (QR) system to predict efficient treatment plan for the patient. To demonstrate the productiveness and applicability of our proposed system is to minimize the waiting time of the patient in hospitals. We use an Apache Kafka, Apache Strom in Cassandra database implementation to achieve the goals.

Index Terms: Queue-Recommendation (QR) system, Treatment time prediction TTP algorithm, Streaming data.

I. INTRODUCTION

Formerly, many number hospitals are overfull and not effective in patient queue system. Facing issues like predicting wait time and its complicated to collect each patient need various operations such as checkup, various medical treatments and all small surgeries during Treatment. Most patients wait for long period of time in queues waiting for their turn to accomplish each task. In this paper we are focusing to help patient complete their treatment tasks in a foreseeable time and avoid overfull in hospital and queues.

II. OUR CONTRIBUTION

Extensive experimentation results and TTP algorithm mainly achieves high accuracy and presentation**.** The Treatment time Prediction for patient algorithm is trained based on Random Forest algorithm for each treatment task and predicted time. Queue recommended system is to recommends a capable treatment plan for each patient. Using streaming data to achieve accuracy of Predicting time for the patient waiting in the Wait line. We used fuzzy logic is to compare the consequences of using chronological and streaming data. Apache Cassandra database is used to store the hospital huge data's.

III. RELATED WORK

To increase the productiveness of data analysis with aspects, different methods of regression and classification algorithms are proposed. Self-adaptive induction of regression trees in Incremental regression tree learning method change at varies different rates and speeds in data streams. No other technique able to construct and regression trees to data streams [1].

Parallel boosted degeneration trees for Web search ranking in Multicore memory systems as well as to scattered setups in clusters and clouds[2].No much Relevance and utility in the predictable future.

Correlation based splitting criterion in multi branch decision tree in its used to find the best splitting variable and relevant splitting thresholds [3]. Not able to Splitting on more than one variable at each node.

A streaming parallel assessment tree Algorithm provides Best error rate using Streaming Parallel Decision Tree. Its provide similar data inaccuracy rate but not sequential data error data perfect [4].

L.Breiman [5] Flexible predictive modeling. Its right kind of randomness makes them accurate classifiers and repressor's. Less accurate than Boosting regression and slower in the runtime.

Force Action recognition via Discriminative Random Forest selection and Top-K Sub volume Search [6]. It improves the system with two factors. First, proposed random forest voting technique to calculate the scores of the importance points, which achieves a multiple orders-of-magnitude speed-up compared with the nearest-neighbor-based scoring method. Second, we developed a top-k search technique which detects multiple action instances concurrently with a single round of branch-and bound search[7].

Muller at the correlation concept is to split the variable. Tree is split into one or more branches at each node [8].To enhance our algorithm even more to use proposed classification applications.

Nunez et al is dealing with stream of data dynamic classification. It is dealing with the problem memory managing and variation of internal parameters [9].

Neural Networks depends on full single occurrence memory, in which method retains all past training are examples of regression algorithm [10]. Partial occurrence memory is oriented to deal with changes in original patterns and history training.

There is no existing research on hospital queuing management and treatment plan recommendations. For that reason, we propose a system with streaming & trained data to solve the problem of patient waiting in hospital queues and Queuing recommendations to tell the suitable treatment plan for each patient.

IV. DESIGN

Fig 1 shows the overall framework of the system. The system fetches live or streaming data from various checkups, x-ray, ct-scan and blood tests. And the data stores in Apache Cassandra Database in big data environment and real time data stream can handle apache kafka for messaging queue and apache storm can process distributed data's which customized programs for process huge time calculation process.

In Application Layer it introduces the TTP model for Treatment time prediction.**Fig.1**

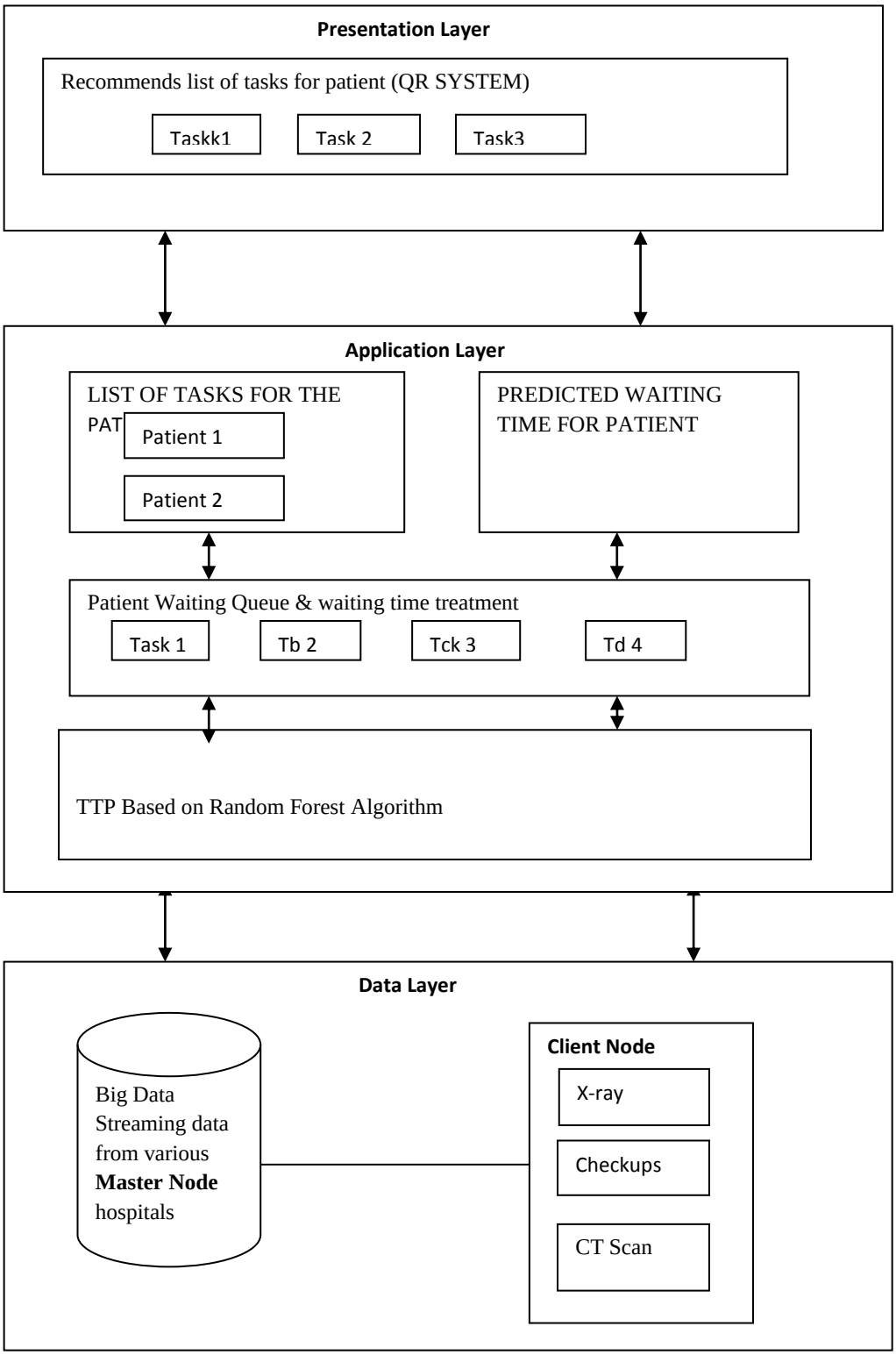

Fig.1: Proposed Framework

It pulls the data from kafka and applies some required manipulation.

At this point usually it like to get better accuracy data, so either send it to some NoSQL and relation database for additional BI calculations or query this NoSQL big database from any other system and waiting time of the patient for long period in the queue based on Random forest algorithm. To produce the Effective result we used QR system to give the better treatment plan for each patient.

A.APACHE PLATFORM:

Apache Kafka as a distributed and robust queue that can handle high volume data and enables to pass message from one end point to another.

Storm is not a queue. It is a system that has distributed real time processing abilities. Execute all kind of manipulations on real time data in parallel.

It handles big volume data which sends to Kafka storm.

B.TREATEMENT TASK PREDICTION ALGORTIHM:

TTP (Treatment task prediction) is improvised from Random Forest. This is to eliminate noisy and complex treatment hospital data.

1. Problem Definition and Data Preprocessing

Problem Definition:

Processing of more huge or unstructured data from hospitals is a big task. Most of the hospital data's are noisy and high dimensional, due to manual entry and many unexpected events during treatments, inconsistent and incomplete data, such as missing of patient age data, and gender etc, Treatment time with only start time. Fast Executing the TTP and QR system is tedious task.

Table 1: Aspects of Treatment data TTP

Aspect Name	Value range
Gender	"Male", "Female"
Age	Patient age
Department	Departments in hospital
Doctor Name	Doctors in hospital
Task Name	Treatment task in all treatment process
Start Time	Treatment task of the start time

End Time	Treatment task of the End Time
Week	Monday to Sunday
Time Range	0 to 23
Time Consumption	End time and start time such as CT scan

Data Preprocessing:

Collecting data from different task treatment

Dimension of the data should be same.

Calculate new value variables of the data .

Remove Noisy and inconsistent data.

Algorithm 1: TTP Algorithm based on RF Algorithm

Input:

Training datasets: $Train_s$;

No of CART trees: m

Output:

1:For i= 1 to m do ; create training subset trainis sampling(Trains);

2: create OOB subsetS (Trains , trainis);

3: create an empty CART tree;

4: for each independent variable xj in trainis do

5: calculate candidate split points vs xj;

6: for each vp in vs do

7: calculate the best split point (xj; vp) avg min

8: end for

9: append node Node (xj; vp) to hi;

10: split data for left branch LR(xj;vp) fxjxj _ vpg;

11: split data for right branch RY(xj;vp) fxjxj>vpg;

12: for each data R in LR (xj;vp); RR(xj;vp)g do

14: calculate 8 (vpLjxj) maxis 8(vijy);

15: if (8(vp(LjR)jxj) _ 8(vpjxj)) then

16: append subnodeNode(xj;vp(LjR)) to Node(xj;vp) asmulti-branch;

17: split data to two dig RL(xj;vpL) and RR(xj;vpR);

18: else

19: collect cleaned data for leaf node Dleaf (IL _ xj _ OL);

20: calculate mean value of leaf node;

21: end if

22: end for

23: remove xj from trainis;

24: end for

25: calculate accuracy CAi I (hi(x)Dy) I (hi(x)Dy)CPI (hi(x)Dz) for hiby testing S;

26: end for

27: *RFTTPH*(*X*;2*j*) 1*k*P*ki*D1 [*CAi_ hi*];

28: **return** *RFTTP.*

C. QR SYSTEM:

Queue Recommendations is to produce convenient treatment plan task for each with prediction time. Number of recommend task which is helpful for hospitals and also patients waiting in the queue for long a wait.

Algorithm 2 : Process of the Queuing Recommendation System:

Input:

Y:Current Patient Data Treatment

RFTTP : the trained TTP model based on the RFalgorithm.

Output:

St(Y): Predicted time with the recommended task.

1: create map *St(Y)HashMap<string*; *double >*;

2: **for** each *Taskx*in Y**do**

3: create array M*i*[] patients-in-waiting of *Taskx*;

4: **for** each patient M*ik*in M*i***do**

5: predict time consumption *TikTTPRF* ;

6: **end for**

7: calculate average waiting time *Ti* 1 *Wi*P*mk*D1 *Tik*;

8: append waiting time *St(Y)<Taskx*; *Ti>*;

9: **end for**

10: sort map *St(Y)* in an low to high order;

11: **for** each <*Taskx*; *Ti*> in *St(Y)***do**

12: **if** (*Taskx*has dependent tasks) **then**

13: put records of the dependent tasks before *Taskx*;

14: **end if**

15: **end for**

16: **return** *St(Y)*.

D. EXPERIMENTAL RESULTS WITH FUZZY LOGIC

1. Streaming data - Patient Treatment Start time and Complete time .

The input method web interface processing for Patient Treatment Time start and complete timing details. [Fig.2,3].

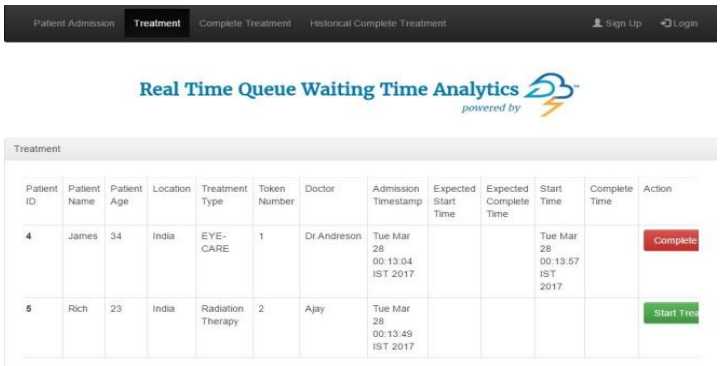

Fig.2: Treatment start and end time

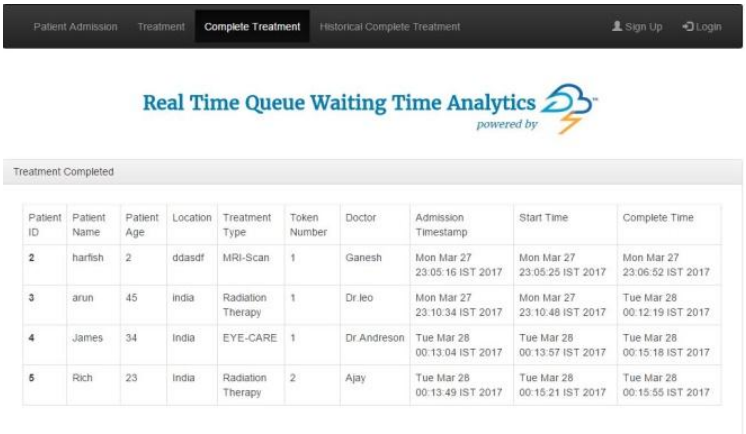

Fig.3 Treatment completed

2. Average waiting time for Patients

To evaluate the efficiency of our QR system with the average patients wait time in with-QR and without-QR system.

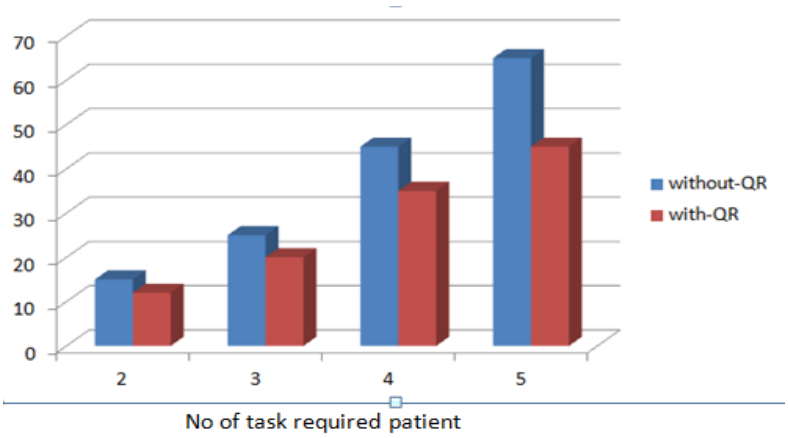

Fig 4: Patient Average waiting time.

3. Performance Evaluation of the historical and streaming data:

To demonstrate the effectiveness of both the proposed algorithm streaming data is more convenient recommendation with minimum path awareness.

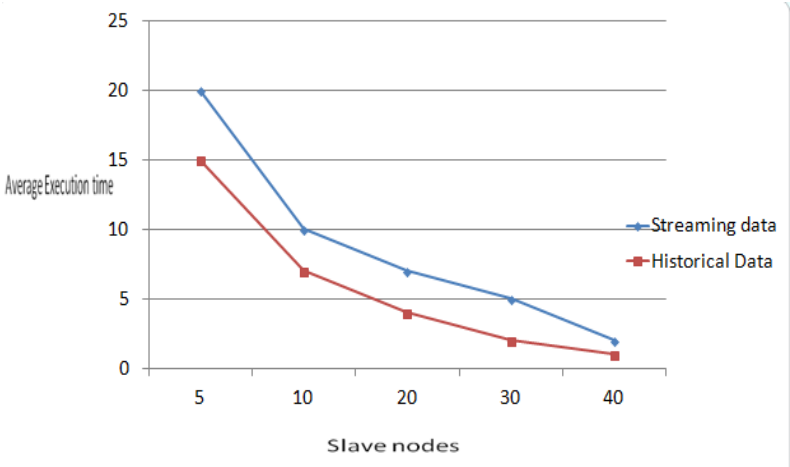

Fig 5: Performance evaluation of both QR and TTP

V: CONCLUSION

In this paper, QR and TTP model is based on Apace Big data environment to store large and live hospital data. Predict patient waiting time for each task based on TTP algorithm and recommending suitable treatment plan for each patient with high accuracy using QR system based on prediction time. Large no of historical hospital

data is overloaded with training data's so we used streaming data to show more efficient and better results.

REFERENCES:

[1] Fidalgo-Merino.R and Nunez.M, ``Self-adaptive induction of regression trees," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8,pp. 1659_1672, Aug. 2011.

[2] Tyree.S, Weinberger.K.Q, and Agrawal. K,``Parallel boosted regression trees for Web search ranking," in *Proc. 20th Int. Conf. World Wide Web (WWW)*, 2012, pp. 387_396.

[3] N. Salehi-Moghaddami, H. S. Yazdi, and H.Poostchi, ``Correlation based splitting criterion in multi branch decision tree," *Central Eur. J. Computer Sci.*, vol. 1, no. 2, pp. 205_220, Jun. 2011.

[4] G.Chrysos, P.Dagritzikos, I. Papaefstathiou, and A. Dollas, ``HC-CART: A parallel system implementation of data mining classification and regression tree (CART) algorithm on a multi-FPGA system,"*ACM Trans.Archit. Code Optim.*, vol. 9, no. 4, pp. 47:1_47:25, Jan. 2013.

[5] N. T. Van Uyen and T. C. Chung, ``A new framework for distributed boosting algorithm," in *Proc. Future Generate. Commun. Newt.(FGCN)*,Dec. 2007, pp. 420_423.

[6] Y. Ben-Haim and E. Tom-Tov, ``A streaming parallel decision tree algorithm," *J. Mach. Learn. Res.*, vol. 11, no. 1, pp. 849_872, Oct. 2010.

[7] L. Breiman, ``Random forests," *Mach. Learn.*, vol. 45, no.1, pp. 5_32,Oct. 2001.

[8] Muller W, Wysotzki F., Automatic construction of decision trees for classification, ANN OPER RES, 1994, 52, 231-24.

[9] P. Domingos and G. Hulten, "Mining high-speed data streams, in Proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2000, pp. 71–80.

[10] E. Ikonomovska, J. Gama, R. Sebasti˜Ao, and D. Gjorgjevik, "Regression trees from data streams with drift detection," in Proceedings of the Discovery Science 2009 DS 2009, ser. Lecture Notes in Artificial Intelligence, J. G. et al., Ed., vol. 5808. Berlin Heidelberg: Springer-Verlag, 2009, pp. 121–135.