International Journal of Scientific Engineering and Technology Research

ISSN 2319-8885
Vol.05,Issue.08,
April-2016,
Pages:1635-1638

www.ijsetr.com

Privacy Preserving Data Mining using Random Decision Tree Framework

SANJAY GANDHI MAMIDI[1], MD. AMANATULLA[2]

[1]PG Scholar, Dept of CSE, Nimra Institute of Science & Technology, AP, India.

[2]Assistant Professor, Dept of CSE, Nimra Institute of Science & Technology, AP, India.

Abstract: Present days we are using driven application based on distribute data. Here the challenge is when ever using the data across the boundaries, then it is difficult to determine how to collaborate effectively across organizational boundaries while increase utility of information. The local data gives the restricting the data usage, the cryptography-based work for privacy-preserving data mining task is slow with the large amount of data. Shows RDT work perfect with minimum computation and smaller cost. The fact that RDT suit for parallel and fully distributed architecture, Develops the protocols o implement privacy preserving RDT.

Keywords: Random Decision Tree(RDT), Bayes Optimal Classifier (BOC).

I. INTRODUCTION

Building and applying Data mining models leads to freely accessing the underlying data, but this is not realistic. Privacy and security concerns restrict the sharing or centralization of data. Privacy preserving data mining is the solution for these problem [1] .But deviation solutions like privacy preserving not strictly protect the data. Even cryptography solutions are not giving the Good results for dealing with big data. To provide the improved efficiency in this paper we propose randomization at the same time cryptography model. This paper gives efficient results over the existing models. And it gives the effective solution for the privacy preserving data - mining for big data challenge. The proposed system based on the Random Decision Tree(RDT)[2].The main advantage of the RDT is reusable , used in multiple data mining tasks, and it is the implementation of the Bayes optimal classifier (BOC) [2], RDT is more better than classification/tree building techniques because it is good for parallel task. Random Decision Tree is the best solution for distributed data mining in the concern of privacy preserving because

- RDT is providing structural mining against mining with big data. Only cryptography model gives slow mining but addition with RDT it gives good results
- RDT approach works for regression, classification, ranking and multi label classification
- Theanother advantage of RDT is giving differentially private data without loss of accuracy.

II. RANDOM DECISION TREES

The RDT builds the iso-depth RD trees and it outperforms the computational speed due to the inheritance property of random portioning for the tree construction [2].In the RDT algorithm. It divides into two stages. In the first phase it constructs the training data for building the trees and populates the nodes with training data. The depth of the tree based on learning data [2]. The depth of the tree equal to the half of the total no of features present in the data. In the process of tree construction starts the list of attributes from the data set, initially generate the tree by using select the attribute without using training data. The tree stops growing once the tree depth is reached, then, use the training data to update the statistics of each node. [3]Only leaf nodes need to be recorded for update the statistics. To update the statistics .We need to update the training data only once. The output values coming from the average values of the random trees.

TABLE I: The Distributed Weather Data Set

	—P1—		—P2—		
	outlook	temperature	humidity	windy	play
P1	sunny	hot	high	weak	no
	sunny	hot	high	strong	no
	overcast	hot	high	weak	yes
	rainy	mild	high	weak	yes
	rainy	cool	normal	weak	yes
	rainy	cool	normal	strong	no
	overcast	cool	normal	strong	yes
P2	sunny	mild	high	weak	no
	sunny	cool	normal	weak	yes
	rainy	mild	normal	weak	yes
	sunny	mild	normal	strong	yes
	overcast	mild	high	strong	yes
	overcast	hot	normal	weak	yes
	rainy	mild	high	strong	no

From the above example dataset (weather data set), distributed between two parties we assumed that It is horizontally portioned 1-7 are owned by party 1 (P1) and 8-14 are owned by party2 (P2).suppose it is vertically portioned outlook and temperature owned by party1 (P1) and humidity, windy owned by P2.

Copyright @ 2016 IJSETR. All rights reserved.

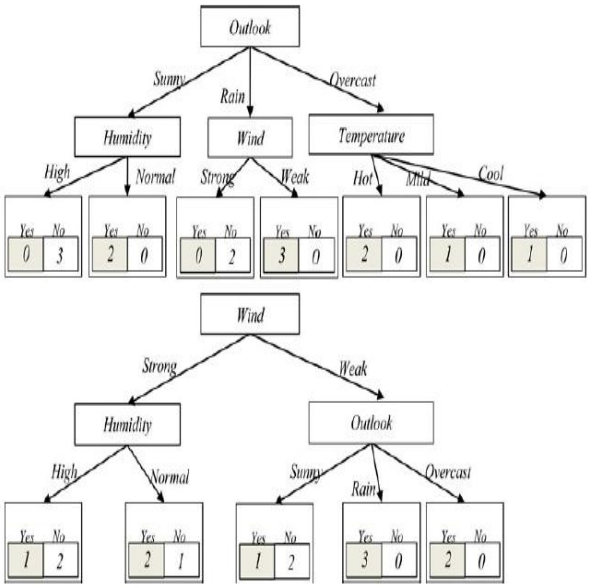

Fig.1. Two random decision trees for the weather data set.

We assume that two RDT are constructed(Fig.1) Suppose a new instance {sunny, mild, normal, weak} is to be classified. From the first random tree the prediction is (2,0) and with second random tree is (1,2) so the non normalized overall class distribution vector value for random decision tree is (1.5,1).

III. PROBLEM STATEMENT

The major problem of existing system is, in the distributed classification train the new instance And adding it to the distributed system. [4]In case of privacy preserving the adding of new instance is Challenging task because, the new instance not reveals the trained data and information beyond the learning from result of data. Let us consider Global data set D (T, R) where T is set Transaction and R is Global Schema. The problem formulated as fallows. We are portioned above data set as k sets (portions) and having the different instances and contains the similar type of information in the instance, and there are no conflict transactions. We know the schema is shared by all instances and take one class attribute C is known to all parties. Let's take one example like Bank, health insurance company, car insurance company have the different data from same group of people here the problem is no conflicting data between different parties.

IV. HORIZONTALLY PARTITION DATA

When the data is horizontally portioned, we are collecting data from different instances. Since all the parties shared same schema or instance of data, the solution is create random trees independently so all the parties securely share the information.[5] Actually basic RDT fallows class distribution at each non-leaf node level, but this information is enough at leaf node Basing on above scenarios we have two cases

- Each participant knows structure of tree.
- Each Participant don't know the structure of tree.

From the above two cases,case2 is difficult than case1 because the structure of tree is un known due to this problem other parties not compute the local leaf nodes every party must contact with the tree owner to compute the local leaf node values.Algorithm1 shows how to build the tree basing on the horizontally portioned data.

Algorithm 1 Building the random trees for horizontally partitioned data

Require: Transaction set T partitioned horizontally between sites $P_1, \ldots, P_k$
Require: n_i, the number of random trees to be created by each participant, such that $\sum_i n_i = m$, the total number of random trees
1: **while** Every party does not agree to all of the random trees **do**
2: {A secure electronic voting protocol can be used if no party should learn which party objects to any of the trees}
3: Each party generates its random trees
4: The structure of every tree is communicated to all of the parties
5: **end while**
6: **for** each tree T_j **do**
7: Each party P_i locally computes the class distribution vectors for each leaf node in T_j
8: Each party P_i encrypts the class distribution vectors for all leaf nodes in T_j using the threshold additively homomorphic encryption system and sends to all other parties
9: All parties then multiply the corresponding encrypted class distribution vector elements they receive for each leaf node to get the encrypted global value for that node
10: **end for**

All parties locally compute the class distribution for leaf nodes for all the trees. This is due to homomorphism property and all the trees in encrypted format. It is collaboratively decrypted and average to get actual distribution vector. [6]Getting the sum not reveals more than getting the total number of trees. Algorithm2 gives the specific steps.

Algorithm 2 classifyInstanceHoriz(instId)

Require: m, the number of random trees built (let $nodeid_j$ represent the root node of the jth tree)
1: **for** $i = 1 \ldots p$ **do**
2: Randomly choose r from the range of random numbers for the cryptographic system
3: $lv_i \leftarrow Encrypt(0, r)$ {Random encryption of 0}
4: **end for**
5: **for** $j = 1 \ldots m$ **do**
6: $currstats \leftarrow$ encrypted $classdistributionvector$ for instance $instId$ as per tree j
7: **for** $i = 1 \ldots p$ **do**
8: $lv_i \leftarrow lv_i * currstats_i$
9: **end for**
10: **end for**
11: Collaboratively decrypt each lv_i and divide by m to get actual statistics

From the weather dataset horizontal partitioning. takes the data from all the attributes in both parties one and two. Suppose incase of party1 outlook, temperature, windy,

humidity attributes takes into consideration but Trees are build based on the 1 to 7 rows only and remaining for party 2.

TABLE II: Horizontal Partitioning 1-7 Rows for Initial Tree Of Portion P1

	—P1—		—P2—		
	outlook	temperature	humidity	windy	play
P1	sunny	hot	high	weak	no
	sunny	hot	high	strong	no
	overcast	hot	high	weak	yes
	rainy	mild	high	weak	yes
	rainy	cool	normal	weak	yes
	rainy	cool	normal	strong	no
	overcast	cool	normal	strong	yes

TABLE III: Horizontal Partitioning. 8-14 Rows For Initial Tree Of Patrician P1

P2	sunny	mild	high	weak	no
	sunny	cool	normal	weak	yes
	rainy	mild	normal	weak	yes
	sunny	mild	normal	strong	yes
	overcast	mild	high	strong	yes
	overcast	hot	normal	weak	yes
	rainy	mild	high	strong	no

V. VERTICALLY PARTITION DATA

In the vertical partitioning the tree constructed based on the vertical data but not with the full attributes random tree constructing based on the party1 and party2.

TABLE IV: RDT Formed Based on P1

	—P1—	
	outlook	temperature
P1	sunny	hot
	sunny	hot
	overcast	hot
	rainy	mild
	rainy	cool
	rainy	cool
	overcast	cool
P2	sunny	mild
	sunny	cool
	rainy	mild
	sunny	mild
	overcast	mild
	overcast	hot
	rainy	mild

Each party collects data from different set of attributes,[7] parties cannot independely construct random trees and they are share the attribute information. Here also we have two cases.

- Attribute information shared by all parties and can independently construct random trees.
- Attribute information not shared by the parties.
- In the horizontal partitioning they reveal the potential sensitive information but in case of vertical it is not possible.

In case of full distributed trees total number of random trees known by the every site let us say M. [8]The algorithm creates random trees (algorithm 3) to build the random trees of depth n/2 where n is the Total number of attributes Therefore we fix the depth of tree as n/2 in algorithm 3.

TABLE V: RDT Formed Based on P2

—P2—		
humidity	windy	play
high	weak	no
high	strong	no
high	weak	yes
high	weak	yes
normal	weak	yes
normal	strong	no
normal	strong	yes
high	weak	no
normal	weak	yes
normal	weak	yes
normal	strong	yes
high	strong	yes
normal	weak	yes
high	strong	no

Algorithm 3 CreateRandomTrees

Require: Transaction set T partitioned vertically between sites $P_1, \ldots, P_k$
Require: P_i holds n_i attributes
Require: p class values, $c_1, \ldots, c_p$, with P_k holding the class attribute
Require: m, the number of random trees to build
1: All parties together compute $n = \sum_i n_i$ using the secure sum protocol [10]
2: $depth \leftarrow n/2$ {The depth of the random trees}
3: **for** $i = 1 \ldots m$ {Build the ith tree} **do**
4: $level \leftarrow 1$
5: $nodeId_i \leftarrow BuildTree(level, depth)$
6: **end for**

VI. CONCLUSION AND FUTURE WORK

We have showed the knowledge of efficient distributed privacy preserving is truly feasible. And we observe that privacy is the important aspect n distributed data sharing. We dealing the distributed data that either horizontally or vertically across multiple sites and performing data mining task. [9]Since RDT used to create and generate accurate and sometimes better model with Smaller cost. We applying privacy preserving with RDT gives good results in this paper as compared to existing results and strong privacy with less computation We are looking to apply RDT with Privacy preserving model when participants known or unknown about structure of tree.

VII. REFERENCES

[1] J. Vaidya, C. Clifton, and M. Zhu, Privacy-Preserving Data Mining.ser. Advances in Information Security first ed., vol. 19, Springer-Verlag, 2005.

[2] W. Fan, H. Wang, P.S. Yu, and S. Ma, "Is Random Model Better? On Its Accuracy and Efficiency," Proc. Third IEEE Int'l Conf. Data Mining (ICDM '03), pp. 51-58, 2003.

[3] W. Fan, J. McCloskey, and P. S. Yu, "A General Framework for Accurate and Fast Regression by Data Summarization in Random Decision Trees," Proc. 12th ACM

SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD '06), pp. 136-146, 2006.

[4] X. Zhang, Q. Yuan, S. Zhao, W. Fan, W. Zheng, and Z. Wang, "Multi-Label Classification without the Multi-Label Cost," Proc. SIAM Int'l Conf. Data Mining (SDM '10), pp. 778-789, 2010.

[5] A. Dhurandhar and A. Dobra, "Probabilistic Characterization of Random Decision Trees," J. Machine Learning Research, vol. 9, pp. 2321-2348, 2008.

[6] G. Jagannathan, K. Pillaipakkamnatt, and R.N. Wright, "A Practical Differentially Private Random Decision Tree Classifier," Proc. IEEE Int'l Conf. Data Mining Workshops (ICDMW '09), pp. 114-121, 2009.

[7] J. Vaidya, C. Clifton, M. Kantarcioglu, and A.S. Patterson, "Privacy-Preserving Decision Trees over Vertically Partitioned Data," ACM Trans. Knowledge Discovery from Data, vol. 2, no. 3, pp. 1-27, 2008.

[8] O. Goldreich, "General Cryptographic Protocols," The Foundations of Cryptography, vol. 2, pp. 599-764, Cambridge Univ. Press, 2004.

[9] D. Gritzalis, Secure Electronic Voting.ser. Advances in Information Security first ed., vol. 7, Springer-Verlag, 2003.

Author Profile:

Sanjay Gandhi Mamidi is currently pursuing his M.Tech(CSE) in Computer Science and Engineering Department,in Nimra Institute of Science & Technology, Ibrahimpatnam. He received his MCA in Pydah College of Engineering and Technology, Visakhapatnam.