Decision Tree Classification on Outsourced Data

Koray Mancuhan kmancuha@purdue.edu

Chris Clifton clifton@cs.purdue.edu

PURDUE UNIVERSITY – COMPUTER SCIENCES – QATAR UNIVERSITY

Motivation

Data Outsourcing ->Popular approach recently, many data storage services:

- Amazon Cloud Services
- Google Cloud Storage
- Dropbox

What about individuals' privacy in outsourced data records?

- Many privacy standards (l-diversity, k-anonymity, differential privacy)
- Many privacy ensuring data outsourcing/publishing methods (suppression/generalization, anatomization …)
- Somehow solved

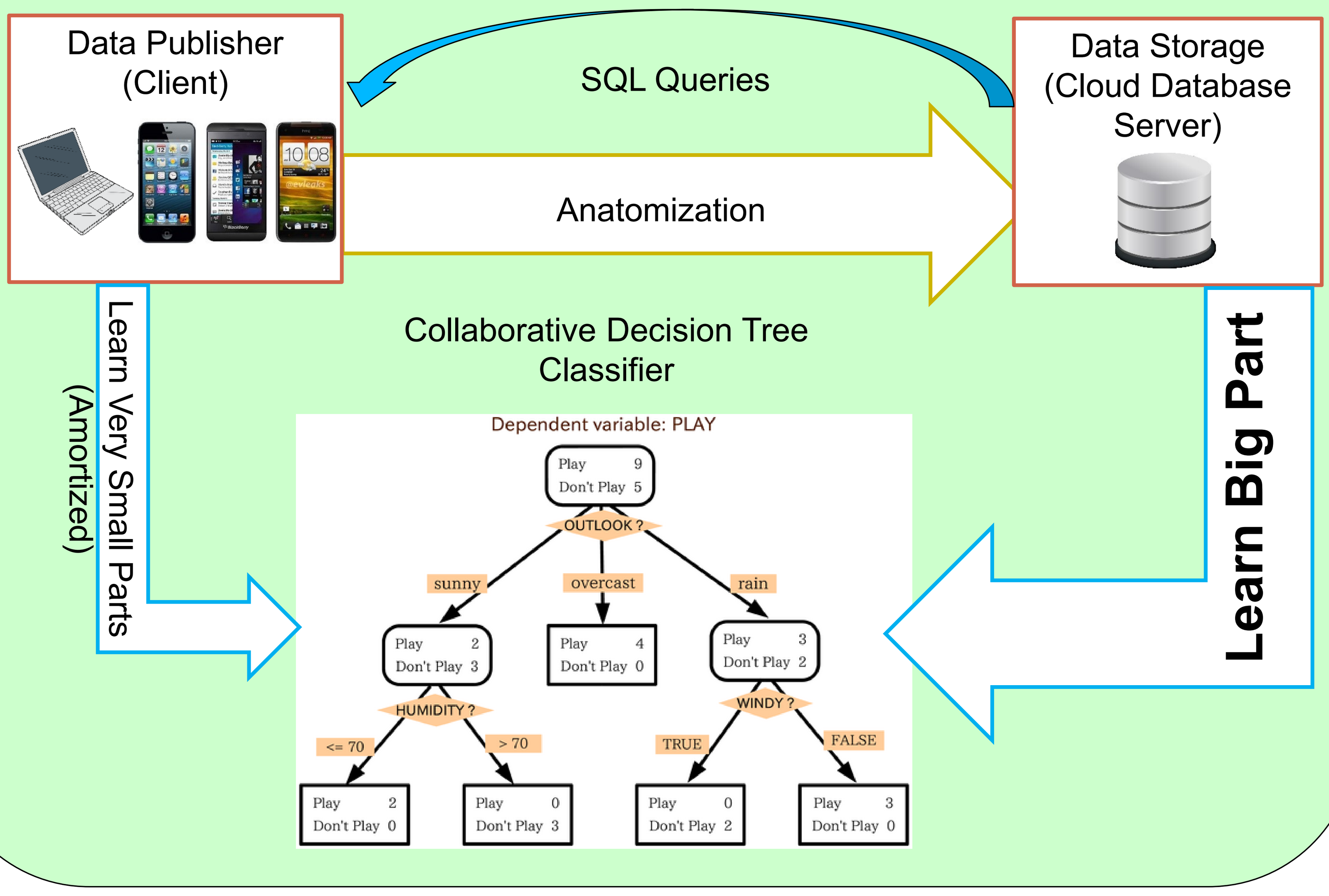
Can we extract useful information from outsourced data records? (Outsourcing occurs according to anatomization in this question)

- Data querying solved
- Data Analytics unsolved?

Our Contribution

Learn decision tree from outsourced data that is in anatomization form

- Low memory and execution time overhead for data publisher
- Most learning effort on cloud server



Anonymization

k-anonymity → Each group has at least 2 tuples e.g., k = 2 here

l-diversity → P(Disease of A = X) < 1/l e.g., l = 2 here

t-closeness → Sensitive values in each group should reflect the original distribution of sensitive values

Anatomy Model

Separate table into two tables, quasi-identifier (QIT) and sensitive table (ST) instead of generalizing records in the same group.

Age	Address	GID
20	Dayton	1
22	Richmond	1
…..	………	2

Quasi-identifier table (QIT)

GID	Disease
1	Cold
1	Fever
2	…..

Sensitive table (ST)

Related Work

SQL Queries over a cloud database based on anatomy model (Nergiz et al.).

Patient (P)	Age (A)	Address (AD)	GID (G)	SEQ (S)
Ike	41	Dayton	1	1
Eric	22	Richmond	1	2
Olga	30	Lafayette	2	3
Kelly	35	Lafayette	2	4
Faye	24	Richmond	3	5
Mike	47	Richmond	3	6
Jason	45	Lafayette	4	7
Max	31	Lafayette	4	8

$Patient_{QIT}$ = Quasi-identifier table

H(SEQ)	GID (G)	Disease (D)
$H_{k2}(1)$	1	Cold
$H_{k2}(2)$	1	Fever
$H_{k2}(3)$	2	Flu
$H_{k2}(4)$	2	Cough
$H_{k2}(5)$	3	Flu
$H_{k2}(6)$	3	Fever
$H_{k2}(7)$	4	Cough
$H_{k2}(8)$	4	Flu

$Patient_{SNT}$ = Sensitive table

- Selection (group by as well), insertion, deletion, update operations

Collaborative Decision Tree Learning

Proposal: Cloud Database Server builds the base decision tree from quasi-identifier table. Data publisher makes sub trees on each leaf using sensitive table and quasi-identifier table. A collaborative decision tree learning (cdtl) from data publisher and data storing party:

- Sub trees are made on-the-fly.
- Small number of instances in base decision tree leaves: *Memory Size Requirement Reduction, Execution Time Reduction*
- Base tree complexity and quasi-identifier tables predictor power is an issue!

Cloud Database Server

A	AD	G	S
41	Dayton	1	1
22	Richmond	1	2
30	Lafayette	2	3
35	Lafayette	2	4
24	Richmond	3	5
47	Richmond	3	6
45	Lafayette	4	7
31	Lafayette	4	8

H(SEQ)	G	D
$H_{k2}(1)$	1	Cold
$H_{k2}(2)$	1	Fever
$H_{k2}(3)$	2	Flu
$H_{k2}(4)$	2	Cough
$H_{k2}(5)$	3	Flu
$H_{k2}(6)$	3	Fever
$H_{k2}(7)$	4	Cough
$H_{k2}(8)$	4	Flu

AD=Dayton

A	AD	G	S
41	Dayton	1	1

AD=Richmond

A	AD	G	S
22	Richmond	1	2
24	Richmond	3	5
47	Richmond	3	6

AD=Lafayette

A	AD	G	S
30	Lafayette	2	3
35	Lafayette	2	4
45	Lafayette	4	7
31	Lafayette	4	8

Client

A	AD	D
30	Lafayette	Flu
35	Lafayette	Cough
45	Lafayette	Cough
31	Lafayette	Flu

D=Flu

A	AD	D
30	Lafayette	Flu
31	Lafayette	Flu

Label: <35

D=Cough

A	AD	D
35	Lafayette	Cough
45	Lafayette	Cough

Label: >=35

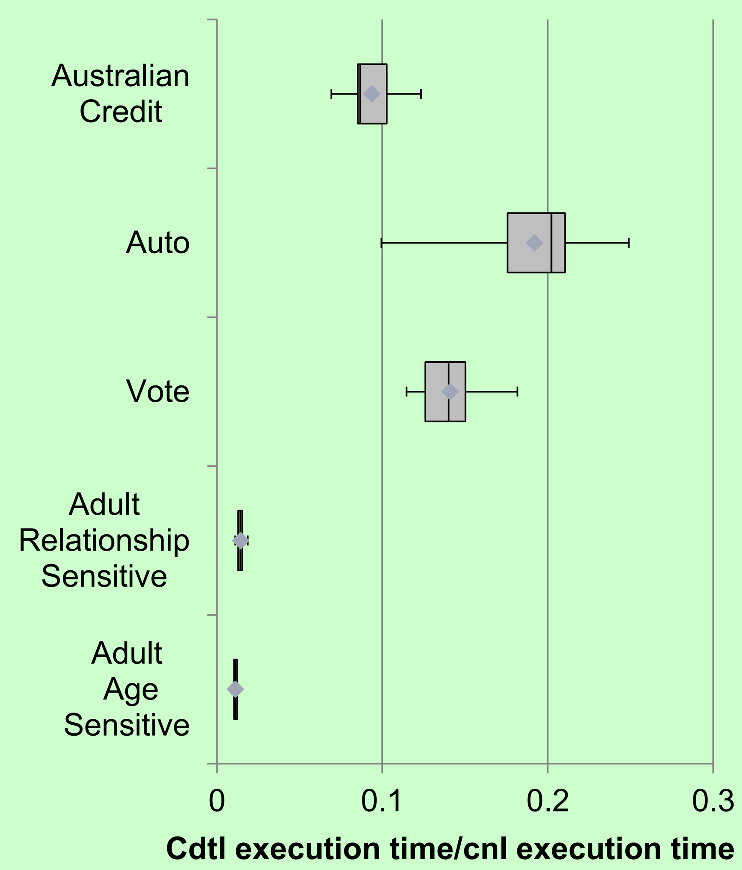
- *All sub trees are encrypted!*
- *All sub trees are stored at cloud database server!*

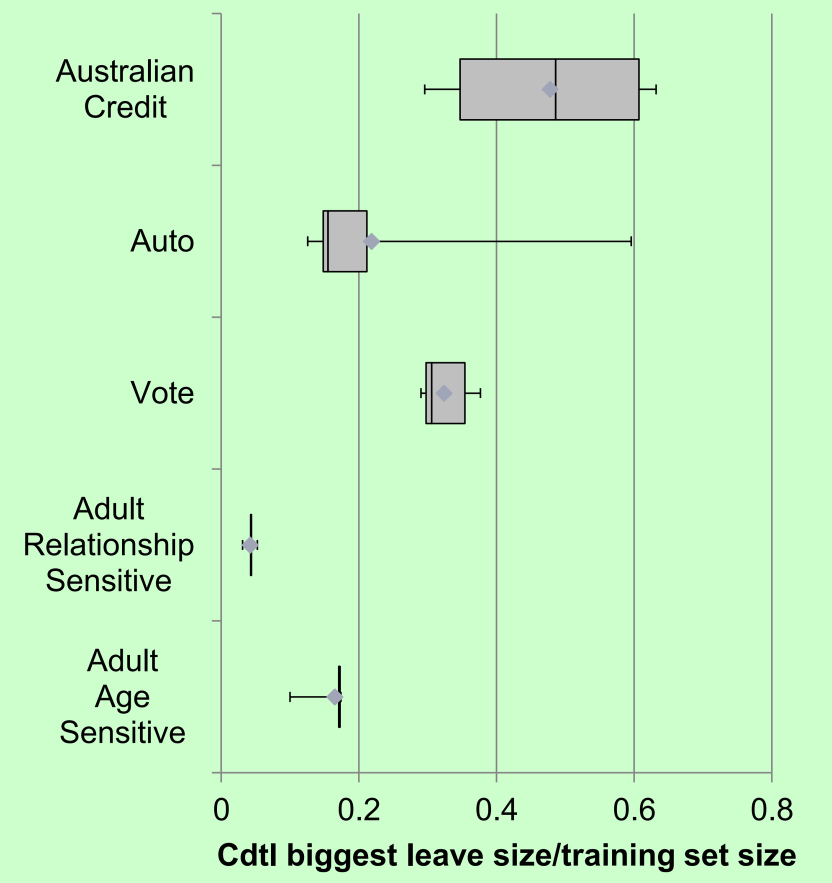
Other straightforward alternatives:

- Client Naïve Learning (cnl): Client retrieves the QIT and ST tables, rebuilds the original data and learns a decision tree. This decision tree is then encrypted and stored in Cloud Database Server.
- Cloud Database Server Learning (cdbsl): Cloud database server learns a base decision tree from quasi-identifying table. Client never makes a modification to this model.

Experiments and Results

- Four datasets from the UCI collection: adult, vote, autos and Australian credit
- Metrics: Accuracy, Execution Time Savings (ets) and Memory Savings (ms)

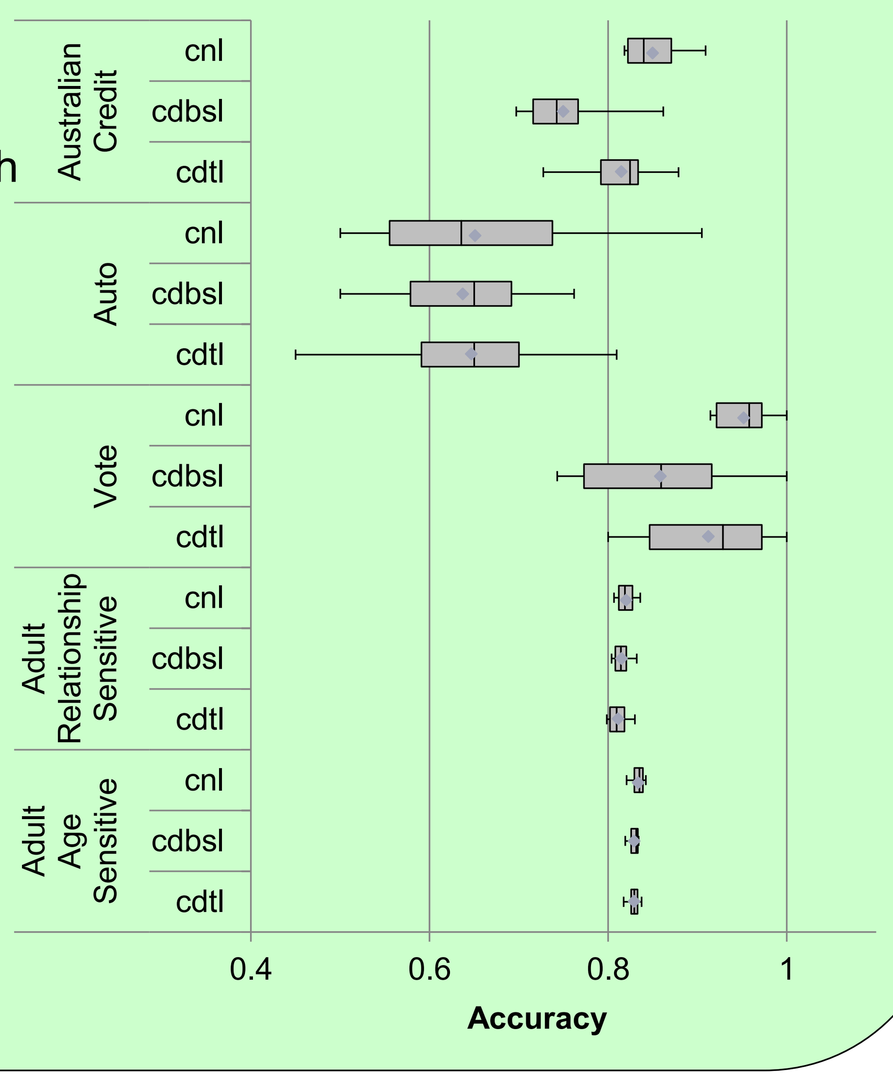




- 10 fold cross validation on each dataset

$$ets = \frac{cdtl\ execution\ time}{cnl\ execution\ time}$$

$$ms = \frac{cdtl\ memory\ requirement}{cnl\ memory\ requirement}$$



This publication was made possible by the support of the NPRP grant 09-256-1-046 from the Qatar National Research Foundation.