Towards Machine Learning Based Access Control

Ph.D. Dissertation Defense

Mohammad Nur Nobi
Department of Computer Science
The University of Texas at San Antonio

Committee:
Ravi Sandhu, Ph.D., Co-Chair
Ram Krishnan, Ph.D., Co-Chair
Palden Lama, Ph.D.
Wei Wang, Ph.D.
Xiaoyin Wang, Ph.D.

July 08, 2022

Introduction

- Access Control
  - The decision to permit or deny a user access to a resource
  - User: a human user, a process, an application, etc.
  - Resource: network, data, application, service, etc.
- There are many mainstream classical approaches for access control
  - Access Control Lists (ACLs), Role Based Access Control (RBAC), Attribute Based Access Control (ABAC), Relationship Based Access Control (ReBAC), etc.
- These approaches have their benefits and numerous advancements over time

Issues in Classical Approaches (ABAC)

I·C·S
The Institute for Cyber Security

C·SPECC
Center for Security and Privacy Enhanced Cloud Computing

Attribute Engineering

- An expert designs attributes based on the metadata
- E.g., 'status' attribute is engineered from 'spending' and 'credit' history

Policy Engineering (Policy Mining)

- To design policy through a manual or automated process
- E.g., <status = 'platinum', type='secured'> <access = 'read, write'>

Generalization

- Focus on capturing given access control state
- E.g., Knowing Alice's access, is it possible to determine Bob's access?

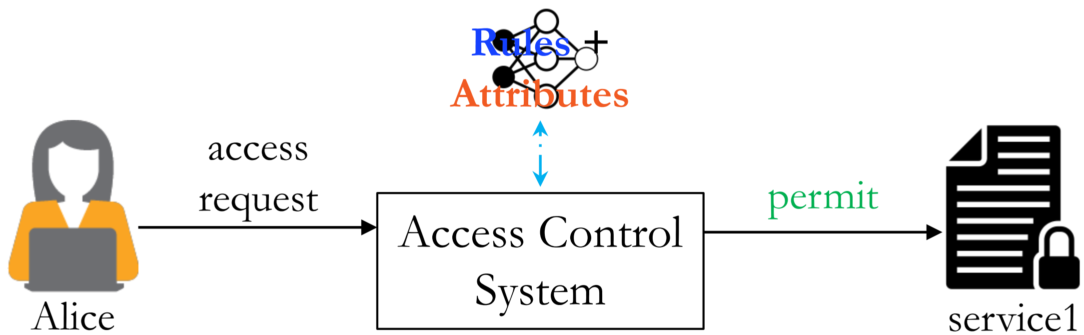
Attribute and Policy Update (administration)

- Revoke existing access or introduce a new access to existing users
- Depends on human, error-prone

UTSA Computer Science

Machine Learning in Access Control (Deep Learning)

- Could it learn from existing access control state of the system?
- Could it learn directly from the metadata?
- Could it make access control decisions that are accurate and generalize better?



- Obviates the need for related procedures
  - Attribute Engineering and Assignments
  - Policy Engineering
- Ease policy updates (Administration)

Thesis Statement

A deep neural network can **precisely learn** the access control state of a large-scale, complex, and dynamic system, **generalize enough** to make accurate decisions for unseen access control requests and **ease access control administration** by employing processes with **minimal human involvement**.

The Institute for Cyber Security

Summary of Contributions

Center for Security and Privacy Enhanced Cloud Computing

Machine Learning Based Access Control (MLBAC)

Comprehensive Literature Review : ML in Access Control

Operational Model of MLBAC

DLBAC (prototype, interpretation)

Administration of MLBAC

Adversarial Attacks in DLBAC

Implementation and Evaluation of DLBAC

UTSA Computer Science

I·C·S
The Institute for Cyber Security

Section-1

C·SPECC
Center for Security and Privacy Enhanced Cloud Computing

Machine Learning Based Access Control (MLBAC)

Comprehensive Literature Review : ML in Access Control

Operational Model of MLBAC

Administration of MLBAC

DLBAC (prototype, interpretation)

Adversarial Attacks in DLBAC

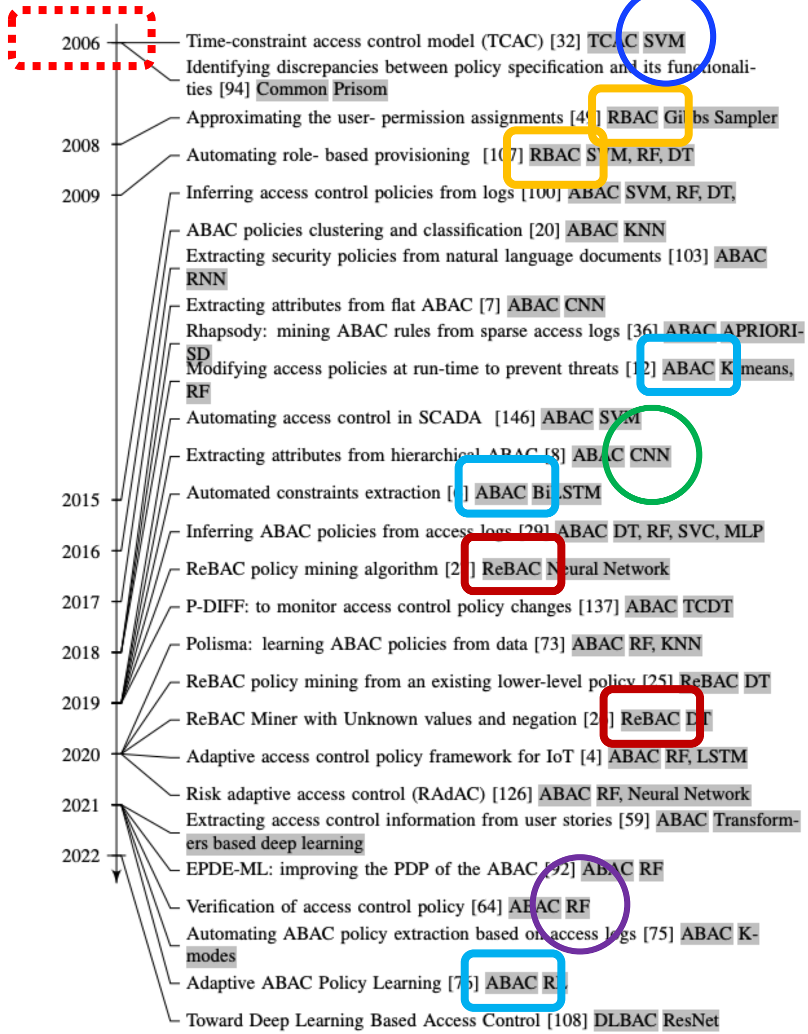
Implementation and Evaluation of DLBAC

UTSA Computer Science

Timeline of ML in Access Control

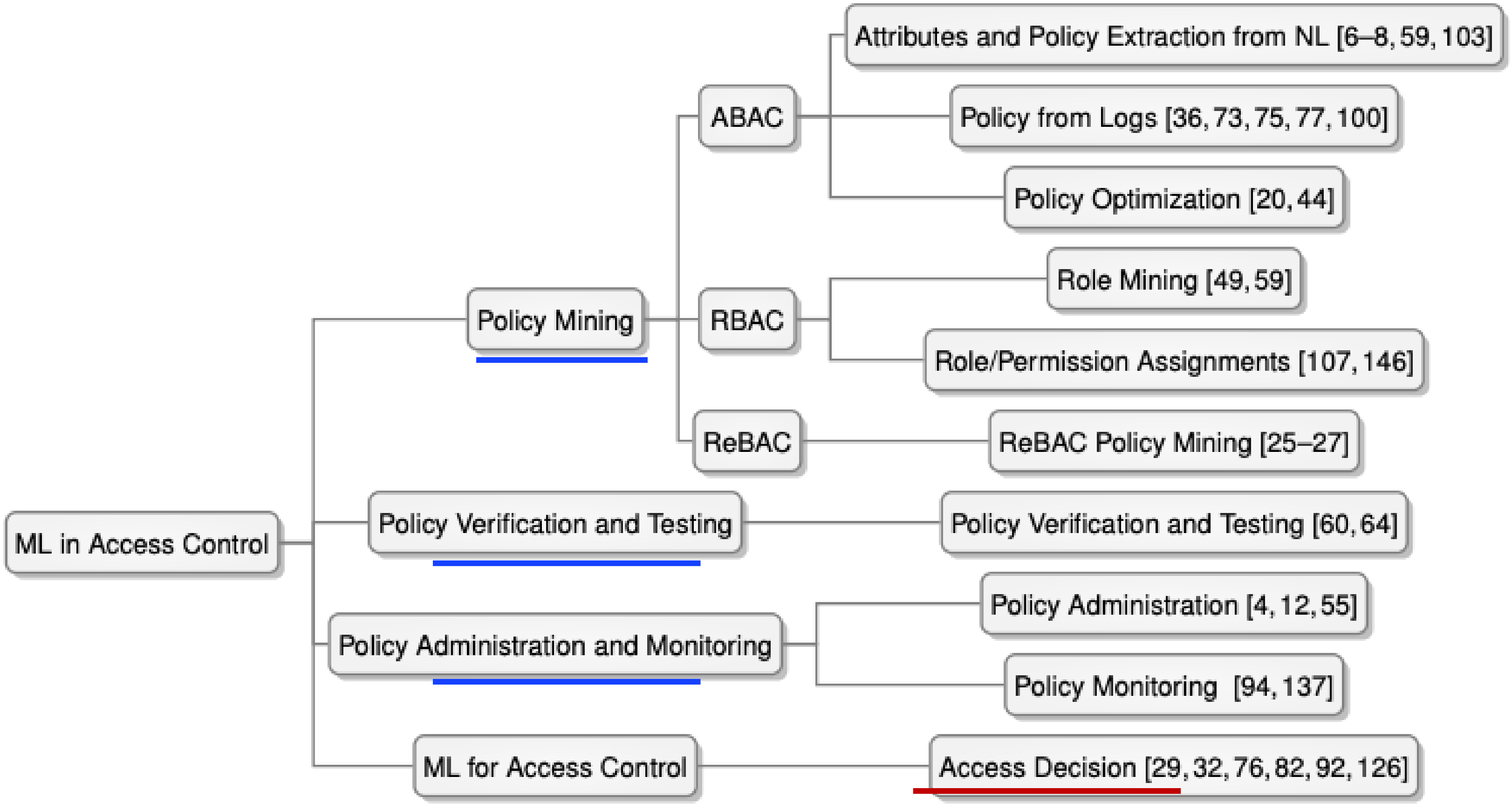
I·C·S
The Institute for Cyber Security

C·SPECC
Center for Security and Privacy Enhanced Cloud Computing



UTSA Computer Science

Taxonomy of ML in Access Control



Publicly Available Datasets for Access Control

I·C·S
The Institute for Cyber Security

C·SPECC
Center for Security and Privacy Enhanced Cloud Computing

Name	Publish Year	Reference	Type	Description
IBM-CM	2004	IBM [1]	Access Policies	Natural language access control policy
University-Data	2005	Fisler et al. [46]	Access Policy	Central grades repository system for a university
Wikipedia	2009	Urdaneta et al. [133]	Access Logs	Access request traces from Wikipedia
AmazonUCI	2011	UCI Repository [11]	Access Logs	Access data of Amazon employees
iTrust	2012	Meneely et al. [99]	Access Policies	Natural language access control policy
CyberChair	2012	Stadt et al. [135]	Access Policies	Natural language access control policy
Collected-ACP	2012	Xiao et al. [138]	Access Policies	Natural language access control policy collected from multiple sources
Amazon-Kaggle	2013	Kaggle [10]	Access Logs	Two years historical access data of Amazon employees (12000 users and 7000 resources)
eDocument	2014	Decat et al. [41]	Access Policy	e-document case study
Workforce	2014	Decat et al. [42]	Access Policy	Workforce management case study
SCADA-Intrusion	2015	Turnipseed et al. [132]	SCADA Data	SCADA dataset for intrusion detection system
Dalpiaz	2018	Dalpiaz et al. [38,39]	User Stories	Over 1600 user stories from 21 web applications
Incident	2018	Amaral et al. [9]	Event Logs	Event log from an incident management process

No attributes

NL Policy Related

Attributes extraction

UTSA Computer Science

Section-1 Summary

- ML in Access Control is nothing new
  - To optimize the underlying process
  - Evaluating potential to infer policy
- Lack of generalized system
  - Target specific application
- Lack of good datasets
- No discussion about ML model's vulnerabilities

Section-2

Machine Learning Based Access Control (MLBAC)

Comprehensive Literature Review : ML in Access Control

Operational Model of MLBAC

DLBAC (prototype, interpretation)

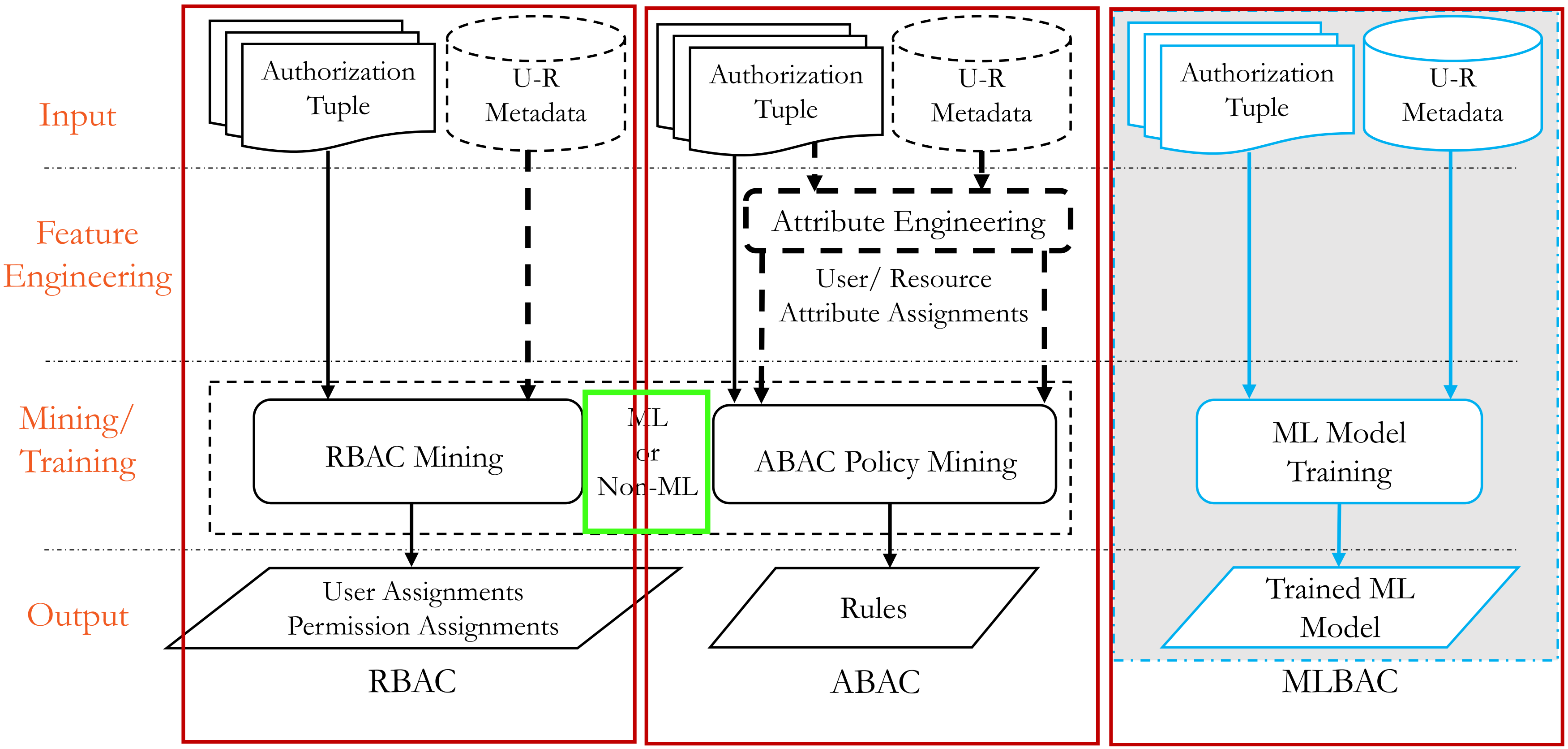
Administration of MLBAC

Adversarial Attacks in DLBAC

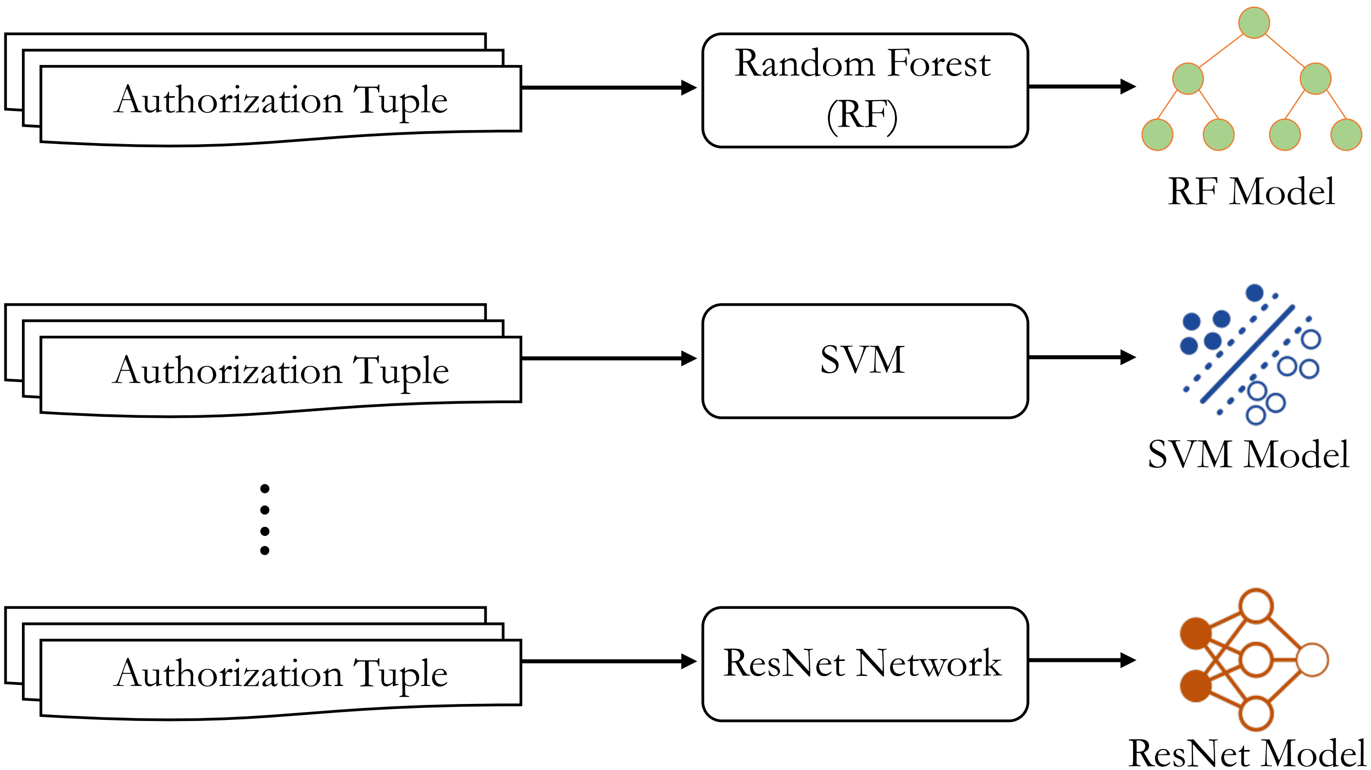
Implementation and Evaluation of DLBAC

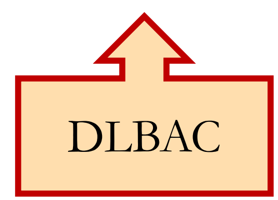
Operational Model Of Machine Learning Based Access Control

Authorization Tuple <Alice, projectA, {read, write}>



Candidate MLBAC Models





We create a DLBAC instance: **DLBAC$_\alpha$**

Syntax of DLBAC$_\alpha$ Dataset

User/ Resource metadata

User: Alice

rank	team	project		join date
developer	dev	projA	…	Nov 2012

Resource: projectA

type	team	project		size
source	dev	projA	…	medium

Operations: op1, op2, op3, op4

Authorization Tuple: <Alice, projectA, {op1, op3}>

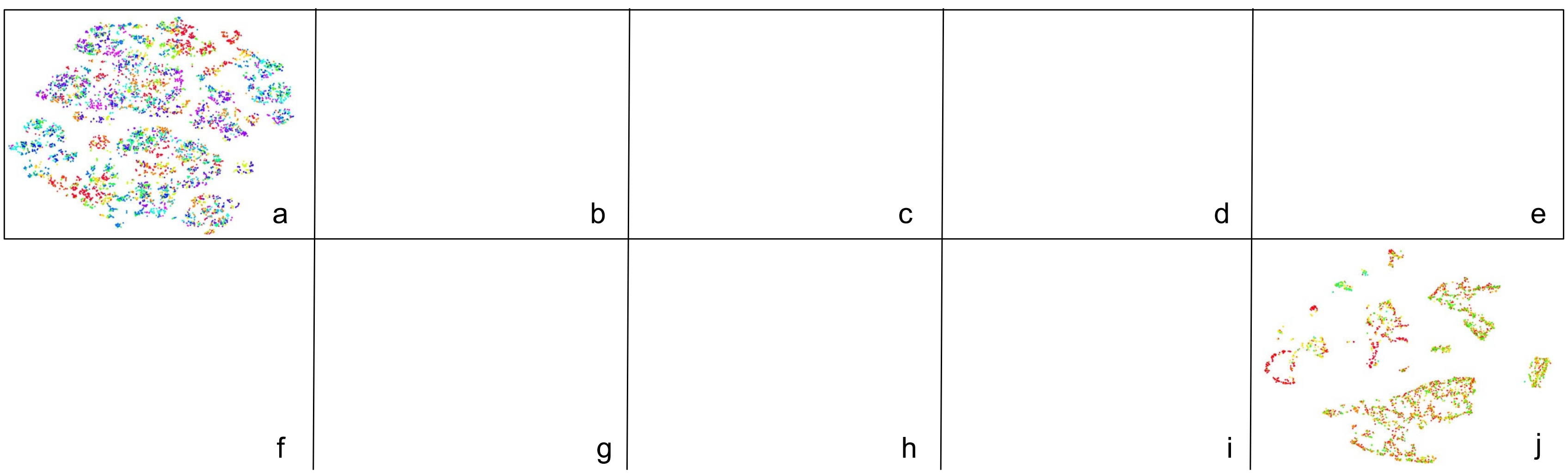
developer	dev	projA	…	Nov 2012	source	dev	projA	…	medium	1	0	1	0

User metadata values | Resource metadata values | Access to operations

A **dataset** for DLBACα is the collection of such authorization tuples (samples)

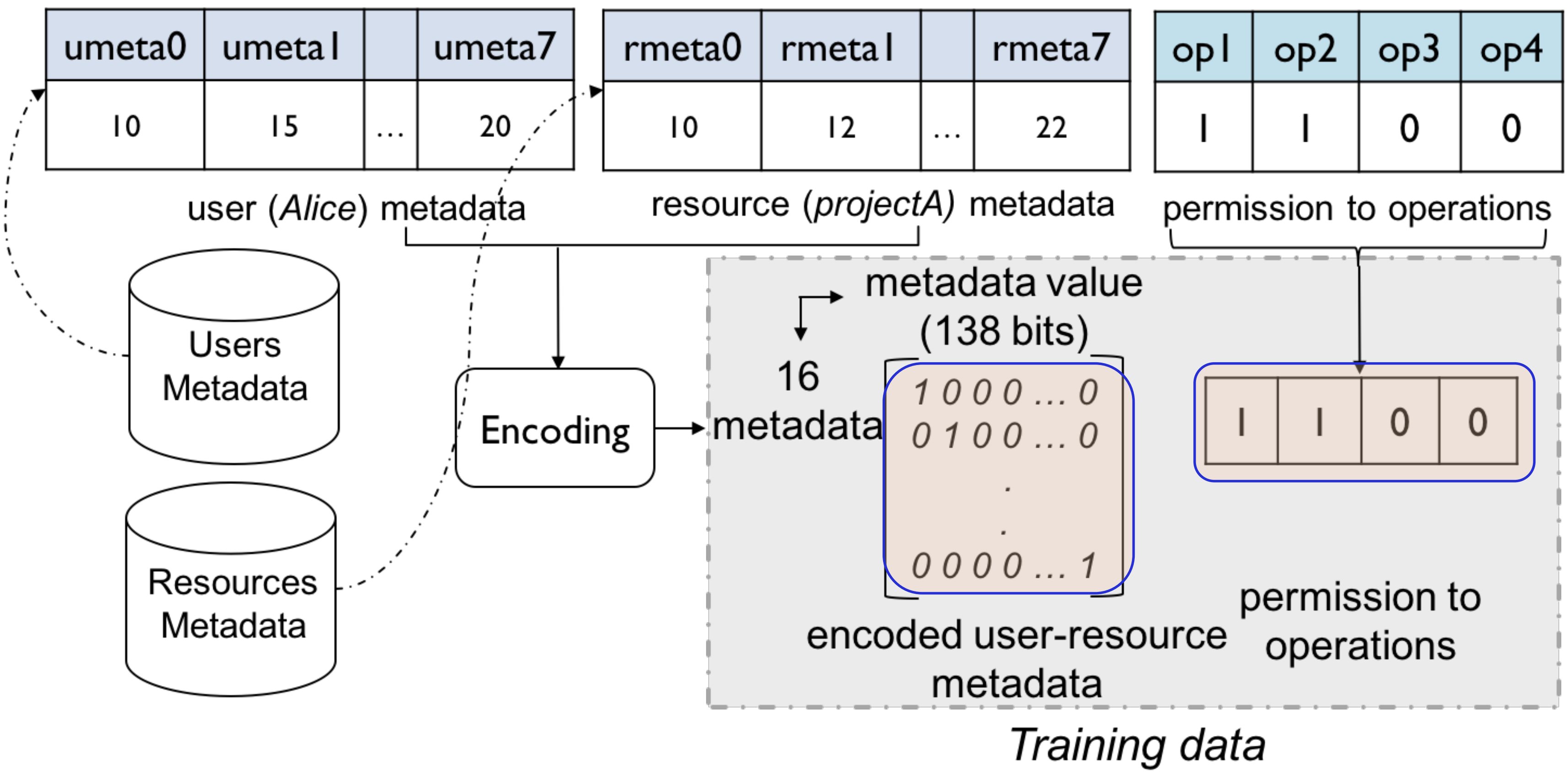
List of Datasets

#	Dataset	Type	Users	User Metadata	Resources	Resource Metadata	Authorization Tuples
1	*amazon-kaggle*	Real-world	9560	8	7517	0	32769
2	*amazon-uci*	Real-world	4224	11	7	0	4224
3	*u4k-r4k-auth11k*	Synthetic	4500	8	4500	8	10964
4	*u5k-r5k-auth12k*	Synthetic	5250	8	5250	8	12690
5	*u5k-r5k-auth19k*	Synthetic	5250	10	5250	10	19535
6	*u4k-r4k-auth21k*	Synthetic	4500	11	4500	11	20979
7	*u4k-r7k-auth20k*	Synthetic	4500	11	7194	11	20033
8	*u4k-r4k-auth22k*	Synthetic	4500	13	4500	13	22583
9	*u4k-r6k-auth28k*	Synthetic	4500	13	6738	13	28751
10	*u6k-r6k-auth32k*	Synthetic	6000	10	6000	10	32557

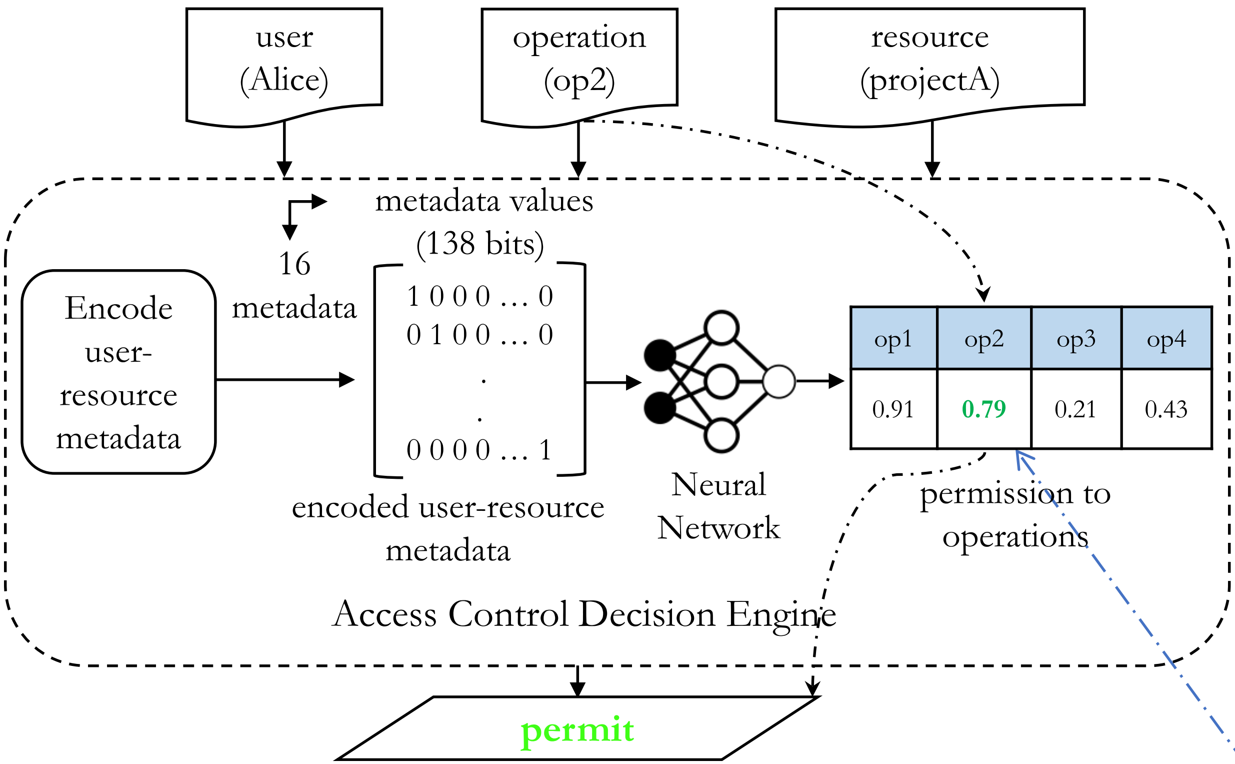


Preparing Training Data for DLBACα

The data type in our datasets are **nominal-categorical**



Decision Making Process in DLBACα



Permit decision is made comparing the output **probability** with a threshold

Evaluation Methodology

Multiple instances of DLBACα

- ResNet ($DLBAC_{\alpha\text{-}R}$)
- DenseNet ($DLBAC_{\alpha\text{-}D}$)
- Xception ($DLBAC_{\alpha\text{-}X}$)

Classical ML Algorithms

- SVM
- Random Forest (RF)
- Multilayer Perceptron (MLP)

State-of-the-art policy mining techniques

- XuStoller [1]
- Rhapsody [2]
- EPDE-ML [3]

[1] Xu et al. 2014. "Mining attribute-based access control policies." IEEE TDSC
[2] Cotrini et al. 2018. Mining ABAC rules from sparse logs. In IEEE Euro S&P.
[3] Liu et al. 2021. Efficient Access Control Permission Decision Engine Based on Machine Learning. Security & Communication Networks.

Evaluation Metrics

80% samples for the training, and 20% testing

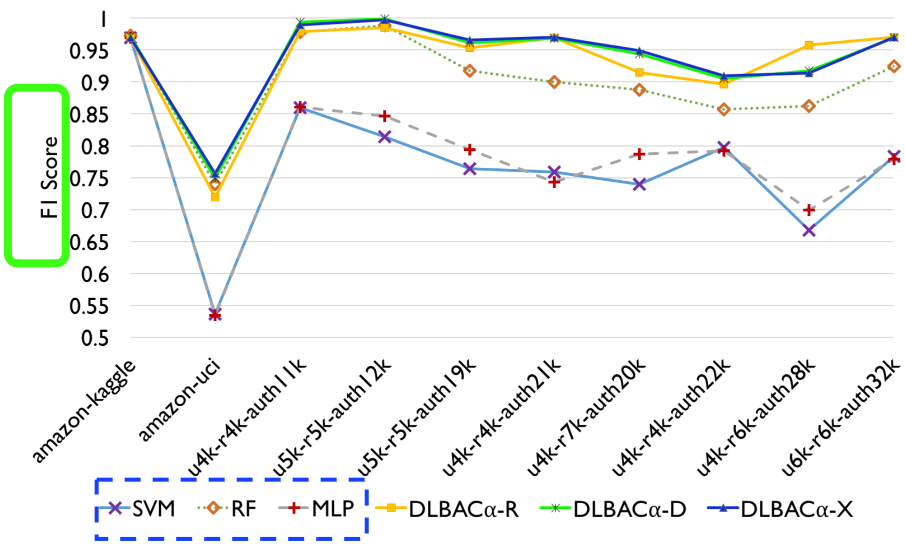
F1 TPR ↑	FPR ↓

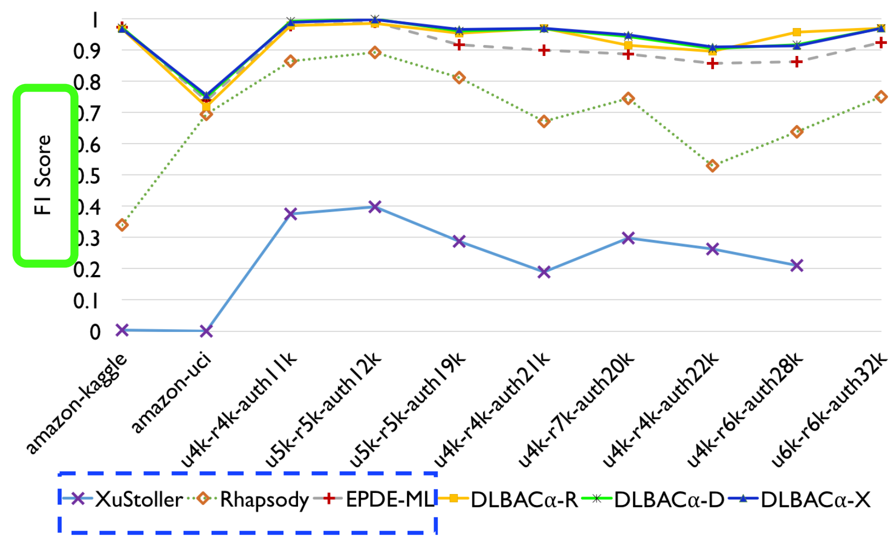
A higher F1 score: better generalization

A higher TPR: accurate and efficient in granting access

A lower FPR: efficient in denying access

Comparison with ML Algorithms and State-of-the-art Policy Mining

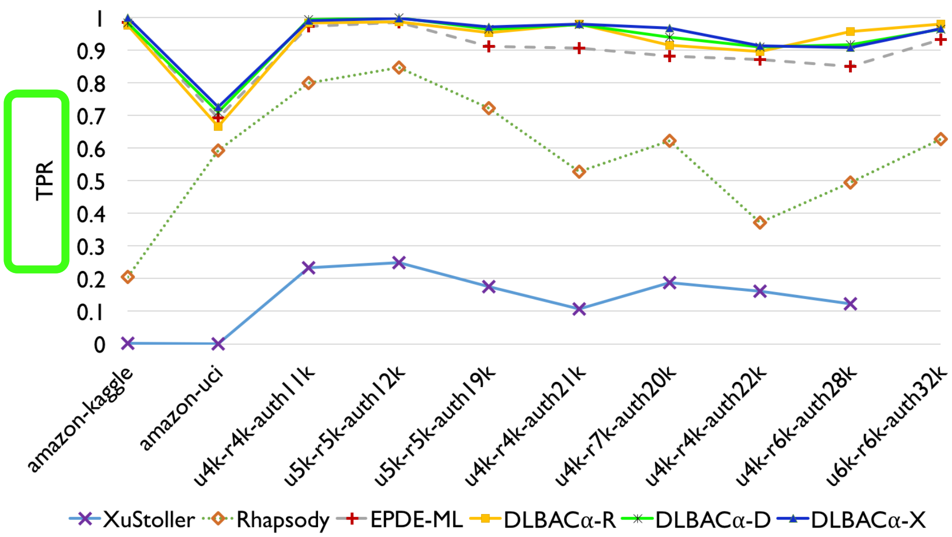




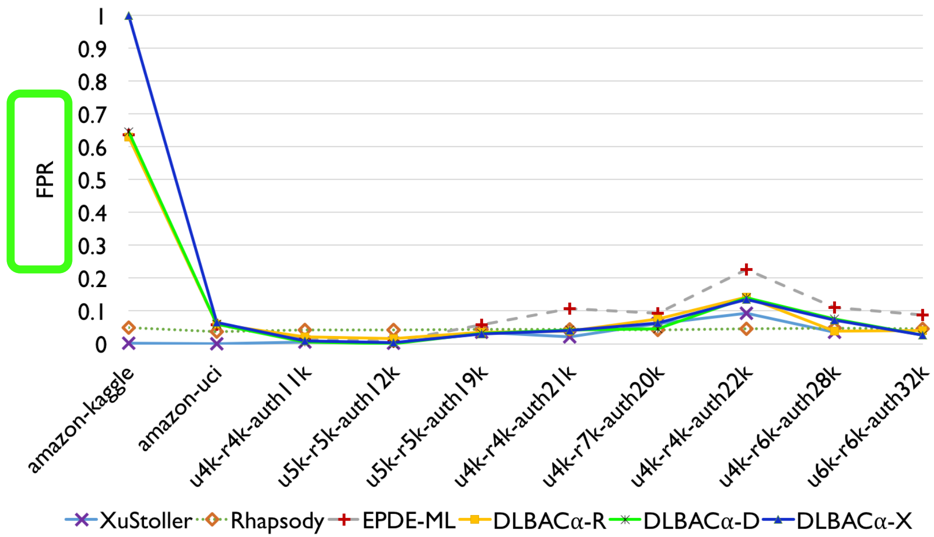
make accurate access decisions and generalize better

Comparison with Policy Mining Algorithms

handling desirable access

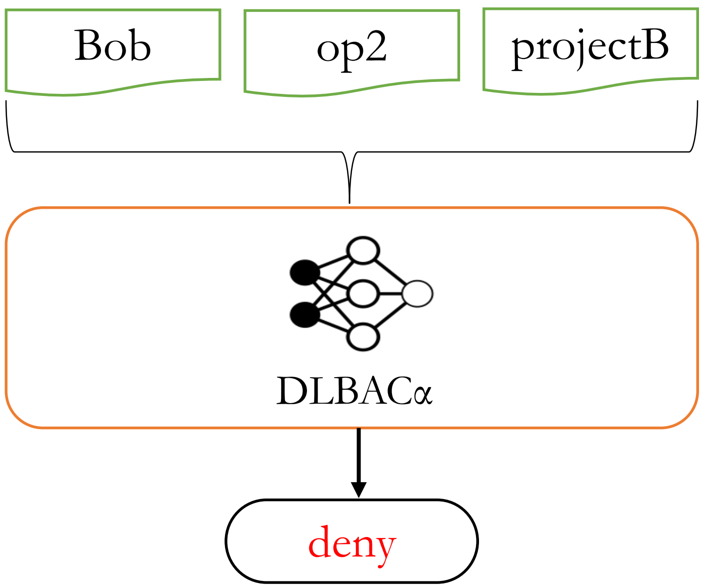


handling unwanted access



Efficient in permitting desired accesses and denying unwanted accesses

Understanding DLBAC Decisions



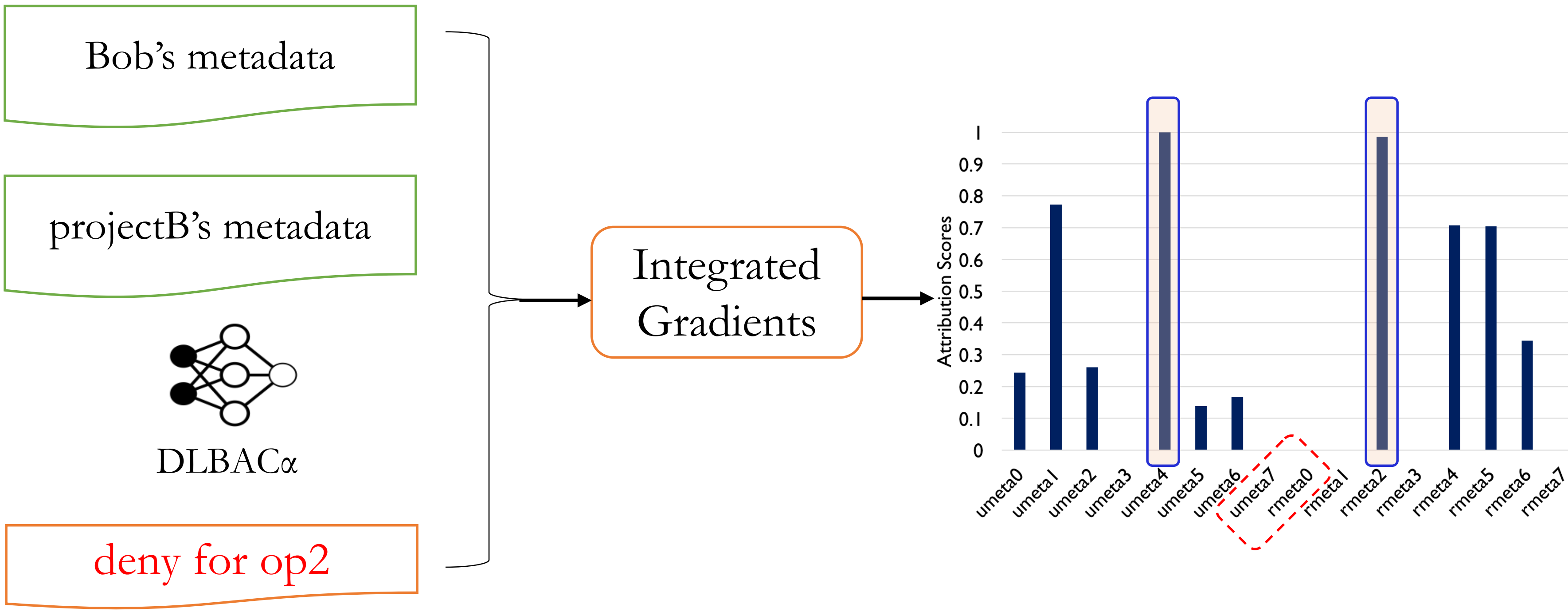
A sample access request

Why does Bob's 'op2' access been denied for projectB resource?

Which metadata are important/ influential for this decision?

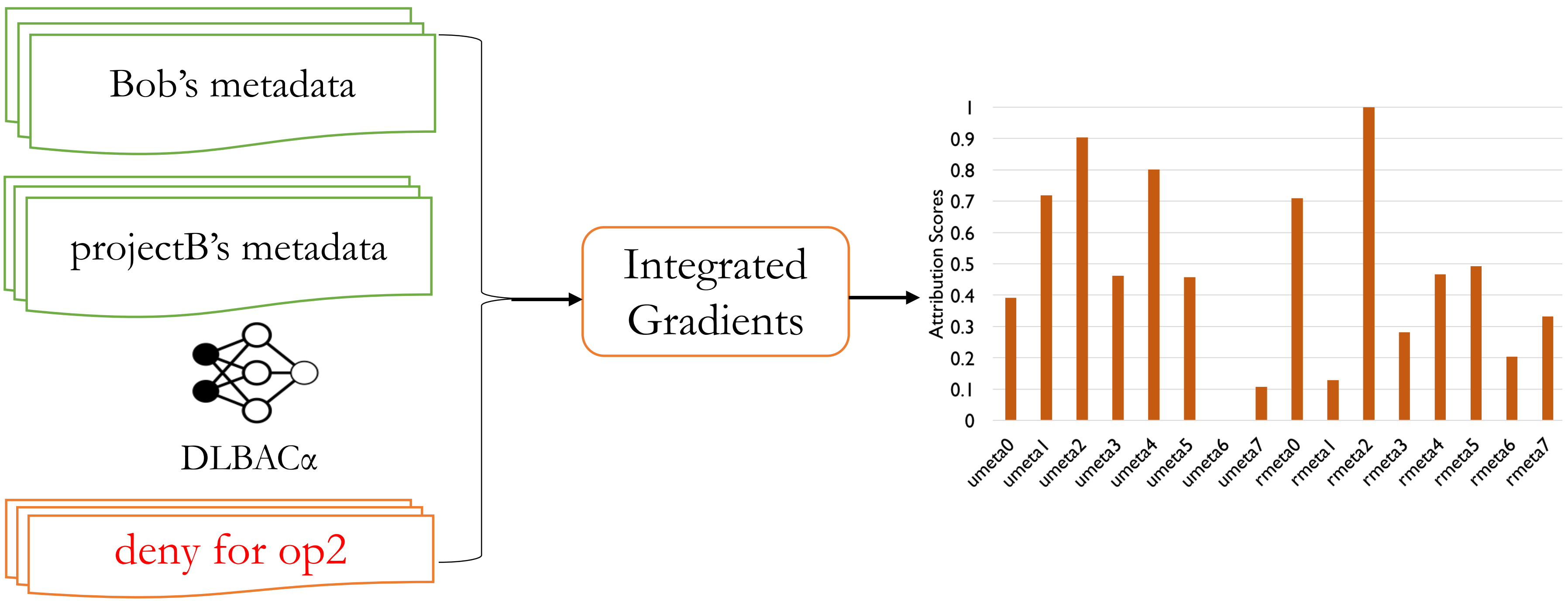
- Propose two approaches
  - Integrated Gradients
  - Knowledge Transferring

Integrated Gradients



Local Interpretation

Integrated Gradients



Global Interpretation

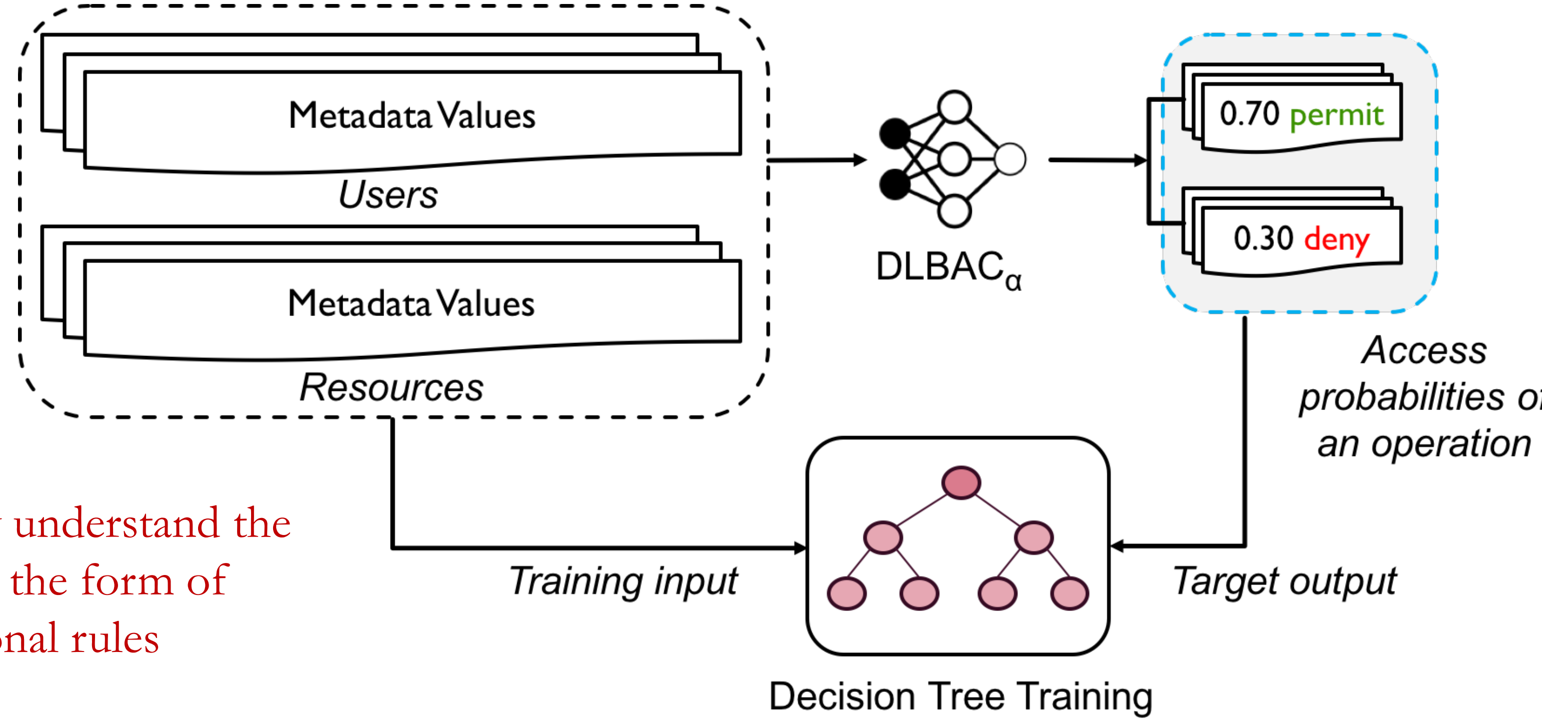
Application of Integrated Gradient-based Understanding

	umeta0	umeta1	umeta2	umeta3	umeta4	umeta5	umeta6	umeta7	rmeta0	rmeta1	rmeta2	rmeta3	rmeta4	rmeta5	rmeta6	rmeta7	access to op1 operation
tuple1	30	51	126	26	37	129	89	5	5	123	95	40	37	129	89	14	deny
	Carol's metadata								projectC's metadata								
tuple2	61	84	5	29	44	105	6	30	5	123	95	40	37	129	89	14	permit
	Dave's metadata								projectC's metadata								
modified tuple1	30	84	126	26	44	129	89	5	5	123	95	40	37	129	89	14	$\boxed{\text{permit}}$
	Carol's metadata								projectC's modified metadata								

- Strengthen the effect of "influential metadata"
- Can be utilized in future access modification

Is there any relations among metadata?

Knowledge Transferring



approximately understand the decision in the form of traditional rules

- Rule: local interpretation
- DT: global interpretation

Section-2 Summary

- DLBAC is an effective operational model for access control
- Black-box decisions are understandable in human terms

- **Issues:**
  - How to change/ update access control state?

Section-3

Machine Learning Based Access Control (MLBAC)

Comprehensive Literature Review : ML in Access Control

Operational Model of MLBAC

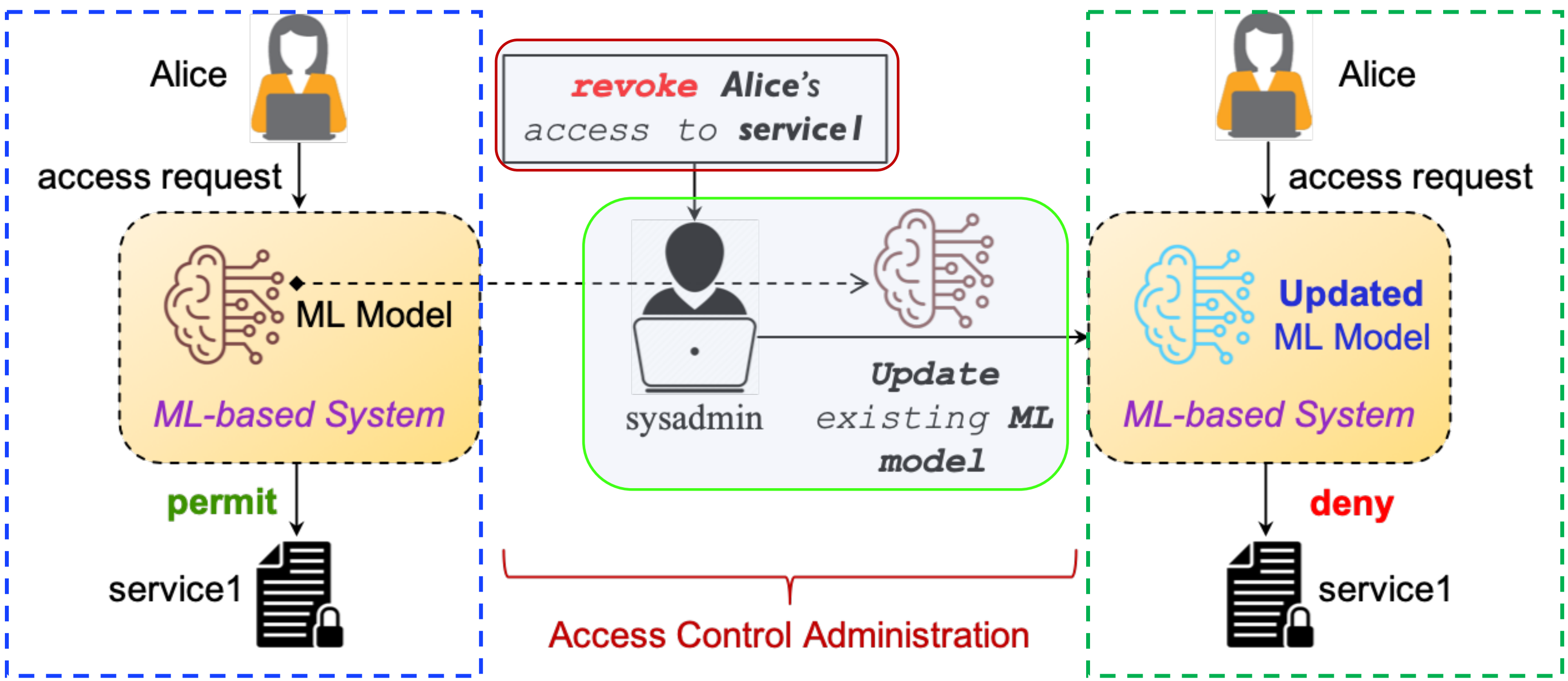
Administration of MLBAC

DLBAC (prototype, interpretation)

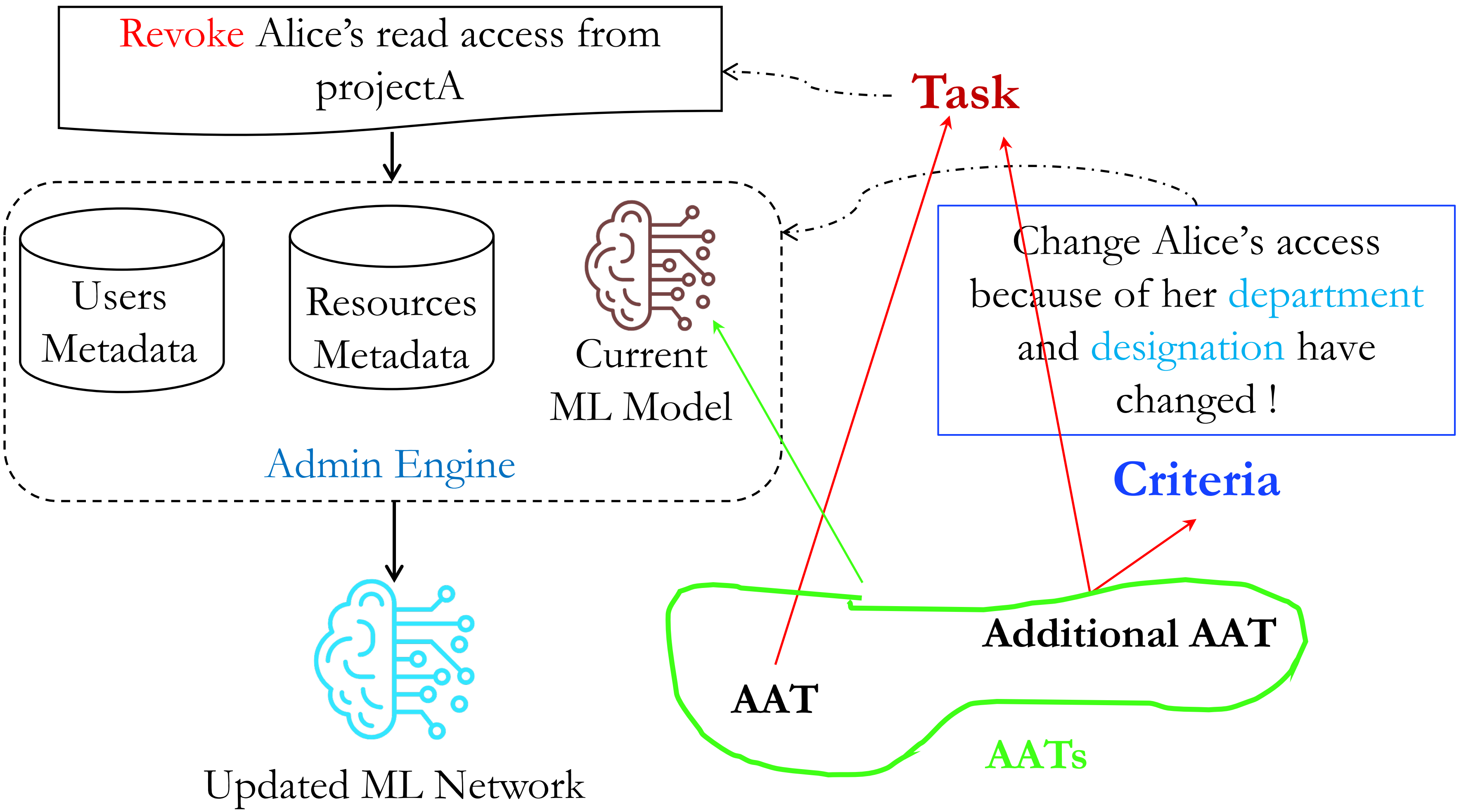
Adversarial Attacks in DLBAC

Implementation and Evaluation of DLBAC

Administration in Machine Learning Based Access Control

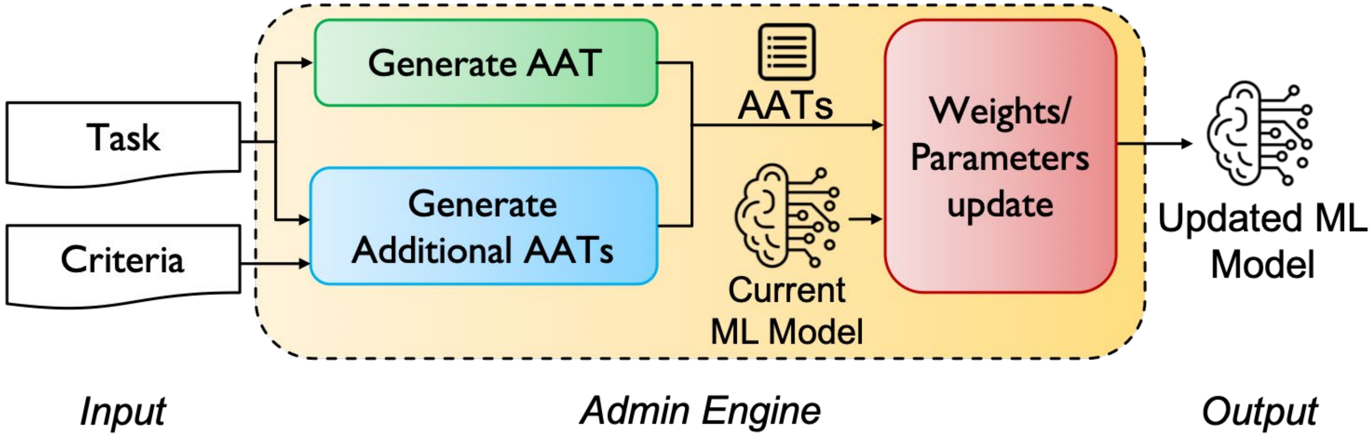


MLBAC Administration Overview

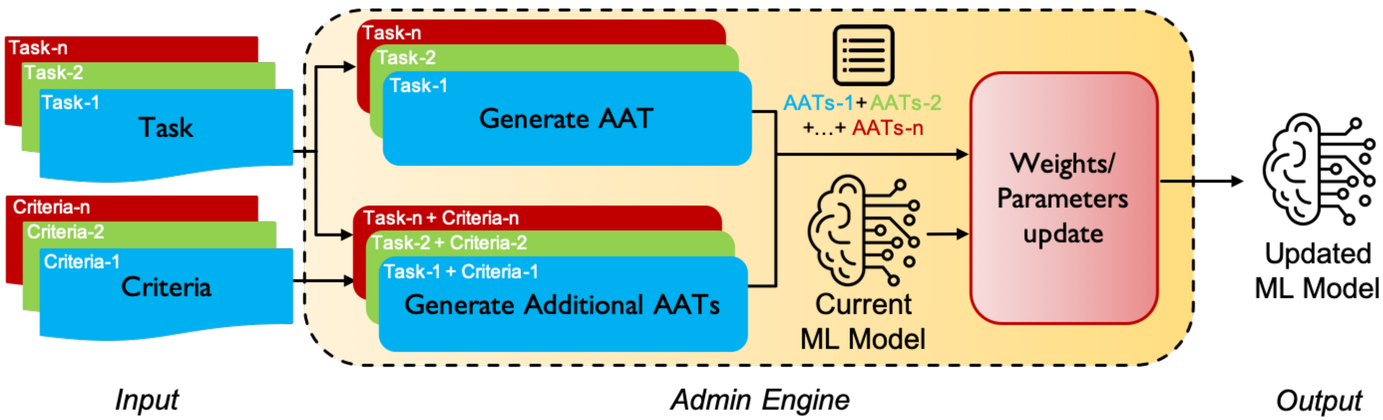


Administration Process Flow

Single Task

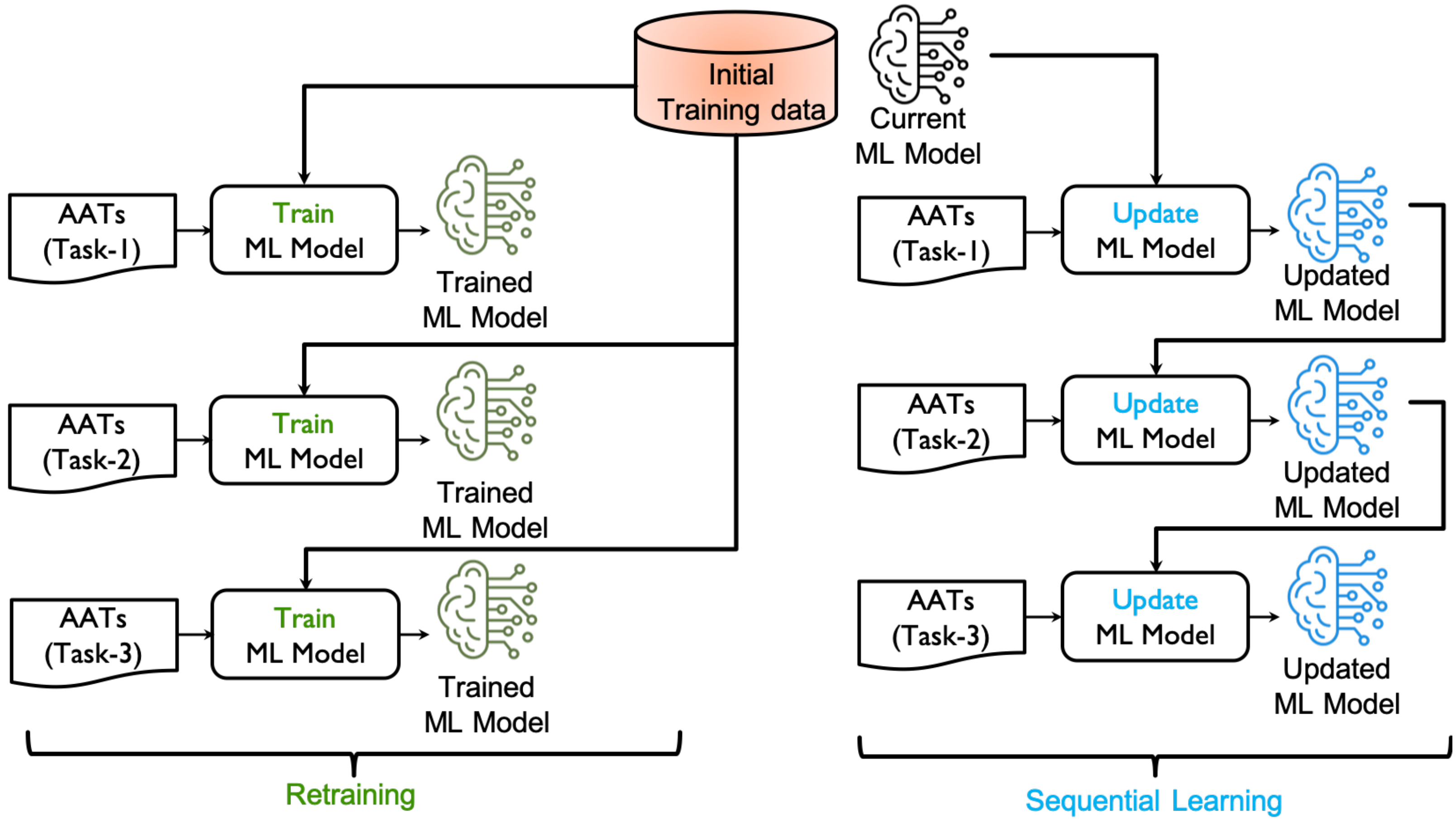


Multi Task



Simulate 2-Tasks, 3-Tasks, and 6-Tasks

Weights/Parameters Update

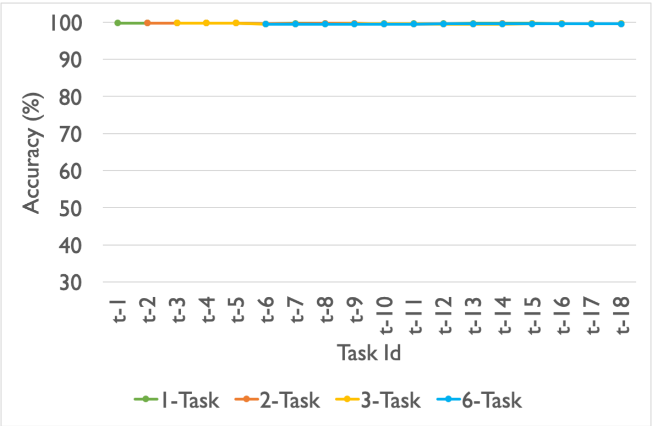


18 random Tasks with different Criteria

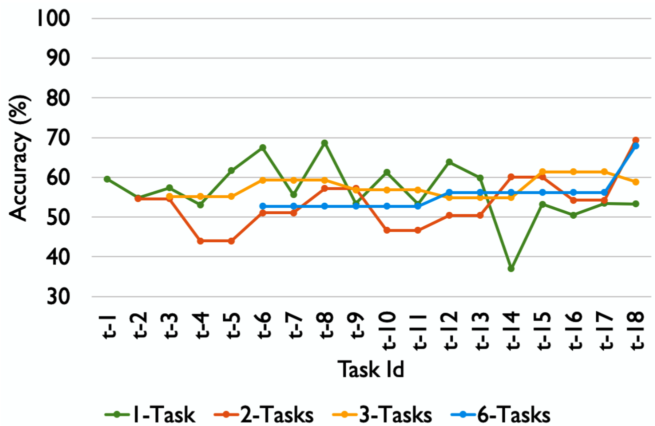
Performance Evaluation

- RF-MLBAC: Add additional estimators
- ResNet-MLBAC: Fine-tuning

- How accurately it can learn new changes (AATs)
- How well it can preserve existing access states for all other users/resources (**OATs**)



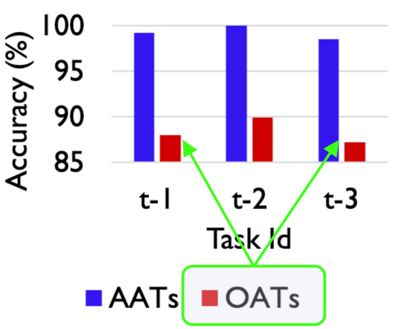
OATs



AATs

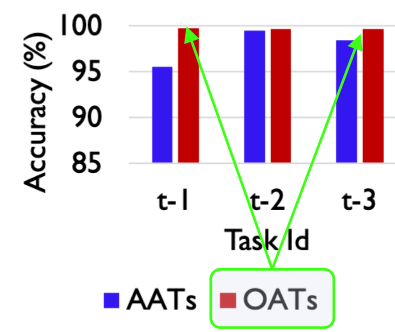
Unable to accommodate new changes with good accuracy !

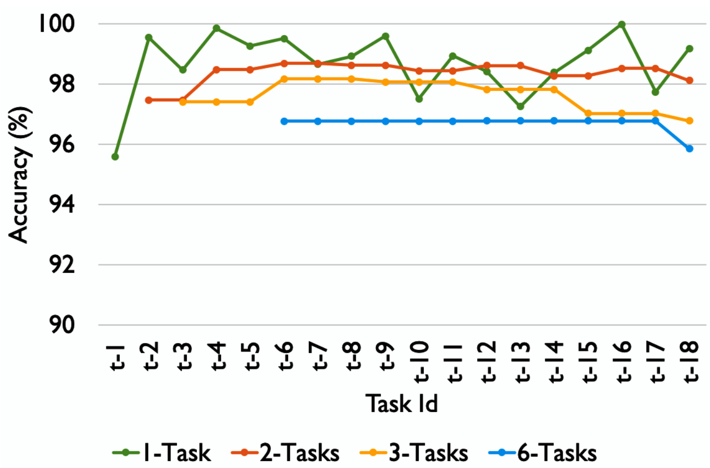
Performance Evaluation (cont'd) (ResNet-MLBAC)



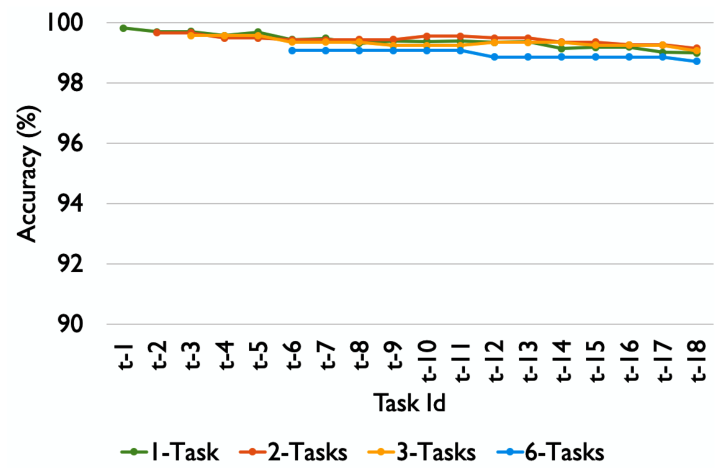
Starts to forget other Access Control state- Catastrophic forgetting

Replay Data





AATs



OATs

Multi-task administration generally provides better performance

Section-3 Summary

- Sequential learning is an effective method
- Deep neural network systems performed better

- Issues:
  - Some dependencies on physical data storage (Replay Data)
  - Designing better “Criteria” is challenging

Section-4 (Part-A)

Machine Learning Based Access Control (MLBAC)

Comprehensive Literature Review : ML in Access Control

Operational Model of MLBAC

Administration of MLBAC

DLBAC (prototype, interpretation)

Adversarial Attacks in DLBAC

Implementation and Evaluation of DLBAC

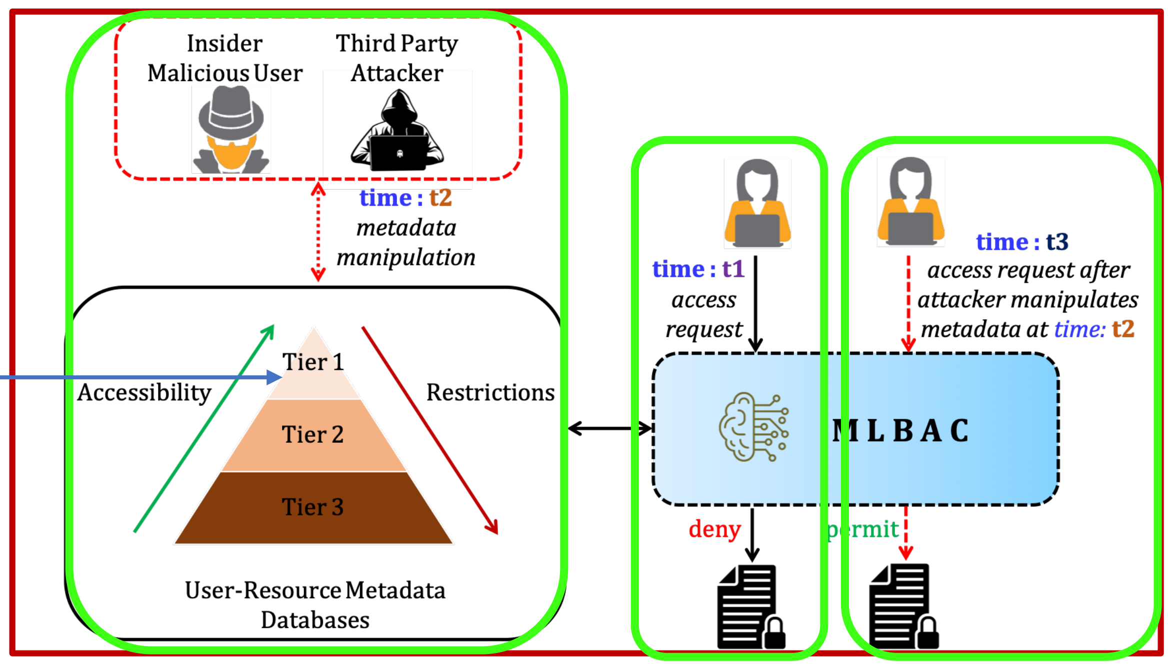
Adversarial Attack in MLBAC

STOP perturbation YIELD

Adversarial Example

Imperceptibility

Modify part of the input to any degree



Adversarial Attack Problem

Actual decision

Perturbation

$$f(x) = y \neq f(x + x_p) = t$$

Target decision

Perturbation weight

↓ perturbation

$$g(x_p) = \mathcal{L}(x + x_p,\, t) + \omega \,\|\, x_p \,\|$$

Accessibility Constraint

Access Restriction

$$g(x_p) = \mathcal{L}(x + x_p,\, t) + \omega \,\|\, x_p \circ c \,\|$$

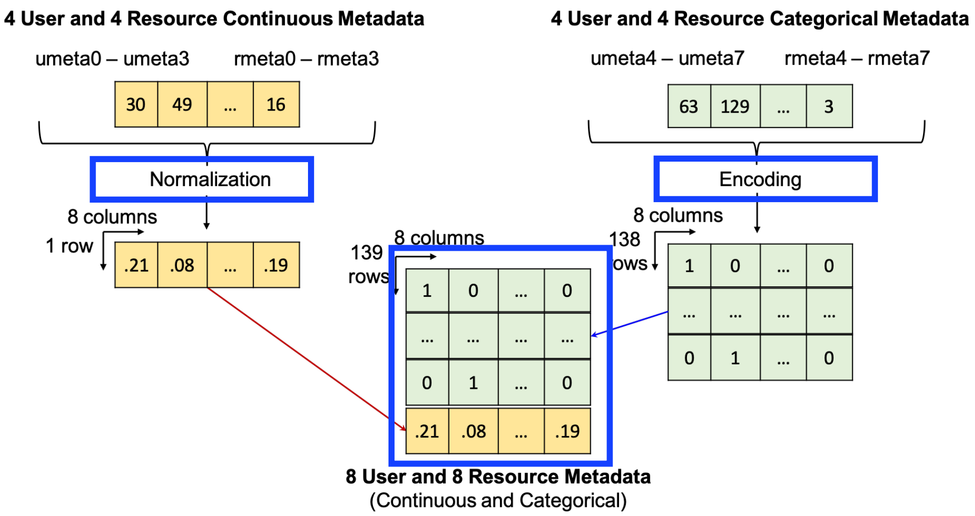
Methodology

Continuous and Categorical

'age,' 'salary', 'security_level,' 'designation'

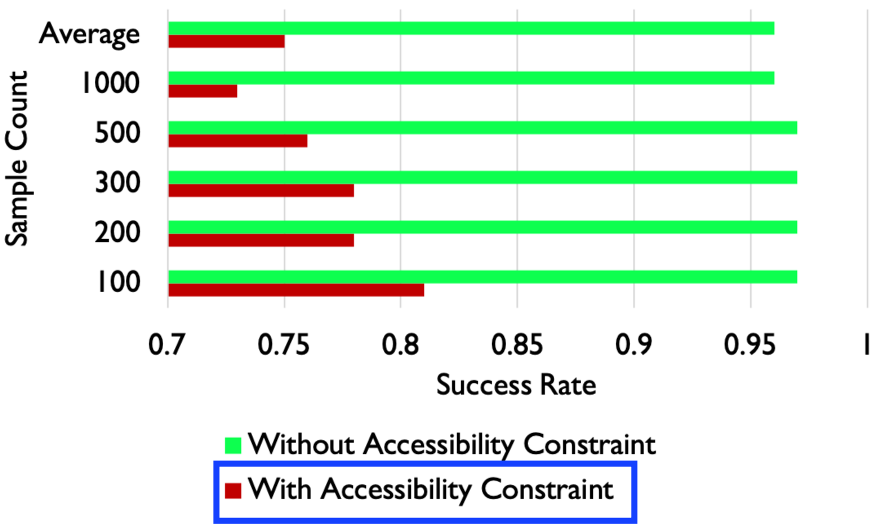
- Accessibility Constraint
  - Pearson's Correlation
  - Value between 0 and 1
  - Higher correlation, more restricted

- Two DLBAC datasets
  - System-1 and System-2

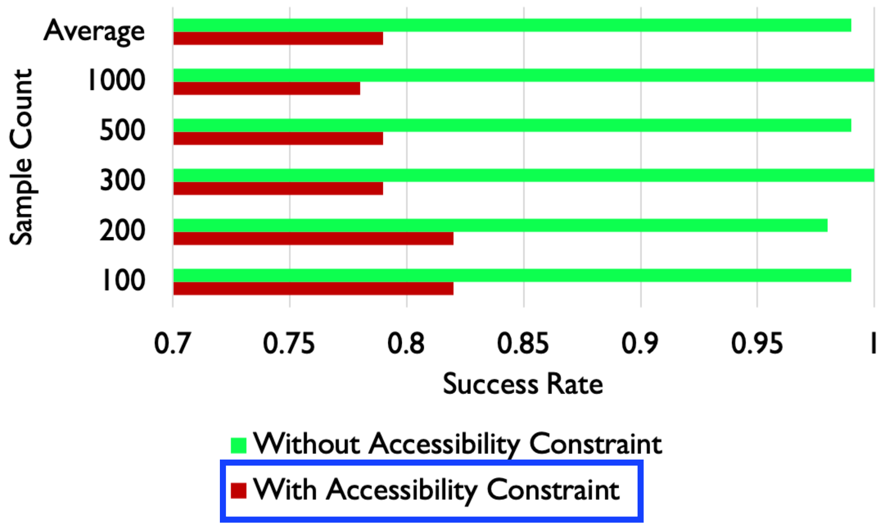


Evaluation

$$\textbf{Success Rate} = \frac{\text{Successfully crafted adversarial examples}}{\text{Samples attempted for the adversarial example creation}}$$



System-1



System-2

Section-4 (Part-A) Summary

- Accessibility constraint minimized the attacks

- Issues
  - Need better defense if no accessibility constraint

Section-4 (Part-B)

Machine Learning Based Access Control (MLBAC)

Comprehensive Literature Review : ML in Access Control

Operational Model of MLBAC

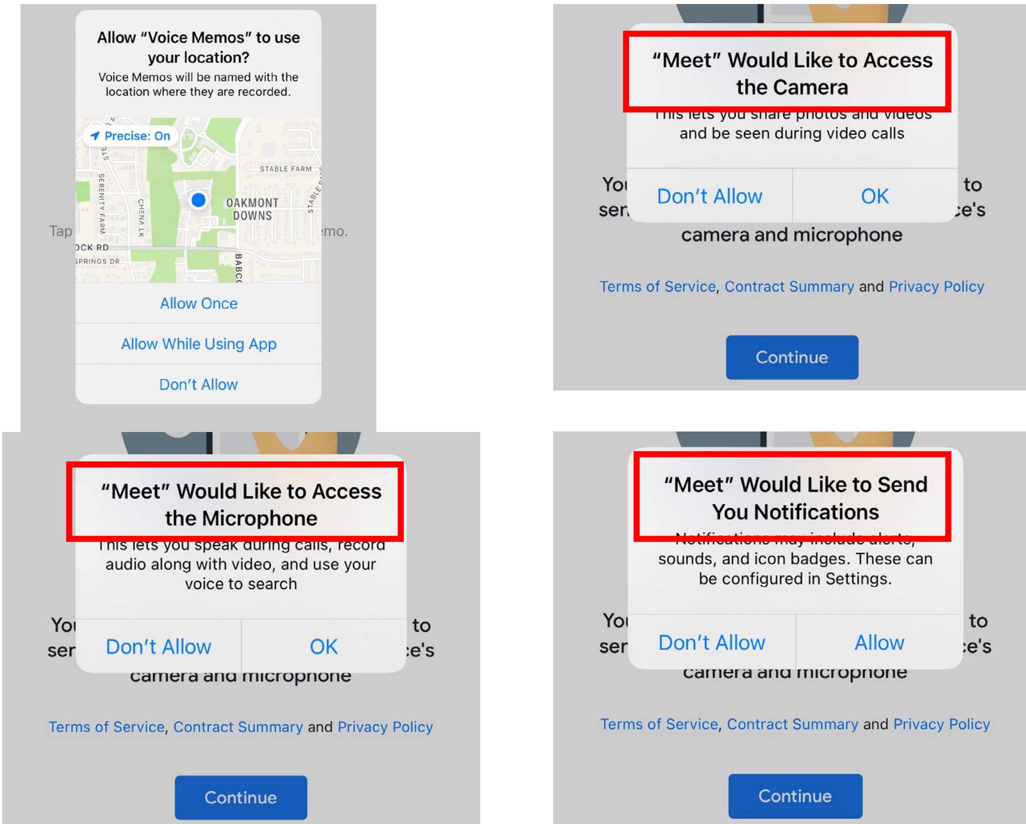
Administration of MLBAC

DLBAC (prototype, interpretation)

Adversarial Attacks in DLBAC

Implementation and Evaluation of DLBAC

DLBAC Assisted Permission Recommendation for Mobile Devices



Ask-On-Install (AOI)

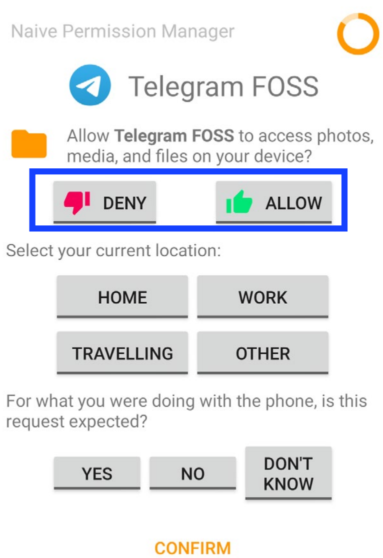
Ask-On-First-Use (AOFU)

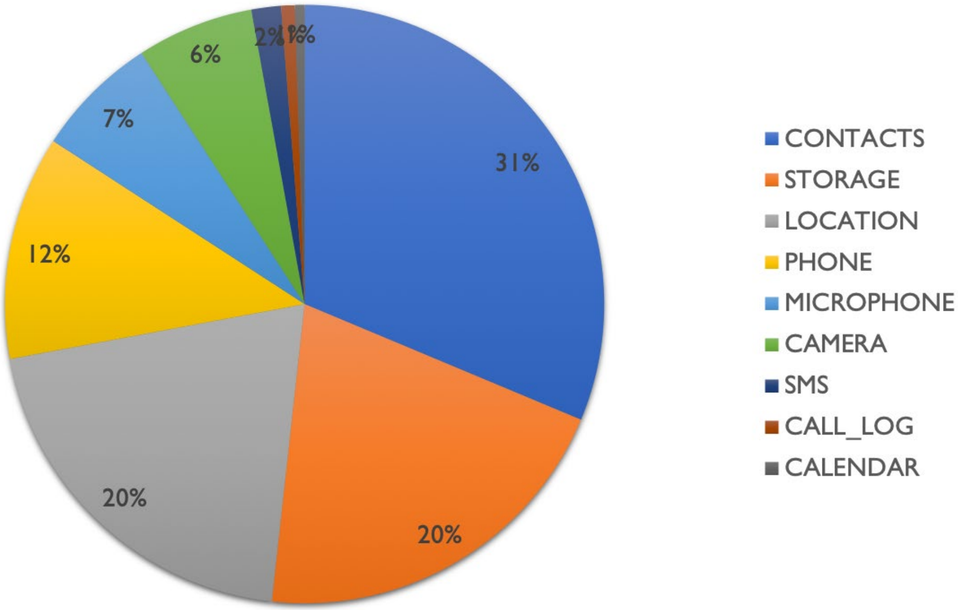
Could DLBAC automate this permission decision?

… abundant permission requests

COP-MODE Dataset

- Developed by Mendes et al. [4], **65K** permission requests
- At each permission request:
  - **Requesting application:** name and play store category
  - **Permission:** name (CONTACTS, STORAGE, etc.) and grant result (allow/deny)
  - **Phone state:** geolocation, plug, call state, network connection , etc.
  - **User context:** time, semantic location, in event or not, etc.





[4] . Mendes, R., Brandão, A., Vilela, J. P., and Beresford, A. R.. Effect of User Expectancy on Mobile App Privacy: A Field Study. In 2022 IEEE PerCom.

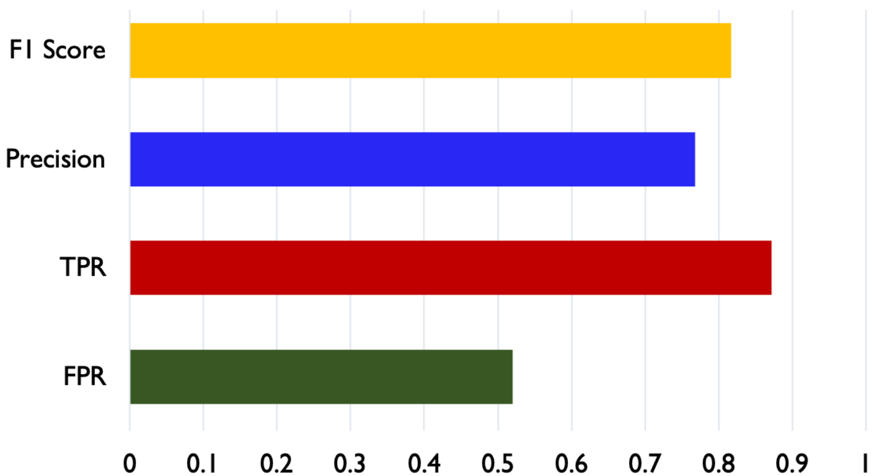
Evaluation

- Three DLBAC instances with: ResNet, DenseNet, and Xception
- State-of-the-art (**Brandão et al.** [5]) Accuracy **88%** and F1 Score **0.90**

Cluster like-minded users, Liu et al. [6]

Accuracy: **74.02%**

Accuracy: ~**88.5 %** F1 Score: ~0.**915**



DLBAC Performance (ResNet)

DLBAC Instances Performance

[5]. Brandão, A. et al. Prediction of Mobile App Privacy Preferences with User Profiles via Federated Learning. In 2022 ACM CODASPY.
[6]. Liu et al. Follow My Recommendations: A Personalized Privacy Assistant for Mobile App Permissions. In SOUPS 2016.

Section-4 (Part-B) Summary

- Clustering like-minded users has an advantage
- Issues
  - Recommendation accuracy needs to be improved

Future Research Directions

DLBAC Issues	• Understanding, Administration, etc. • Accuracy is lower in some cases
MLBAC Verification	• Measuring Correctness • Testing Framework
Bias and Fairness	• Data could comes from untrusted sources • Imbalance data may bias the decision
Adversarial Issues	• Adversarial attack for Classical ML based systems • Need more strong defense mechanism
DLBAC in Tandem	• Reinforcing access decision • Monitoring and feedback

Dissertation Publications

- **Nobi, Mohammad Nur**, Ram Krishnan, Yufei Huang, Mehrnoosh Shakarami, and Ravi Sandhu. "Toward Deep Learning Based Access Control." In ACM CODASPY. 2022.
- **Under Review**
  - (ESORICS 2022) **Mohammad Nur Nobi**, Ram Krishnan, Yufei Huang, and Ravi Sandhu. “Administration of Machine Learning Based Access Control”.
  - (itaDATA 2022) **Mohammad Nur Nobi**, Ram Krishnan, and Ravi Sandhu. “Adversarial Attacks in Machine Learning Based Access Control”.
  - (ACM Computing Survey, arXiv) **Mohammad Nur Nobi**, Maanak Gupta, Lopamudra Praharaj, Mahmoud Abdelsalam, Ram Krishnan, and Ravi Sandhu. “Machine Learning in Access Control: A Taxonomy and Survey”.

Source code and datasets URL:

https://github.com/dlbac/DlbacAlpha
https://github.com/mlxac/MLBAC-Admin
https://github.com/mlxac/MLBAC-AdversarialAttack

Thank You

Questions and Comments

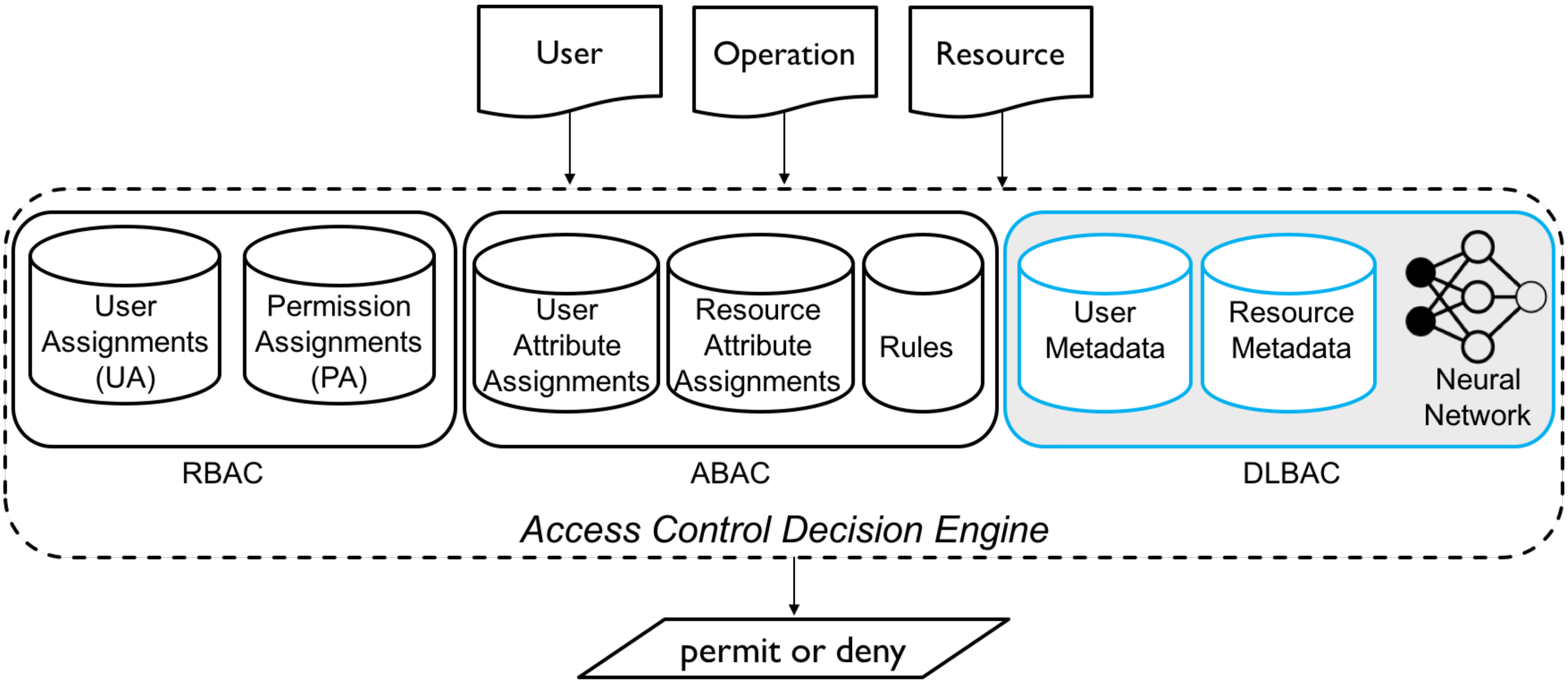
The Institute for Cyber Security

Center for Security and Privacy Enhanced Cloud Computing

UTSA Computer Science

Backup

Decision Making in Classical Approaches vs. DLBAC



DLBAC works with any deep neural network

Dataset Generation

Generate a synthetic dataset using Xu et al. [1]

<Alice, projectA, {op1, op3}>

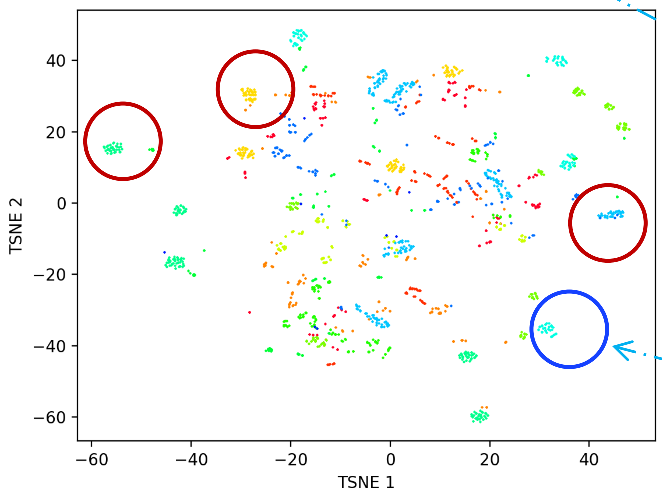
Each dot represents an authorization tuple

Each color indicates a unique combination of access operations

The position of a tuple is based on both user and resource metadata values

Two tuples are closed to each other, which indicates they have similar user-resource metadata values

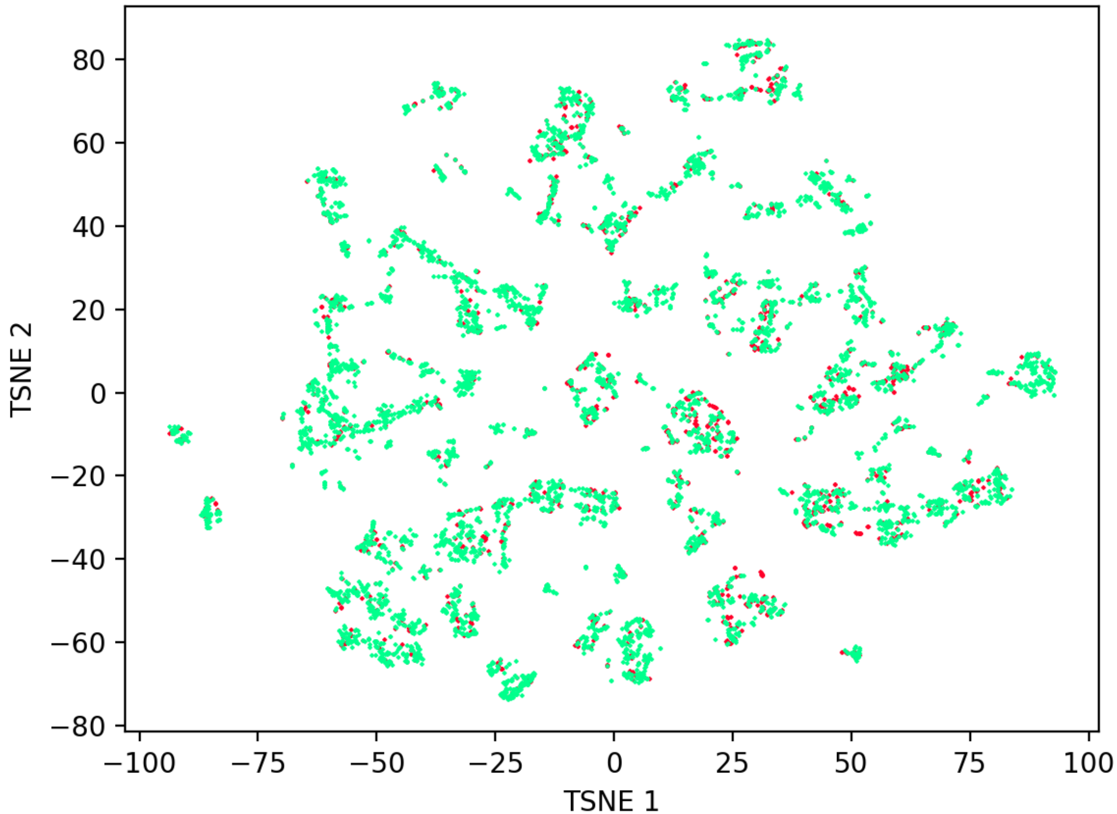
multiple tuples of the same color indicate they have the same access



t-SNE plot of a synthetic dataset

1. Xu et al. 2014. "Mining attribute-based access control policies." IEEE TDSC.

Data Characteristics in Real-world Systems



A dataset representing Amazon* access control system

* https://www.kaggle.com/c/amazon-employee-access-challenge/

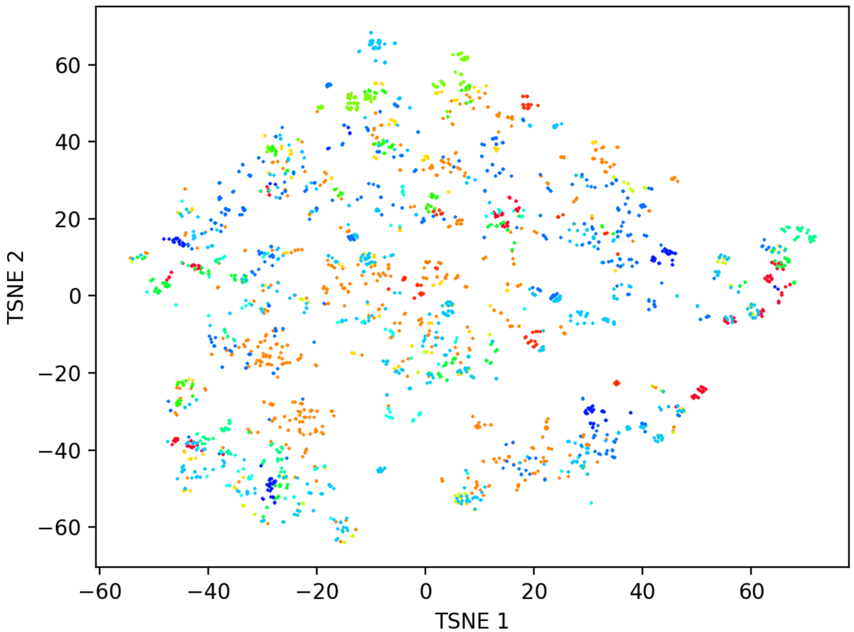
Network Architectures

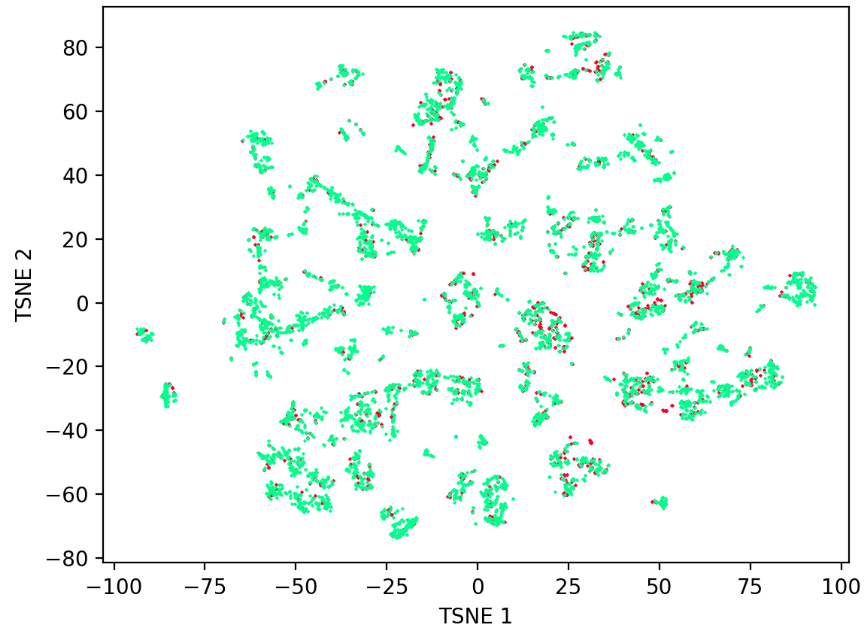
For dataset 1-4: ResNet8
For dataset 5-10: ResNet50

Layers	Output Size	DenseNet-121
Convolution	112×112	
Pooling	56×56	
Dense Block (1)	56×56	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$
Transition Layer (1)	56×56	
	28×28	
Dense Block (2)	28×28	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$
Transition Layer (2)	28×28	
	14×14	
Dense Block (3)	14×14	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 24$
Transition Layer (3)	14×14	
	7×7	
Dense Block (4)	7×7	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 16$
Classification Layer	1×1	

ResNet, DenseNet

Dataset Generation

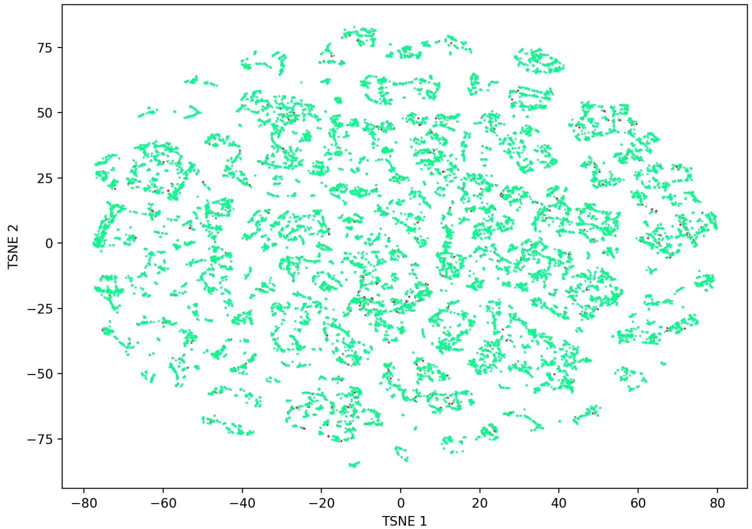




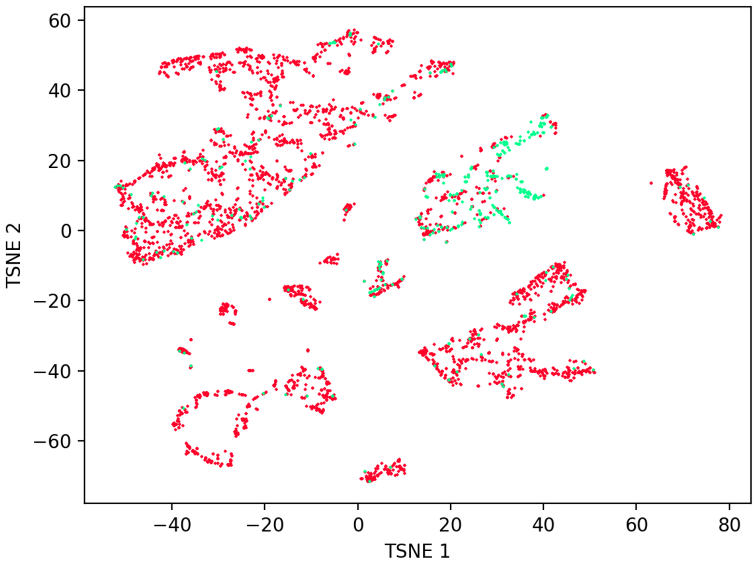
A dataset with 800 users and 665 resources, 3 hidden metadata, **fixed set of metadata values**.

A real-world dataset from Amazon

Characteristics of AmazonKaggle and AmazonUCI Datasets



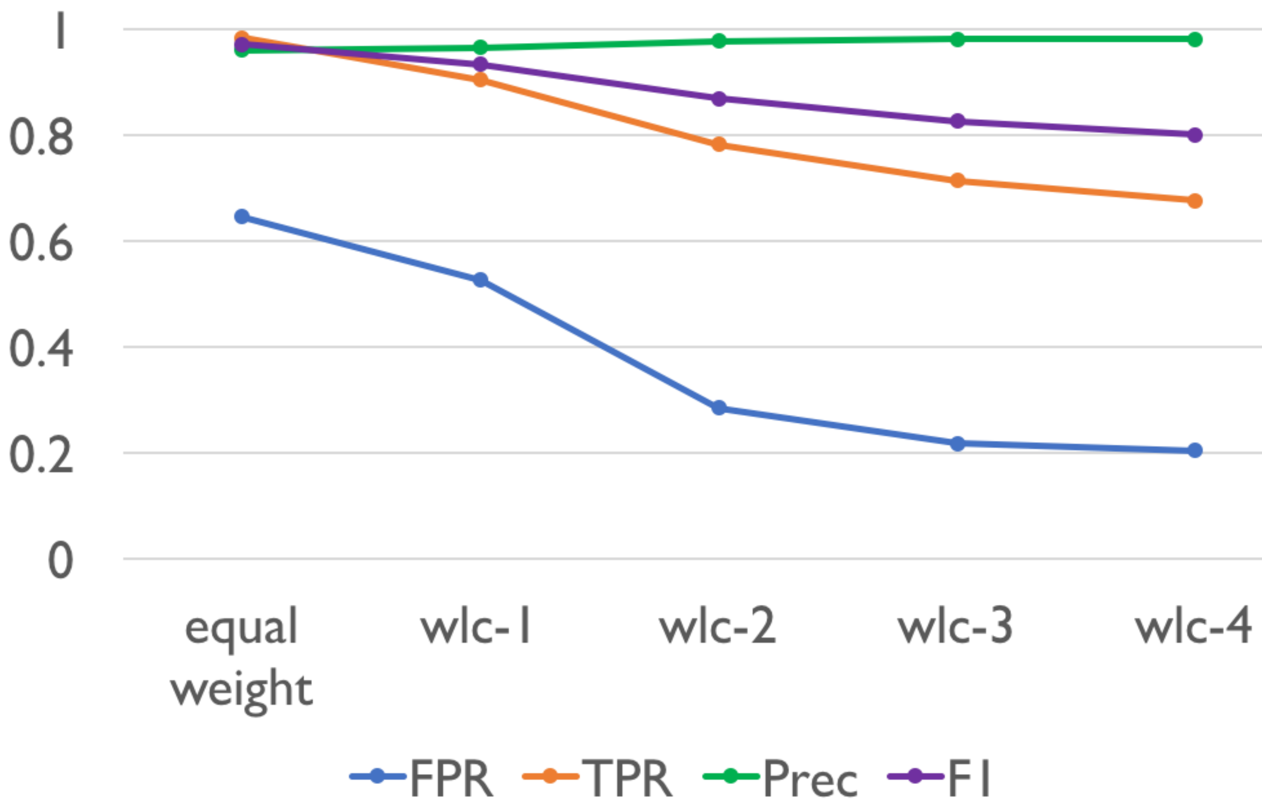
Amazon Kaggle Dataset



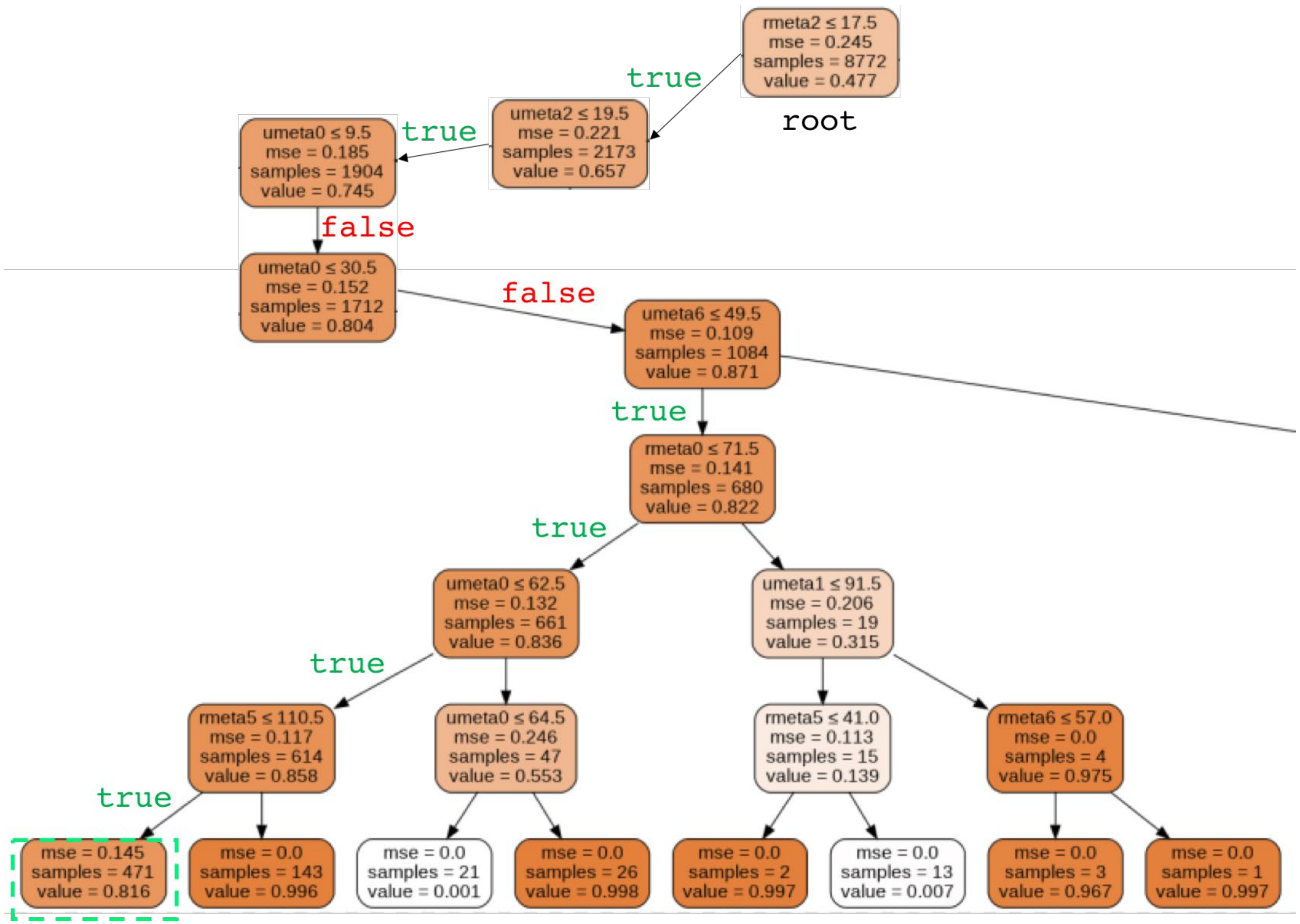
Amazon UCI Dataset

Highly imbalanced !

FPR Performance Improvement in DLBACα



Decision Tree Generated from KT in DLBACα



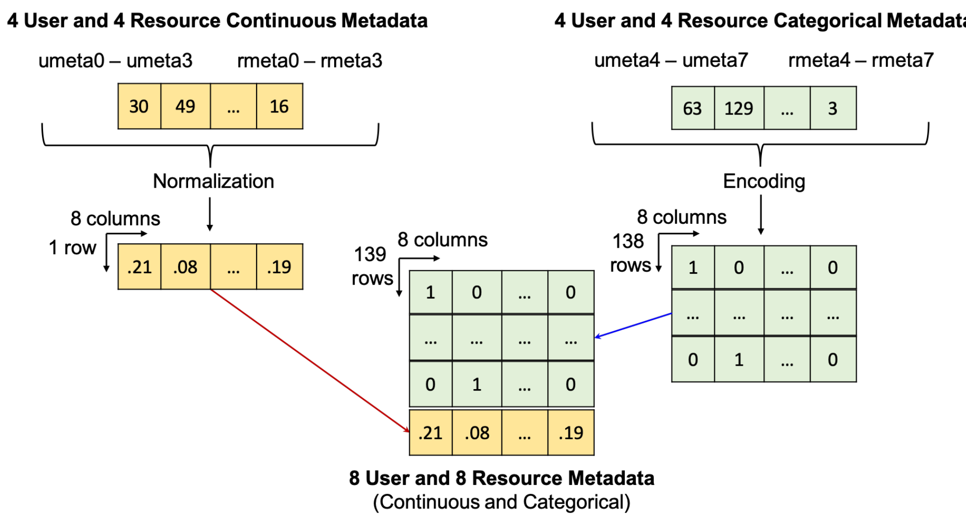
List of Tasks and Criteria for MLBAC Administration

Task Id	Task	Criteria	Size of AATs
t-1	$\langle uid = 259, rid = 112, op3, permit\rangle$	$\langle umeta0 \in \{9\}, umeta6 \in \{6\}, rmeta0 \in \{9\}, rmeta3 \in \{46\}\rangle$	43
t-2	$\langle uid = 4624, rid = 4634, op4, deny\rangle$	$\langle umeta2 \in \{58, 49\}, umeta3 \in \{39\}, rmeta3 \in \{39\}\rangle$	94
t-3	$\langle uid = 1992, rid = 1858, op1, permit\rangle$	$\langle umeta2 \in \{11\}, rmeta2 \in \{11\}, rmeta3 \in \{48, 91\}\rangle$	92
t-4	$\langle uid = 5049, rid = 5177, op4, permit\rangle$	$\langle umeta1 \in \{6\}, umeta4 \in \{47, 71\}, rmeta1 \in \{6\}\rangle$	215
t-5	$\langle uid = 2034, rid = 2041, op2, deny\rangle$	$\langle umeta4 \in \{10\}, rmeta1 \in \{6, 10\}, rmeta4 \in \{10\}\rangle$	75
t-6	$\langle uid = 1348, rid = 1083, op2, permit\rangle$	$\langle umeta3 \in \{46, 50, 53\}, umeta5 \in \{13\}, rmeta3 \in \{46, 50, 53\}, rmeta5 \in \{13\}\rangle$	187
t-7	$\langle uid = 1345, rid = 1092, op4, permit\rangle$	$\langle umeta0 \in \{24, 64\}, umeta6 \in \{7\}, rmeta0 \in \{24, 64\}, rmeta6 \in \{7\}\rangle$	139
t-8	$\langle uid = 442, rid = 580, op3, permit\rangle$	$\langle umeta3 \in \{49\}, umeta5 \in \{47, 111\}, rmeta5 \in \{47, 111\}, rmeta7 \in \{49\}\rangle$	134
t-9	$\langle uid = 2599, rid = 2593, op1, permit\rangle$	$\langle umeta0 \in \{11\}, umeta1 \in \{17\}, rmeta0 \in \{11\}, rmeta1 \in \{17\}\rangle$	66
t-10	$\langle uid = 4112, rid = 1241, op2, permit\rangle$	$\langle umeta1 \in \{18\}, rmeta1 \in \{18\}, rmeta3 \in \{45, 47, 113\}\rangle$	75
t-11	$\langle uid = 2135, rid = 4875, op3, deny\rangle$	$\langle umeta2 \in \{13\}, umeta4 \in \{71, 96\}, rmeta2 \in \{13\}, rmeta4 \in \{71, 96\}\rangle$	118
t-12	$\langle uid = 660, rid = 560, op1, permit\rangle$	$\langle umeta3 \in \{88\}, umeta5 \in \{48, 111\}, rmeta5 \in \{48, 111\}, rmeta7 \in \{88\}\rangle$	107
t-13	$\langle uid = 2019, rid = 2056, op2, deny\rangle$	$\langle umeta4 \in \{12\}, rmeta1 \in \{78, 82\}, rmeta4 \in \{12\}\rangle$	121
t-14	$\langle uid = 1228, rid = 1088, op1, permit\rangle$	$\langle umeta2 \in \{11, 63\}, umeta5 \in \{20\}, rmeta5 \in \{20\}\rangle$	97
t-15	$\langle uid = 2825, rid = 3044, op2, permit\rangle$	$\langle umeta6 \in \{8\}, rmeta1 \notin \{6, 10\}, rmeta2 \in \{61, 62\}, rmeta6 \in \{8\}\rangle$	107
t-16	$\langle uid = 965, rid = 861, op4, permit\rangle$	$\langle umeta3 \in \{45\}, umeta7 \in \{20\}, rmeta3 \in \{45\}, rmeta6 \in \{20\}\rangle$	63
t-17	$\langle uid = 3745, rid = 3843, op3, permit\rangle$	$\langle umeta0 \in \{31\}, umeta6 \in \{2, 5, 9, 18\}, umeta7 \in \{4, 13\}, rmeta0 \in \{31\}\rangle$	83
t-18	$\langle uid = 2488, rid = 2495, op3, permit\rangle$	$\langle umeta1 \in \{58\}, rmeta1 \in \{58\}, rmeta2 \in \{58, 61\}\rangle$	116

Methodology

Continuous and Categorical data

- Objective function optimization
  - LowProFool algorithm
  - categorical and continuous data
  - Custom loss and objective function
  - Perturbation control towards gradient
- Determining Accessibility Constraint
  - Correlation for metadata vs. decision
  - Value between 0 and 1
- ResNet as candidate ML method
- Two DLBAC datasets
  - System-1 and System-2



Data Cleaning / Pre-processing

- Categorical data : apply One-Hot Encoding
- Removed SENSORS permission's request, only 1 such sample
- **Conflicts exists** (~800 requests): adopt **grant-override** approach
- Introduce a new category name UNKNOWN for missing values
- Not all the features are related or usable (device ID, bootTime, answerType, etc.)

#	Name	Data Type	HasMissingValues
1	callState	Categorical	no
2	screenIsInteractive	boolean	no
3	networkStatus	Categorical	no
4	plugState	Categorical	no
5	selectedSemanticLoc	Categorical	no
6	category	Categorical	no
7	isTopAppRequestingApp	boolean	yes
8	isForeground	boolean	yes
9	isInEvent	boolean	yes
10	hour	Categorical	no
11	isWeekend	boolean	no
12	permission	Categorical	no

Input:

Reqs. Apps info, device's info, Permission

Output:

Access decision