SENSITIVE DATA EXTRACTION PREVENTION FROM MACHINE LEARNING BASED CLOUD SERVICE PROVIDERS

LAKSHMI KANTHAN V
MASTER OF COMPUTER APPLICATION,
DHANALAKSHMI SRINVASAN ENGINEERING COLLEGE,
PERAMBALUR.

Ms. M. BAVITHRA
ASSISTANT PROFESSOR,
MASTER OF COMPUTER APPLICATION,
DHANALAKSHMI SRINVASAN ENGINEERING COLLEGE,
PERAMBALUR

.

ABSTRACT: Prevent sensitive data extraction by cloud service providers using SDEP (sensitive data extraction prevention) algorithm. Deep Private-Feature Extractor (DPFE), a deep model which is trained and evaluated supported information theoretic constraints. Using the selective exchange of data between a user's device and a service provider, DPFE enables the user to stop certain sensitive information from being shared with a service provider, while allowing them to extract approved information using their model.

It utilize the log-rank privacy, a completely unique measure to assess the effectiveness of DPFE in removing sensitive information and compare different models supported their accuracy-privacy trade-off.by using both technique combined privacy and security is high. the info holder shares a public dataset: anonymity of people are threatened; to data holders participate during a model training procedure with their private data; a model provider shares a publicly-learned model: the privacy of the individuals' data used for training is at risk; an user shares his/her data with the service provider: private information are often revealed to the service provider; a service provider shares query answers with the top user.

KEYWORDS: Data extraction, Deep neural network architecture, Data privacy, SDEP (sensitive data extraction prevention) algorithm.

I. INTRODUCTION

The increasing collection of personal data generated by, or inferred from, our browsing habits, wearable devices, and smartphones, alongside the emergence of the data fromthe Internet of Things (IoT) devices are fuelling a wide range of novel applications and services. These include healthcare and wellbeing apps, financial management services, personalized content recommendations, and social networking tools. Many of these systems and apps rely on continuous data sensing and collection at the user side, and upload of the data to a service provider for consequent analysis.

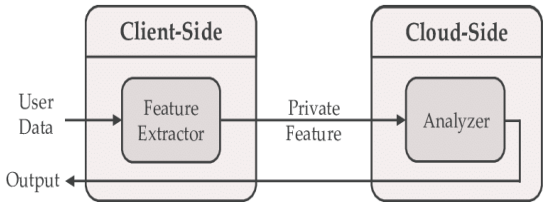

Fig : 1.1 client-side and cloud-side

II.METHODOLOGY

EXISTING SYSTEM

The main contributions of the existing system are:

1) A hybrid user-cloud framework for the user data privacy preservation problem which utilizes a private-feature extractor as its core component;
2) Designing the privatefeature extractor based on information theoretic concepts leading to an optimization problem;
3) Proposing a deep neural network architecture to solve the optimization problem; and
4) Proposing a measure to evaluate user privacy and verify the feature extractor module.

PROPOSED SYSTEM

Extra option will be provided to user for privacy preserving. SDEP (sensitive data extraction prevention) algorithm will extract sensitive data and replace it with noise data. this may enhance privacy for sensitive data . even if the permission from the user is granted. Prevent sensitive data extraction by cloud service providers using SDEP (sensitive data extraction prevention) algorithm. Deep Private-Feature Extractor (DPFE), a deep model which is trained and evaluated supported information theoretic constraints. Using the selective exchange of information between a user's device and a service provider, DPFE enables the user to forestall certain sensitive information from being shared with a service provider, while allowing them to extract approved information using their model.

We utilize the log-rank privacy, a novel measure to assess the effectiveness of DPFE in removing sensitive information and compare different models supported their accuracy-privacy trade-off.by using both technique combined privacy and security is high.

(i) data holder shares a public dataset: anonymity of individuals are threatened;

(ii) data holders participate during a model training procedure with their private data;

(iii) a model provider shares a publicly-learned model: the privacy of the individuals' data used for training is at risk;

(iv) an user shares his/her data with the service provider: private information may be revealed to the service provider;

(v) a service provider shares query answers with the end user

III.SYSTEM ARCHITECTURE

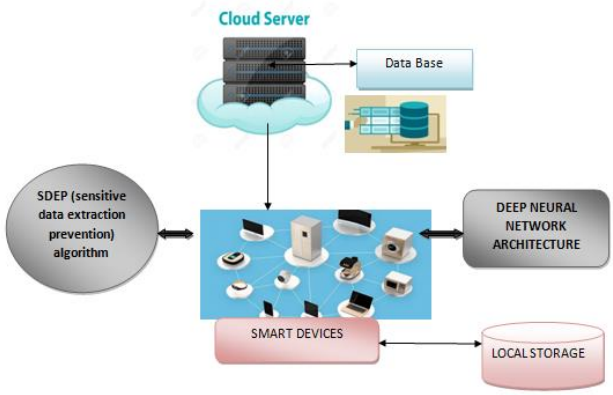

Fig : 3.1 system architecture

NETBEANS IDE

The java code has been written in a simple to use "NetBeans IDE": which may be a

reusable framework for simplifying the event of other desktop applications. When an application supported the NetBeans Platform is run, the platform's Main class is executed. Available modules are located, placed in an in-memory registry, and therefore the modules' begin tasks are executed. Generally, a module's code is loaded into memory only because it is required . Applications can install modules dynamically. Any application can include the Update Centre module to permit users of the appliance to download digitally signed upgrades and new features directly into the running application. Reinstalling an upgrade or a replacement release doesn't force users to download the whole application again

MYSQL

MySQL is that the most popular Open Source Relational SQL management System. MySQL is one among the simplest RDBMS getting used for developing various web-based software applications. MySQL is developed, marketed and supported by MySQL AB, which may be a Swedish company. The name of MySQL is that the combination of My and SQL, MySQL. MySQL may be a management system that permits you to manage relational databases. it's open source software backed by Oracle. It means you'll use MySQL without paying a dime.

THE THREE-OOP PRINCIPLE:

A.ENCAPSULATION:

Encapsulation is the mechanism that binds together code and thus the knowledge it manipulates and keeps both safes from outside interference and misuse. A method to believe encapsulation is as a protective wrapper that forestalls the code and data from beginning arbitrarily accessed by other code defined outside the wrapper. Access to the code and data inside the wrappers is tightly controlled through a well-defined interface. To relate this to the important world, consider the automated transmission of an automobile. It encapsulates many bits of knowledge about our engine, like what proportion you're accelerating, the pitch of the surface you're on, and thus the position of the shift lever.

B.INHERITANCE:

Inheritance is that the process by which one object acquires the properties of another object. This is often important because it supports the concept of hierarchical classification. As mentioned earlier, most knowledge is formed manageable by hierarchical classification. Inheritance interacts with encapsulation also. If a given class encapsulates some attributes, then the subclass will have the same attributes plus any that it adds as a neighborhood of its specialization. Java supports two sorts of inheritance.They are,

1) SINGLE INHERITANCE:

The derived class is inherited from one super class

BLOCK DIAGRAM:

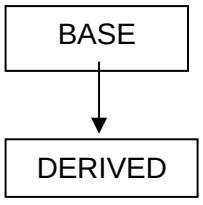

fig : 3.2 single inheritance block diagram

2) MULTILEVEL INHERITANCE:

This contains the hierarchical of classes.

BLOCK DIAGRAM:

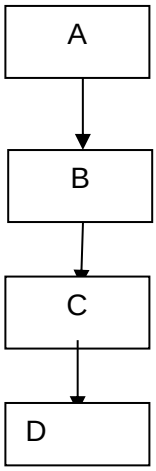

fig : 3.3 multilevel inheritance block diagram

IV.SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in the work product. It provides a way to check the functionality of components, sub-assemblies, and a finished product There are various types of test. Each type addresses a specific testing requirement.

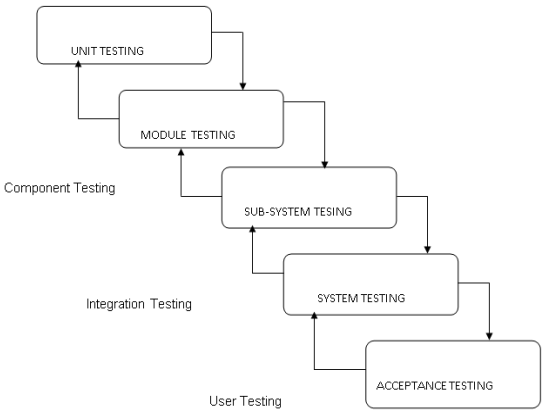

fig : 4.1 system testing

SYSTEM MAINTENANCE

System maintenance is an ongoing activity, which covers a wide variety of activities, including removing program and design errors, updating documentation and test data and updating user support. For the purpose of convenience, maintenance may be categorized into three classes, namely:

Corrective Maintenance

This type of maintenance implies removing errors in a program, which might have crept in the system due to faulty design or wrong assumptions. Thus, in corrective maintenance, processing or performance failures are repaired.

Adaptive Maintenance

In adaptive maintenance, program functions are changed to enable the information system to satisfy the information needs of the user.

V. CONCLUSION

The system uses the selective exchange of data between a user's device and a service provider, DPFE enables the user to forestall certain sensitive information from being shared with a service provider, while allowing them to extract approved information using their model.

To evaluate Simple and DPFE models, we designed the subsequent four experiments and assessed different models supported their accuracy-privacy trade-off: 1) the system compared Simple and DPFE models to point out the prevalence of DPFE fine-tuning; 2) the system assessed the effect of various intermediate layers to point the appropriateness of upper layers; 3) the system evaluated the effect of extending attribute set and showed that preserving privacy becomes harder; 4) the system considered mean and

variance of Rank-privacy measure to ensure privacy.

VI. REFERENCES

[1] N. Vallina-Rodriguez, J. Shah, A. Finamore, Y. Grunenberger, K. Papagiannaki, H. Haddadi, and J. Crowcroft, (2012) "Breaking for commercials: characterizing mobile advertising," in Proceedings ofthe 2012 Internet Measurement Conference. ACM, pp. 343–356.

[2] A. Acquisti, L. Brandimarte, and G. Loewenstein, (2015) "Privacy andhuman behavior in the age of information," Science, vol. 347, no.6221, pp. 509–514.

[3] M. Haris, H. Haddadi, and P. Hui, (2014) "Privacy leakage in mobilecomputing: Tools, methods, and characteristics," arXiv preprintarXiv:1410.4978.

[4] H. Haddadi and I. Brown, (2014) "Quantified self and the privacy challenge,"Technology Law Futures.

[5] F. D. Garcia and B. Jacobs, (2010) "Privacy-friendly energy-metering viahomomorphic encryption," in International Workshop on Security and Trust Management. Springer, pp. 226–238.

[6] C. Fontaine and F. Galand, (2007) "A survey of homomorphic encryptionfor nonspecialists," EURASIP Journal on Information Security, vol.2007, no. 1, p. 013801.

[7] P. Garcia Lopez, A. Montresor, D. Epema, A. Datta, T. Higashino, A. Iamnitchi, M. Barcellos, P. Felber, and E. Riviere, (2015) "Edge-centric computing: Vision and challenges," ACM SIGCOMM Computer Communication Review, vol. 45, no. 5, pp. 37–42.

[8] S. A. Osia, A. S. Shamsabadi, A. Taheri, H. R. Rabiee, N. Lane, andH. Haddadi, (2017) "A hybrid deep learning architecture for privacypreservingmobile analytics," arXiv preprint arXiv:1703.02952.

[9] S. A. Osia, A. S. Shamsabadi, A. Taheri, H. R. Rabiee, and H. Haddadi, (2018) "Private and scalable personal data analytics using a hybridedge-cloud deep learning," IEEE Computer.

[10] R. Agrawal and R. Srikant, (2000) "Privacy-preserving data mining," inACM Sigmod Record, vol. 29, no. 2. ACM, pp. 439–450.