IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 11, November 2015

A Survey on Privacy and Security of Data Classification

Brindha.M[1], Prof. S.V.Hemalatha[2]

PG scholar, Dept. of Computer Science and Engg., KalaignarKarunanidhi Institute of Technology, Coimbatore, India[1]

Asst. Prof., Dept. of Computer Science and Engg., KalaignarKarunanidhi Institute of Technology, Coimbatore, India[2]

Abstract: Data mining, the mining of hidden predictive information from huge databases, is a powerful new technology with grand latent to help companies focus on the most vital information in their data warehouses. Data mining tools expect future trends and behaviors, allowing businesses to make positive, knowledge-driven decisions. Text classification is the method of conveying text documents based on assured categories. Due to the rising trends in the field of internet and computers, billions of text data are processed at a known time and so there is a require for systematize these data to offer easy storage and accessing .Many text classification approaches were developed for efficiently solving the difficulty of identifying and classifying these data. During the data retrieval of the classified data, privacy and security is the challenging task. In this paper, a survey on privacy and security of the classified data using classification after encryption has been discussed. A classifier is used to define the suitable class for each text document based on the input algorithm used for classification. Encryption is the procedure of encoding messages or information in such a way that only approved parties can read it, which provides high security and privacy.

Keywords: privacy of data, classifier, encryption.

1. INTRODUCTION

Today's digital infrastructure supports novel ways of storing, dealing out, and disseminating data. In fact, we can store our data in remote servers, access consistent and capable services provided by third parties, and use computing authority accessible at multiple locations athwart the network. Furthermore, the increasing adoption of moveable devices jointly with the dispersal of wireless relations in home and work environments has led to a more dispersed computing scenario. These advantages come at a price of superior seclusion risks and vulnerabilities as a huge amount of information is being distributed and stored, often not under the straight control of its owner.

Ensuring proper seclusion and security of the information stored, converse, processed, and dispersed in the cloud as well as of the users accessing such an information is one of the grand confront of our modern society. As a issue of fact, the advancements in the Information Technology and the dispersal of novel paradigms such as data outsourcing and cloud computing, while permitting users and group to easily entrée high value submission and services, introduce novel seclusion risks of indecent information revelation and dissemination.

Classification is one of the usually used errands in data mining applications. For the past decade, due to the increase of diverse seclusion issues, many theoretical and practical solutions to the classification difficulty have been planned under different protection models. However, with the recent reputation of cloud computing, users now have the chance to outsource their data, in encrypted form, as well as the data mining errands to the cloud. Since the data on the cloud is in encrypted form, existing privacy-preserving classification methods are not valid.

In cryptography, encryption is the method of encoding messages or information in such a way that only official revelry can read it. Encryption does not of itself avert interception, but denies the message contented to the interceptor. In an encryption scheme, the proposed statement information or message, referred to as plaintext, is encrypted using an encryption algorithm; engender cipher text that can only be read if decrypted. For technical reasons, an encryption method regularly uses a pseudorandom encryption key produce by an algorithm. It is in opinion promising to decrypt the message without possessing the key, but, for a elegant encryption method, large computational possessions and skill are necessary. An authorized beneficiary can easily decrypt the message with the key provided by the inventor to beneficiary, but not to not permitted interceptors.

2. LITERATURE REVIEW

A. Privacy Preserving K-NN Classification

Bharath K. Samanthulla, Yousef Elmehdwi and Wei Jiang has described about the efficient way of classification task under data mining, and for privacy, encryption algorithm is used. They have proposed k-Nearest Neighbor classification technique over encrypted data for secrecy of data, speculation of clients input query and hide the entree patterns. The technique used here is SMIN$_n$ which improves the efficiency of retrieval of encrypted data. The proposed SMIN protocol is more intricate than further protocols and due to space boundaries, it required to offer its security evidence rather than providing proofs for every protocol. Therefore, here they include a agreed security proof for the SMIN protocol based on the regular replication argument. Thus the solution obtained by privacy-preserving k-NN (ppkNN) classification protocol

DOI 10.17148/IJARCCE.2015.41161

over encrypted data is evaluated the recital under different parameter.

B. Building Castles Out of Mud: Practical Access Pattern Privacy and Correctness on Untrusted Storage

Peter Williams, Radu Sion and Bogdan Carbunar introduce a new realistic method for secluded data storage with proficient entrée model seclusion and accuracy. A storage user can organize this method to concern encrypted reads, writes, and include to a potentially snooping and malicious storage service supplier, without instructive information or admittance patterns. The supplier is not capable to launch any association amid successive accesses, or even to discriminate between a read and a write. Moreover, the user is provided with sturdy correctness promise for its process – illegal supplier performance does not go unobserved. They built a first realistic system – orders of enormity faster than accessible implementations – that can carry out over numerous queries per second on 1Tbyte+ databases with full computational seclusion and rightness. To guard data stored in such an untrusted attendant model, protection systems should present users guarantee of data discretion and access seclusion. As a first line of guard, to ensure discretion, all data and related meta-data can be encrypted at the user side by non-malleable encryption, prior to being stored on the server. The data relics encrypted all over its lifetime on the server and are decrypted by the user upon recovery. Encryption provides vital seclusion guarantees at low cost. It however, is only a first step as important information is still leaked through the entrée pattern of encrypted data. For example, consider an adversarial storage provider that determines a particular section of the encrypted database match up to an alphabetically sorted keyword file. This is not difficult, particularly if the challenger has any information of the client-side software. The adversary can then compare plaintext keywords, identified by their position in the catalog, to documents, by observing which locations in the encrypted index are simplified when a new encrypted file is uploaded. In general, it is complicated to vault the amount of information leaked by entrée patterns. In existing work, one proposed approach for ensuring client entrée pattern seclusion (and confidentiality) tackles the case of a single-owner model. Particularly, service supplier hosts information for a customer, yet does not find out which items are entrée. Note that in this setup the customer has full managed and tenure over the data and other parties are talented to entrée the same data during this customer only. One major occasion of such mechanisms is Oblivious RAM (ORAM). For ease, in the ensuing they will use the term ORAM to tender to any such outsourced data procedure. One of the main drawbacks of offered ORAM techniques is their generally time difficulty. Exclusively, in real-world setups ORAM yields carrying out times of hundreds to thousands of seconds per solitary data access. In this paper they introduce a first realistic insensible data access practice with correctness. The key insights lie in new constructions and complicated reshuffling protocols that yield realistic computational difficulty (to O(log n log log n)) and storage space overheads (to O(n)). They also initiate a first practical realization that allows a throughput of more than a few queries per second on 1Tbyte+ databases, with full computational solitude and accuracy, orders of degree faster than obtainable approaches. The result obtained is they commence a first sensible oblivious data admission procedure with accuracy for obtaining the key insights lie in new assembly and primitive reshuffling practice that yield realistic computational difficulty and storage expenses.

C. Fully Homomorphic Encryption Using Ideal Lattices

C. Gentry propose a fully homomorphic encryption scheme – i.e., a plan that permit one to estimate course over encrypted data without being capable to decrypt. Our solution comes in three steps. First, they provide a common result – that, to raise an encryption method that permits assessment of chance circuits, it suffice to raise an encryption method that can estimate (slightly augmented versions of) its own decryption circuit; they call a method that can estimate its (augmented) decryption circuit bootstrappable. Next, they explain a public key encryption method using best lattices that is roughly bootstrappable. Lattice-based cryptosystems normally have decryption algorithms with short circuit complexity, habitually subjugated by an inner product calculation that is in NC1. Also, ideal lattices offer both additive and multiplicative homomorphisms (modulo a public-key model in a polynomial ring that is signified as a lattice), as wanted to estimate general circuits. Regrettably, our primary scheme is not fairly bootstrappable – i.e., the vigor that the method can properly estimate can be logarithmic in the lattice facet, just like the vigor of the decryption circuit, but the concluding is superior to the former. In the final step, they demonstrate how to vary the method to diminish the vigor of the decryption circuit, and thus find a bootstrappable encryption method, without tumbling the vigor that the method can evaluate. Conceptually, they accomplish this by enabling the encrypter to make the decryption course, send-off less effort for the decrypter, much similar to the server foliage less effort for the decrypter in a server-aided cryptosystem.They leave out full details due to lack of space, but mention that one can construct an algorithm RandomizeCTE for our E2 that can be functional to ciphertexts yield by Encrypt E 2 and Estimate E 2, correspondingly, that makes equivalent production distributions. The course privacy of E2 instantly implies the (leveled) circuit seclusion of our (leveled) fully homomorphic encryption method. The thought is simple: to create a casual encryption ψ 0 of π from a scrupulous encryption ψ of π, they purely add an encryption of 0 that has a much larger random "error" vector than ψ – super-polynomially better, so that the new error vectors statistically eliminate all information regarding ψ's error vector. This entails one more re-definition of CE.

D. Implementing Gentry's Fully-Homomorphic Encryption Scheme

C. Gentry and S. Halevi depict a effective accomplishment of a variant of Gentry's fully homomorphic encryption

DOI 10.17148/IJARCCE.2015.41161

method (STOC 2009), similar to the alternate used in an previous accomplishment effort by Smart and Vercauteren (PKC 2010). Smart and Vercauteren implemented the primary "somewhathomomorphic" method, but were not able to execute the bootstrapping functionality that is required to get the whole method to work. They show a number of optimizations that permit us to execute all aspects of the method, together with the bootstrapping functionality. Our foremost optimization is a key-generation process for the primary somewhat homomorphic encryption, that does not need full polynomial inversion. This reduces the asymptotic complexity from˜O(n2.5) to˜O(n1.5) when effective with dimension-n lattices (and virtually reducing the time from lots of hours/days to a few seconds/minutes). Other optimizations contain a batching technique for encryption, a vigilant analysis of the degree of the decryption polynomial, and various space/time trade-offs for the fully-homomorphic method. They hardened our accomplishment with lattices of numerous dimensions, consequent to numerous safety levels. From a "toy" scenery in dimension 512, to "small," "medium," and "large" sceneries in dimensions 2048, 8192, and 32768, respectively. The public-key size ranges in size from 70 Megabytes for the "small" scenery to 2.3 Gigabytes used for the "large" scenery. The moment to run one bootstrapping operation (on a 1-CPU 64-bit machine with large memory) range from 30 seconds for the "small" setting to 30 minutes for the "large" setting. Just before a bootstrappable scheme, Gentry described in a somewhat homomorphic method, which is usually a GGH-type process over perfect lattices. Gentry shortly proved that amid a suitable key-generation process, the safety of that method can be (quantumly) condensed to the worst-case solidity of some lattice troubles in perfect lattices. This somewhat homomorphic method is nevertheless bootstrappable, so Gentry described in a conversion to squash the decryption procedure, dropping the degree of the decryption polynomial. This is done by adding to the public key an additional hint regarding the secret key, in the appearance of a "sparse subset-sum" problem (SSSP). Namely the public key is increased with a big set of vectors, such that there exists a very sparse split of them that adds up to the furtive key. A ciphertext of the primary method can be "post-processed" using this additional hint, and the post-processed ciphertext can be decrypted with a low-degree polynomial, thus attaining a bootstrappable scheme.

E. SHAREMIND: A Framework for Fast Privacy-Preserving Computations.

D. Bogdanov, S. Laur, and J. Willemson, Gathering and dispensation responsive data is a tricky task. In fact, there is no common recipe for building the essential information systems. In this paper, they present a provably sheltered and proficient general-purpose computation scheme to address this problem. Our solution SHAREMIND is a essential machine for privacy-preserving data dispensation that relies on share computing techniques. This is a typical way for securely evaluating functions in a multi-party computation environment. The novelty of our elucidation is in the picking of the secret distribution method and the plan of the protocol suite. They have made many practical decisions to make large-scale split computing viable in practice. The protocols of SHAREMIND are information-theoretically safe in the honest-but-curious sculpt with three computing participants. Although the honest-but-curious sculpt does not accept cruel participants, it still provides considerably improved seclusion preservation when compared to typical centralized databases. Alternative is to judge the crisis as a multi-party computation task, where the data giver want to firmly summative data without instructive their secret inputs. Nevertheless, the analogous cryptographic elucidation rapidly becomes basically inflexible when the number of participants grows away from few hundreds. Moreover, data giver is frequently disinclined to keep on online throughout the whole computation and their computers can be simply taken more than by adversarial services. As a way out, they propose ahierarchical result, where all computations are prepared by enthusiastic minor parties who are less liable for peripheral sleaze. More specifically, they suppose that merely a few minor parties can be ruined throughout the computation. Consequently, they can use furtive sharing and split computing techniques for privacy-preserving data aggregation. In scrupulous, data givers can securely offer their inputs by transferring the equivalent shares to the miners. As a result, the miners can firmly evaluate any combined value without more interfaces with the data givers. They have proposed a novel approach for rising privacy-preserving applications. The SHAREMIND framework relies on protected multi-party computation, but it also introduces numerous new ideas for civilizing the effectiveness of both the applications and their maturity process. The main hypothetical involvement of the structure is a suite of computation protocols functioning over rudiments in the ring of 32-bit integers instead of the normal finite field. This non-standard option permitted us to build easy and proficient protocols. They have also implemented a fully useful prototype of SHAREMIND and showed that it offers superior performance when compared to extra similar frameworks. Besides that, SHAREMIND also has an easy to use function development crossing point allowing the programmer to deliberate on the implementation of data mining algorithms and not to fret about the seclusion issues. However, the recent implementation has some limitations; most remarkably it can apply only three computing parties and can contract with just one semi-honest rival. Hence the main trend for prospect research is soothing these restrictions by rising computational primitives for further three parties. They will also need to study the potential for given that protection guarantees beside energetic adversaries. An additional aspect needing auxiliary enhancement is the application programmer's interface. A compiler from a higher-level language to our present assembly-like instruction set is absolutely needed. Implementing and benchmarking a wide range of existing data-mining algorithms will stay behind the theme for further growth as well.

DOI 10.17148/IJARCCE.2015.41161

F. Privacy Preserving Query Processing

Haibo Hu, Jianliang Xu, Chushi Ren and Byron Choi has described the data are secretive assets of the data holder and should be secluded beside the cloud and querying user; Alternatively, the query force reveal sensitive information of the client and should be secluded near the cloud and data holder**.** Query dispensations that conserve both the data seclusion of the proprietor and the query seclusion of the client are a new examine problem. They proposed a holistic and proficient solution that comprises a protected traversal support and an encryption method based on seclusion homomorphism. The framework is scalable to huge datasets by leveraging an index-based loom. The method is to let the user guide in the distance access and keep path of the traversal path, so that neither the data holder nor the cloud knows the precise node the client is accessing, let alone the query point. On the other hand, to protect data seclusion, the client has only access to an encrypted version of the register, and must progress the query processing together with the cloud that can decrypt the distances it computes nearby. As such, the reserve entrée is a joint process of the user and data cloud, in which neither party has admittance to the concrete distances. The solution obtained is a protected index traversal framework, based on which protected protocols are devised for typical types of queries. Through speculative proofs and recital evaluation, that approach is shown to be not only realistic, but also proficient and vigorous under diverse parameter settings.

Table 1: comparison of various techniques for privacy and security of encrypted data

S.NO	AUTHOR	ADVANTAGE	DISADVANTAGE	TECHNIQUE USED
1	Bharath K. Samanthulla, Yousef Elmehdwi and Wei Jiang	Protects data confidentiality, user's query privacy, and hides data access patterns.	Computation Cost is quite high and poor run time performance.	Privacy Preserving k-NN classification.
2	Peter Williams, Radu Sion and Bogdan Carbunar	Efficiency and privacy was improved.	Computational complexity and privacy leak.	Oblivious data access protocol.
3	Craig Gentry	Security level is good and ciphertext scheme for use keys.	Computational complexity and untrusted server scheme.	Fully homomorphic encryption scheme.
4	Craig Gentry and Shai Halevi	space-efficient and running-time advantage of the optimization.	Ciphertext size is high.	Key-generation method for the homomorphic Encryption.
5	Dan Bogdanov, Sven Laur, and Jan Willemson	Easy to use application development interface and performance improved.	Did not providing security guarantees against active adversaries and application programmer's interface.	Sharemind protocol
6.	Haibo Hu, Jianliang Xu, Chushi Ren, Byron Choi	Improve the efficiency of the query processing and performance.	Did not provide Mutual privacy protection for queries on senior Unstructured datasets.	Processing Private Queries over Untrusted Data Cloud through Privacy Homomorphism.

DOI 10.17148/IJARCCE.2015.41161

3. VARIOUS PRIVACY RISKS

The privacy is an individual's precise to manage the gathering, use and revelation of information regarding him or herself may present risk, seems to be a new one to many. That it should be a new model to those responsible for association risk, however, needs to be addressed. Personal information is a benefit, the worth of which is secluded and superior by a group of safety practices and trade processes. Like other equipped risks, those allied to the safety of special information profit from the analysis of a official risk management regulation. Issues of seclusion are associated with the supervision of Personal Information (PI), which describes much of the data composed by an association concerning its employees, prediction and customers. Its characteristic is that it can be associated to a particular individual, either honestly or ultimately. While all PI is to be cherished, some rudiments are considered particularly sensitive, warranting special care and therefore, presenting added risk.

A. Privacy Risks for Users

Attribute-based access control: Established approaches for modifiable contact to assets are based on user verification and therefore cannot be adopted in the cloud, where the interacting parties can be unrevealed to each other. Notice has been then agreed to leaving from user authentication and, in the name of seclusion and practicality, providing admission organizes solutions behind credential-based and attribute-based provisions. In this way, users can easily contact all the resources available from servers lacking the need to memorize passwords or supervise an accurate report for each of the servers they admittance. In fact, identification permits a server to confirm whether the user requesting admission to a service satisfies the condition necessary to gain the admission. Consideration has been also dedicated to the use of *anonymous credentials,* essential privacy-enhanced solutions and mechanisms for record definition and management. Anonymous credentials enable a user to selectively reveal subsets of attributes from a credential, and even to prove that the attributes restricted in a credential please a certain condition without instructive the exact quality values. The combination of anonymous credentials with access direct policy languages is a problem.

User's privacy preferences: The release of user's personal information is often matched by approaches that can be seen as symmetric to the ones adopted by servers for modifying the revelation of resources or services. However, access control-like stipulation do not entirely fit the users' security requirements, since they may need a way to identify *partialities* on the information to disclose based on the sympathy of such information. For instance, a user may desire to disclose her identity card over her passport if both identifications can allow the access to the requested service. Few proposals have tackled this issue. The proposal in authorize a user to connect a different cost with each record in her collection representing its sympathy and to reduce the total cost of a concession process. The solution obtained is a minimum discovery is a division of identification and assets in the user collection whose discharge allows the user to access the service requested, while minimizing the warmth of the set of essentials released. Although this solution enables users to categorize and control all their testimonial and control their release, there are still numerous open issues that need to be addressed.

B. Privacy Risks for Stored Data

Confidentiality and integrity: The data stored and managed by a cloud server can comprise responsive information that neither the cloud server nor unofficial users should read. The difficulty of defensive data privacy from the eyes of the storage server has earliest been addressed in the Database as a Service scenario. The proposed solutions consist in encrypting the data before storing them at the exterior server. In this way, only official users, who know the decryption key, can access the data satisfied. Indexing information is then attached with the encrypted data to permit the server to moderately estimate users' queries directly on the encrypted data. The solutions planned to defend the sensitive relations while preventive the acceptance of encryption are based on the mutual use of encryption and destruction.

Selective access: In a cloud circumstances neither the data proprietor nor the cloud server can implement the owner's access direct policy, for confidentiality or presentation reasons, correspondingly. In fact, the cloud server cannot frankly enforce access control limitations because it force not be trusted to enforce them and also because the policy modifiable access to the data may depend on the data satisfied. The data owner would instead want to arbitrate every access demand to filter the uncertainty result, thus nullifying the compensation of storing data at an external server. It is therefore needed to plan a mechanism such that the data themselves implement limitations on the set of users who can access them. The solutions planned are based on discriminating encryption. Discriminating encryption consists in encrypting diverse portions of the data using different keys, and in allocated keys to users in such a way that each user can decrypt all and only the pieces of information she is allowed to access. This approach is attached with key derivation techniques for providing competent access to the data

C. Privacy Risks for Data Accesses

Integrity: The cloud server evaluating a query may be lazy and estimate the user's query on a subset of the data to save computational resources, or it could perform the query on a non-up-to-date report of the data. It is consequently essential to define a mechanism that consents users to check the reliability of query outcome.

Query outcomes assure veracity checks if they are: correct, complete and fresh. The veracity of query consequences in a cloud scenario is a difficulty that has been addressed only lately. Correctness is typically provided by digital signature approaches. Completeness can be provided either by defining valid data structures on the data, or by

DOI 10.17148/IJARCCE.2015.41161

inserting sentry in the data compilation that must also belong to query outcomes. Authenticated data organization approaches have the advantage of providing a full agreement of query completeness, in difference to the probabilistic promise accessible by sentinels. However, authenticated data structures are less supple than sentinels, since they offer veracity assurances only for queries working on the characteristic on which the structure has been defined, and have a higher administration cost. In fact, valid data structures cannot easily contain updates to the outsourced data. Freshness is provided by making authenticated data structures and/or sentry reliant on an erratic that modify over time.

Query privacy: A significant issue that wants to be addressed when data are stored at outside cloud servers consists in preserving the seclusion of the accesses themselves. Queries can be oppressed for performing different types of conclusions. The first type of conclusion can arise in circumstances where the accessed data can be either secret or not. As an example, consider a medical database stored at a cloud server and suppose that a user accesses it looking for dealing and cures for stomach ulcer. The user’s query discloses to an witness that either the user or a person close to her endures from stomach ulcer. The second type of inference applies when the accessed data are secret and are encrypted. Queries can be subjugated for inferring information about the data content. In this case, it is not enough to protect the privacy of queries especially taken, but it is also necessary to protect the privacy of patterns of accesses, meaning that the server should not be able to infer whether two different queries designed at the same intention information. In fact, by observing accesses to the data, the cloud server could develop the information on the frequencies with which diverse pieces of information are accessed to reconstruct the association between the plaintext data and the encrypted data. Solutions planned for defensive access and pattern privacy when data are stored in the clear are classically based on Private Information Retrieval (PIR) techniques and provide entrée and pattern privacy at a high computational and communication cost. Suggestions operating on secret data introduce privacy-preserving indexing techniques able to compute the query effect while providing assurance of access and pattern privacy. One of these solutions consists in the definition of a hobble index for data group and storage.

4. CONCLUSION

To protect user privacy, various privacy-preserving classification techniques have been proposed over the past decade. This survey paper provides the characterization of encryption techniques with its pros and cons. This paper also describes about various privacy risks obtained during data storing and retrieval.

REFERENCES

[1] Bharath K. Samanthula, Yousef Elmehdwi, and Wei Jiang, “k-Nearest Neighbor Classification Over Semantically Secure Encrypted Relational Data”, ieee transaction, vol 27,no 5,p 1261-1273, 2015.

[2] P. Williams, R. Sion, and B. Carbunar, “Building castles out of mud: Practical access pattern privacy and correctness on untrustedstorage,” in Proc. 15th ACM Conf. Comput. Commun. Security, 2008, pp. 139–148.

[3] C. Gentry, “Fully homomorphic encryption using ideal lattices,” in Proc. 41st Annu. ACM Sympos. Theory Comput., 2009, pp. 169–178.

[4] C. Gentry and S. Halevi, “Implementing gentry’s fully-homomorphic encryption scheme,” in Proc. 30th Annu. Int. Conf. Theory Appl. Cryptographic Techn.: Adv. Cryptol., 2011, pp. 129–148.

[5] D. Bogdanov, S. Laur, and J. Willemson, “Sharemind: A framework for fast privacy-preserving computations,” in Proc. 13th Eur.Symp. Res. Comput. Security: Comput. Security, 2008, pp. 192–206.

[6] H. Hu, J. Xu, C. Ren, and B. Choi, “Processing private queries over untrusted data cloud through privacy homomorphism,” in Proc. IEEE 27th Int. Conf. Data Eng., 2011, pp. 601–612.

[7] S. De Capitani di Vimercati, S. Foresti, and P.Samarati, “Managing and accessing data in the cloud: Privacy risks and a approaches,” in Proc. 7th Int.Conf. Risk Security Internet Syst., 2012, pp. 1–9.

DOI 10.17148/IJARCCE.2015.41161