ISSN: 2321-7782 (Online)

Volume 3, Issue 11, November 2015

International Journal of Advance Research in Computer Science and Management Studies

Research Article / Survey Paper / Case Study

Available online at: www.ijarcsms.com

Survey on CBIR using K-Secure Sum Protocol in Privacy Preserving Framework

Supriya G. More[1]
ME Student in Computer Engg., ACEM
Pune, India

Ismail Mohammed[2]
Prof. In Computer Engg Dept., ACEM
Pune, India

Abstract: We propose a privacy protection framework for large-scale content-based information retrieval(CBIR). It offers two layers of protection. To begin with, robust hash values are utilized as queries to avoid uncovering unique content or features. Second, the customer can choose to exclude certain bits in a hash values to further expand the ambiguity for the server. Due to the reduced information, it is computationally difficult for the server to know the customer's interest. The server needs to give back the hash values of every single possible to the customer. The customer performs a search within the candidate list to locate the best match. Since just hash values are exchanged between the client and the server, the privacy of both sides is ensured. It is acknowledged through hash-based piecewise inverted indexing. The thought is to gap a highlight vector into pieces and list every piece with a sub hash value. Each sub hash value is connected with an inverted index list. The framework has been broadly tested using a large scale image database. We have assessed both retrieval performance and privacy-preserving performance for a specific content identification application. Two unique developments of robust hash algorithms are utilized. One depends on random projections; the other depends on the discrete wavelet transform. We consider the majority voting attack for evaluating the query category and identification.

I. INTRODUCTION

In the Internet era, multimedia content is massively produced and distributed. In order to efficiently locate content in a large-scale database, content-based search techniques have been developed. They are used by content based information retrieval (CBIR) [1] systems to complement conventional keyword-based techniques in applications such as near-duplicate detection, automatic annotation, recommendation, etc. In such a typical scenario, a user could provide a retrieval system with a set of criteria or examples as a query; the system returns relevant information from the database as an answer. Recently, with the emergence of new applications, an issue with content-based search has arisen sometimes the query or the database contains privacy-sensitive information [3][1]. In a networked environment, the roles of the database owner, the database user, and the database service provider can be taken by different parties, who do not necessarily trust each other. A privacy issue arises when an untrusted party wants to access the private information of another party. In that case, measures should be taken to protect the corresponding information.

The main challenge is that the search has to be performed without revealing the original query or the database. This motivates the need for privacy-preserving CBIR (PCBIR) systems. Privacy raised early attention in biometric systems, where the query and the database contain biometric identifiers. Biometric systems rarely keep data in the clear, fearing thefts of such highly valuable data. Similarly, a user is reluctant in sending his biometric template in the clear. Conventionally, biometric systems [5] rely on cryptographic primitives to protect the database of templates. In the multimedia domain, privacy issues recently emerged in content recommendation. With recommendation systems, users are typically profiled. Profiles are sent to service providers, which send back personalized content. Users are today forced to trust the service providers for the use of their profiles. Although CBIR systems have not been widely deployed yet, similar threats exist. Recently, the one-way privacy model for CBIR was investigated [1]. The one-way privacy setting assumes that only the user wants to keep his information secret because the database is public. Public databases against which users may wish to run private queries have become commonplace nowadays. Some of them already integrate similarity search mechanisms, such as Google Images or Google Goggles. It is likely that others will soon follow that path, turning Flickr, YouTube, Facebook into content-based searchable collections (in addition to already being tag searchable). Put in a larger picture, PCBIR [1] is one of many aspects on privacy protection in the big data era where profiling becomes ubiquitous. For example, recent news claims that advertisers and Facebook can generate user profiles of political opinions and behaviors. Latest research discovers that websites are actually fingerprinting users on the Internet by their system (e.g. browser) configurations. There is already some initiatives in web search privacy. The trend shows that privacy protection will become an indispensable part of future content-based search systems.

II. Problem Statement

A privacy issue arises when an untrusted party wants to access the private information of another party. In that case, measures should be taken to protect the corresponding information. The main challenge is that the search has to be performed without revealing the original query or the database. This motivates the need for privacy-preserving CBIR (PCBIR) systems. In order to protect privacy, original content cannot be used as queries. Sometimes even features are not safe, because they still reveal information about the original content. Instead of encryption, we generate queries from original content by robust hashing.

III. Relevant Objectives

We propose a privacy protection framework for large-scale content-based information retrieval. It offers two layers of protection. First, robust hash values are used as queries to prevent revealing original content or features. Second,the client can choose to omit certain bits in a hash value to further increase the ambiguity for the server.

IV. Solving Approach

The one-way privacy setting assumes that only the user wants to keep his information secret because the database is public. Public databases against which users may wish to run private queries have become commonplace nowadays. Some of them already integrate similarity search mechanisms, such as Google Images or Google Goggles. It is likely that others will soon follow that path, turning Flickr, YouTube, Facebook into content-based searchable collections (in addition to already being tag searchable). Put in a larger picture, PCBIR is one of many aspects on privacy protection in the big data era where profiling becomes ubiquitous.

V. Literature Survey

Sr. No.	Paper Name	Published Year	Author	Description
1	A Privacy-Preserving Framework for Large-Scale Content-Based Information Retrieval	2015	Li Weng, Laurent Amsaleg, April Morton, and Stéphane Marchand-Maillet	In this work, author proposed a privacy-enhancing framework for large-scale content-based information retrieval. It can be used for any CBIR system based on features and similaritysearch. The framework is mainly based on robust hashing and piece-wise inverted indexing.
2	One-Way Private Media Search on Public Databases	2013	Giulia Fanti, Matthieu Finiasz, and Kannan Ramchandran	The objective of this article is to argue that one-way-private content-based media classification is both inevitable and feasible.

3	Image Feature Extraction in Encrypted Domain With Privacy-Preserving SIFT	2012	Chao-Yong Hsu, Chun-Shien Lu	In this paper, a homomorphic encryption-based privacy-preserving SIFT (PPSIFT) approach to deal with the privacy-preserving problem encountered in a cloud computing environment, where the server can finish the tasks of SIFTbased applications without learning anything to breach the user's privacy.
4	Data-Oriented Locality Sensitive Hashing	2010	Wei Zhang, Ke Gao, Yong-dong Zhang, and Jin-tao Li	The author presented a new Data-Oriented LSH index which reduces heavy memory cost of Euclidean LSH caused by randomly selected projecting directions.

VI. The Proposed Framework

A naive solution to the client privacy problem is that the server sends the whole database to the client. This is not feasible for a limited bandwidth and violates the server privacy P3. Even if the client obtains the whole database, it may not be able to store or process the database. A compromise is that the client removes some details from the query to create some ambiguity for the server. A potential privacy-enhanced search procedure works as follows:

1. The client creates a partial query, and sends it to the server;
2. The server generates an extended query list based on the partial query;
3. The server performs a search with the extended query list, and sends back all matching items;
4. The client performs a search within the received set of matching results using the original query

On the one hand, the partial query enables the server to narrow down enough the search scope for the sake of performance; on the other hand, it should be difficult for the server to infer the original query. The information in the partial query is a measure of privacy related to P1.

The server sends all items that match the extended query list to the client. We denote the set of matching items by A. A must be large enough to create sufficient ambiguity for the server, but it must be feasible for the client to find the final match (i.e., A should not be too large). The information in A is another measure of privacy – it is proportional to P2 and inversely proportional to P3.

A good system should keep P1, P2, andP3 sufficiently large. Note that P1 is a necessary condition for P2. Increasing P1 also increases P2 and decreases P3, because the size of the matching set (denoted as |A|) increases. In practice, |A| is upper bounded by the available bandwidth, the computing power of the client, and the size of the database; it is lower bounded by the minimum privacy requirement.

Specifically, there are the following requirements on the partial query:

1. It is difficult to infer the original query;
2. It is feasible to generate and perform search with the extended query list;
3. The properties of A, e.g. the size and the diversity, can be controlled by the partial query;
4. It is easy to estimate P1.

There are the following requirements on the matching set A:

1. A should be compact enough to save bandwidth;
2. A contains the best answers, e.g., the (approximate) nearest neighbors;
3. The diversity of elements in A is sufficiently large;
4. The server cannot tell which are the best answers by analyzing A;
5. A should not reveal too much information about the database.

The above requirements are achieved by the rest of the framework, which consists of query generation, database indexing, and database search. They are described in the following. Figure 1 shows a schematic diagram of the framework.

a) Query Generation

In order to protect privacy, original content cannot be used as queries. Sometimes even features are not safe, because they still reveal information about the original content [7], [8]. Instead of encryption, we generate queries from original content by robust hashing. Robust hashing is also called perceptual hashing [9], [10] or robust fingerprinting [11], [12] (for multimedia data), or locality-sensitive hashing (LSH) [13] (for generic data). It is a framework that maps multimedia data to compact hash values.

Ideally, a robust hash value is a short string of equally probable and independent bits. It can be used to persistently identify or authenticate the underlying content, just like a "fingerprint". The basic property of robust hashing is that similar content should result in similar hash values. More importantly, hash algorithms have the one-way property that it is computationally difficult to infer the input from the output, because hashing is essentially a many-to-one mapping.

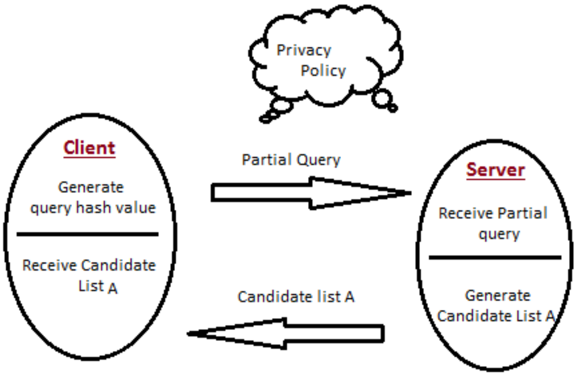

Fig. 5.1. A Schematic diagram of proposed framework

b) Database Indexing

The database indexing is based on the concept of piece-wise inverted indexing. We assume there is a general feature extraction component. The extracted feature vectors are capable of characterizing the underlyingcontent. They first undergo an orthogonal transform and dimension reduction. Only significant features are preserved. The elements of a featurevectorare dividedinto n groups.A robusthash value hi (i = 0,1,···,n − 1) is computed from the ith group. We call it asub-hash value. The above step creates a new coordinate system, with each coordinate represented by a sub- hash value. Finally, a multimedia object in the database is indexed by the overall hash value H = h0||h1||···||hn−1, i.e., the concatenation of sub-hash values.

Each sub-hash value is associated with an inverted index list (also called a hash bucket). The list contains the IDs (identification information) of multimedia objects corresponding to the sub-hash value. The size of a sub-hash value l depends on the significance of its corresponding feature elements.

c) Database Search

When privacy protection is not required, the proposed framework can work as efficiently as a normal CBIR scheme. In general, there are several possibilities to perform database search. They mainly differ in the domain for distance computation, which can be the feature space, the quantized feature space, or the hash space. In order to facilitate the explanation, we assume that an original query is a hash value. It can be generated by the client, or the server. In the former case, P1 is still preserved, but there is no guarantee for P2. While in the latter case, since the client sends the original content to the server, no privacy is preserved for the client.

1) Approximate Nearest Neighbor Search: When the server receives a hash value, it checks the table for each sub- hash value and optionally performs a nearest neighbor search within a Hamming sphere. For each binary sub-hash value, the multimedia object IDs within a small Hamming radius r are retrieved. When r ≥ 1, we call it multi-probing, because this is similar to the concept of multi-probe LSH [14]–[16]. Additionally, when side information is available, different policies can be applied to prioritize sub-hash values in the neighborhood.

2) Approximate Nearest Neighbor Search With Privacy: When privacy protection is "turned on", the hash value of the query content must be generated by the client. A partial query is then formed by omitting some bits in one or more sub-hash values according to a privacy policy. In general, the more bits are missing, the more client privacy (P1, P2) is preserved. The partial hash value is sent to the server along with the privacy policy, i.e., positions of the absent bits. If b bits are omitted from each sub-hash value, the server has to check 2b · n hash buckets. All the candidate IDs are sent back to the client, together with the corresponding hash values. The client eventually performs a search by comparing the hash values in the list with the original one.

VII. CONCLUSION

The framework has been implemented and extensively evaluated in different scenarios. We show that the privacy level, e.g., the number and the diversity of candidates can be tuned by the privacy policy. Some guidelines are given on how to choose the omitted bits. We have demonstrated both retrieval performance and privacy-preserving performance for a particular content identification application. Experiment results show that query items with near-duplicates are likely to be vulnerable to majority voting. The chance of success is equivalent to the chance that a query item has more near-duplicates than other irrelevant items in the candidate list. The results also show that the success rate decreases with the number of omitted bits and the number of distinct items.

References

1. Li Weng, Member, IEEE, Laurent Amsaleg, April Morton, and Stéphane Marchand-Maillet, "A Privacy-Preserving Framework for Large-Scale Content-Based Information Retrieval" at IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL. 10, NO. 1, JANUARY 2015.
2. G. Fanti, M. Finiasz, and K. Ramchandran, "One-way private media search on public databases: The role of signal processing," IEEE Signal Process. Mag., vol. 30, no. 2, pp. 53–61, Mar. 2013.
3. C.-Y. Hsu, C.-S. Lu, and S.-C. Pei, "Image feature extraction in encrypted domain with privacy-preserving SIFT," IEEE Trans. Image Process., vol. 21, no. 11, pp. 4593–4607, Nov. 2012.
4. W. Zhang, K. Gao, Y.-D. Zhang, and J.-T. Li, "Data-oriented locality sensitive hashing," in Proc. ACM Int. Conf. Multimedia, 2010, pp. 1131–1134.
5. M. Diephuis, S. Voloshynovskiy, O. Koval, and F. Beekhof, "DCT sign based robust privacy preserving image copy detection for cloud-based systems," in Proc. 10th Workshop Content-Based Multimedia Indexing (CBMI), Jun. 2012, pp. 1–6.
6. J. Bringer, H. Chabanne, and A. Patey, "Privacy-preserving biometric identification using secure multiparty computation: An overview and recent trends," IEEE Signal Process. Mag., vol. 30, no. 2, pp. 42–52, Mar. 2013.