Multi-Keyword Top-K Similarity Search With A Privacy Preserving Over Encrypted Data

K. Ramya Laxmi[1*], M Bhanusree Reddy[2], Kasarla Sindhu Reddy[3], Matta Prabhu Teja[4], Y Rithima [5]

Abstract

In order to support a variety of big data applications in industries like health care and scientific research, cloud computing offers individuals and businesses enormous computing power and scalable storage capacities. As a result, an increasing number of data owners are outsourcing their data to cloud servers for great convenience in data management and mining. However, sensitive information is frequently present in data sets like health records in electronic documents, raising privacy problems if the documents are made public or shared with partially untrusted third parties on the cloud. The multi-keyword top-k search challenge for big data encryption against privacy breaches is examined in this project in an effort to provide an effective and safe solution. We specifically build a special tree-based index structure and random traversal algorithm for the privacy concern of query data, which makes even the same query produce different visiting paths on the index and can also keep query accuracy unchanged under stronger privacy. We suggest a group multi-keyword top-k search method based on the concept of partition, where a group of tree-based indexes is built for all documents, in order to increase query efficiency. Finally, we combine these techniques into a secure and effective strategy to tackle our suggested top-k similarity search. Extensive experimental results on real-world data sets show that our suggested strategy can greatly outperform state-of-the-art techniques in terms of privacy protection, scalability, and query processing performance.

Keywords: Cloud computing, privacy preserving, data encryption, ranked search, multi-keyword top- K search, random traversal.

[1*]Assistant Professor, Dept. of CSE,Sreyas Institute of Engineering and Technology, Nagole, Hyderabad.
[2,3,4,5] B. Tech students, Dept. of CSE, Sreyas Institute of Engineering and Technology, Nagole, Hyderabad.

***Corresponding Author:** K. Ramya Laxmi
*Assistant Professor, Dept. of CSE, Sreyas Institute of Engineering and Technology, Nagole, Hyderabad.

INTRODUCTION

Recently, cloud computing has become a disruptive trend in both the IT and research communities. Its key features, such as high scalability and pay-as-you-go models, have made it possible for cloud consumers to buy powerful computing resources as services in accordance with their actual needs. As a result, cloud users no longer have to worry about the wastage of computing resources or the complexity of managing hardware platform. To make data administration, data mining, and query processing easier, more and more businesses and people are now outsourcing their data and deploying their services to cloud servers. However, in order to fully appreciate these benefits of cloud computing, businesses and consumers must also consider the privacy issue posed by the outsourced data. Due to the prevalence of sensitive data sets in many applications, such as emails, electronic health records, and financial transaction records, when the data owner outsources such sensitive data to the cloud servers that are only partially trusted, the data is easily accessible and can be unlawfully accessed and analysed by cloud service providers. Data Xiaofeng Ding, Peng Liu, and Hai Jin are with the Services Computing Technology and System Lab, Cluster and Grid Computing Lab, the School of Computer Science and Technology, Huazhong University of Science and Technology because the analysis of these data sets may provide profound insights into a number of important areas in society (such as e-research, healthcare, medical, and government services). Before releasing their data to the cloud, owners require services that are efficient, scalable, and privacy-preserving. Data encryption refers to a mathematical calculation and algorithmic system that convert plaintext into cyphertext, which is an unreadable form to unauthorized parties. It has been widely utilised for data privacy preservation in data sharing scenarios. Before outsourcing to cloud servers, data is encrypted using one of the many data encryption schemes that have been offered. However, using these technologies for data encryption typically comes with a high cost in terms of data utility, making it difficult to use traditional data processing techniques that were created for unencrypted data. In many database and information retrieval applications, the keyword-based search is a crucial data operator, but its conventional processing techniques cannot be directly applied to encrypted data. As a result, a popular area of research is how to handle such queries over encrypted data while yet ensuring data privacy. Fortunately, numerous searchable encryption-based approaches have been researched. Deal with single keyword searches, for instance, and multi-keyword Boolean searches are supported. The boolean search, on the other hand, is unrealistic since it results in large communication costs, and the single phrase search is not intelligent enough to allow sophisticated queries. In light of this, more recent efforts like concentrate on multikeyword ranked search, which is more useful in the pay-as-you-go cloud paradigm. However, the majority of these approaches are unable to simultaneously provide good data security and high search efficiency, particularly when used with large amounts of encrypted data.

I. PROBLEM STATEMENT

When dealing with encrypted data, conventional data processing methods that were designed for unencrypted data no longer work properly. Since using these methods for data encryption generally results in significant costs for the use of the data. The majority of these methods, especially when applied to substantial volumes of encrypted data, are unable to deliver excellent data security and great search efficiency simultaneously. Building a trapdoor takes a lot of time, and searching still costs a lot of money.

II. EXISTING SYSTEM

Data encryption refers to mathematical calculations and algorithmic systems that convert plaintext into cipher text, which is an unreadable form for unauthorised parties, and has been widely employed for data privacy preservation in data-sharing scenarios. However, using these methods to encrypt data typically comes at a high cost to the usefulness of the material. In many databases and information retrieval applications, the keyword-based search is a common data operator, but its conventional processing methods cannot be directly applied to encrypted data. As a result, a popular area of research is how to handle such queries over encrypted data while yet ensuring data privacy. Fortunately, numerous searchable encryption-based approaches have been researched.

For example, deal with the single keyword search, and works to support the multi-keyword Boolean search. However, the single keyword search is not smart enough to support advanced queries and the Boolean search is unrealistic since it causes high communication costs. Therefore, more recent works focus on the multi-keyword ranked search, which is more practical in the pay-as-you-go cloud paradigm. But most of these methods cannot meet the high search efficiency and the strong data security simultaneously, especially when applying them to big data encryption poses great scalability and efficiency challenges.

III. PROPOSED SYSTEM

The multi-keyword top-k search, which has become a very popular database operator in many significant applications, is the specific sort of multi-keyword ranked search that is the subject of this study. This search simply has to return the k documents with the greatest relevance scores. We offer the vector space approach, which represents documents and queries as vectors, to facilitate multi-keyword search. The relevance scores between documents and queries should be computed to facilitate top-k search.

Additionally, we suggest a group multi-keyword top-k search scheme (GMTS), which is based on partition and provides top-k similarity search over encrypted data, to increase query efficiency for improved user experiences. In this plan, the data owner creates searchable indexes for each group after dividing the dictionary's keywords into various groups. To increase data security, we suggest the random traversal algorithm (RTRA), in which the data owner creates a binary tree as a searchable index and gives each node a random switch so that the data user can provide each query a random key.

In order to preserve the high accuracy of inquiries, the data user can alter the outcomes and navigational paths of queries by utilising various keys. Finally, we combine the GMTS and the RTRA to create a reliable and effective solution to the issue we have raised.

IV. IMPLEMENTATION MODULES

Data Owner

The data owner uploads document collection D to the cloud server, but this collection may contain sensitive information. To protect data privacy, the data owner has to encrypt D before outsourcing it to the cloud server. Furthermore, in order to enable the cloud server to process query efficiently over the encrypted document collection C, the data owner constructs an encrypted searchable index Ie locally. Finally, the data owner outsources both the encrypted document collection C and the encrypted searchable index Ie to cloud, and shares the secret key of trapdoor generation and document decryption to authorized data users with secure channels.

Cloud Server

Cloud server is considered to be ”honest-but-curious” as adopted in most works on secure cloud data search. The server is honest as it runs the programs and algorithms correctly, it is curious since the cloud service providers can easily access and analyse the encrypted data, and even record queries to learn additional information. The cloud server calculates the relevance scores between trapdoor T and the documents in index Ie, and returns k documents with the highest scores to the data user.

Data User

When the data user wants to search with a query, s/he generates the trapdoor T for this query firstly by query encryption, and then submits the trapdoor to cloud server for query processing. We assume data users are trusted entities and the trapdoors are generated by data users themselves.

ALGORITHM

ADVANCED ENCRYPTION STANDARD (AES)

The more popular and widely adopted symmetric encryption algorithm likely to be encountered nowadays is the Advanced Encryption Standard (AES). It is found at least six times faster than triple DES.

A replacement for DES was needed as its key size was too small. With increasing computing power, it was considered vulnerable against exhaustive key search attack. Triple DES was designed to overcome this drawback but it was found slow.

The features of AES are as follows

- Symmetric key symmetric block cipher
- 128-bit data, 128/192/256-bit keys
- Stronger and faster than Triple-DES
- Provide full specification and design details
- Software implementable in C and Java

Operation of AES

AES is an iterative rather than Feistel cipher. It is based on ‘substitution–permutation network’. It comprises of a series of linked operations, some of which involve replacing inputs by specific outputs (substitutions) and others involve shuffling bits around (permutations). Interestingly, AES performs all its computations on bytes rather than bits. Hence, AES treats the 128 bits of a plaintext block as 16 bytes. These 16 bytes are arranged in four columns and four rows for processing as a matrix.

Unlike DES, the number of rounds in AES is variable and depends on the length of the key. AES uses 10 rounds for 128-bit keys, 12 rounds for 192-bit keys and 14 rounds for 256-bit keys. Each of these rounds uses a different 128-bit round key, which is calculated from the original AES key.

Decryption Process

The process of decryption of an AES ciphertext is similar to the encryption process in the reverse order.

Each round consists of the four processes conducted in the reverse order

- Add round key
- Mix columns
- Shift rows
- Byte substitution

Since sub-processes in each round are in reverse manner, unlike for a Feistel Cipher, the encryption and decryption algorithms need to be separately implemented, although they are very closely related.

AES Analysis

In present day cryptography, AES is widely adopted and supported in both hardware and software. Till date, no practical cryptanalytic attacks against AES have been discovered.

Additionally, AES has built-in flexibility of key length, which allows a degree of ‘future-proofing’ against progress in the ability to perform exhaustive key searches.

However, just as for DES, the AES security is assured only if it is correctly implemented and good key management is employed.

V. RESULTS

To implement this project, we initially start the server to connect with the database. Fig 6.1when we press on user registration option; we will get the below form for registration.user and owner both registers successfully. when they press on login option, they will get this login page. Fig 6.2 By selecting a text file from the device, owner can upload the selected file

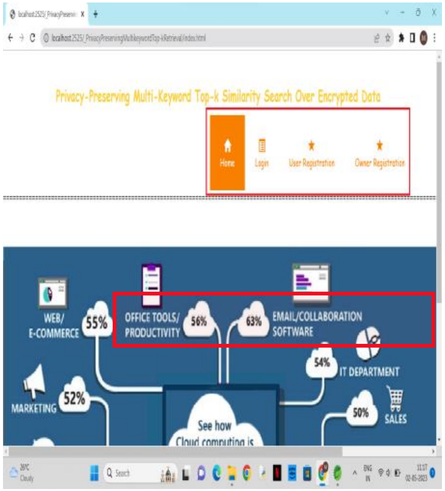
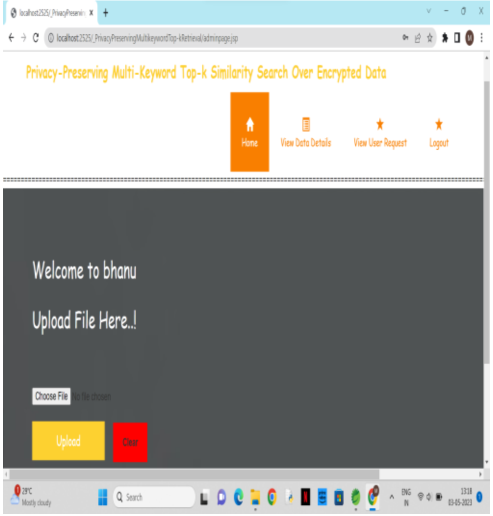

Privacy-Preserving Multi-Keyword Top-k Similarity Search Over Encrypted Data
Home
Login
User Registration
Owner Registration
WEB/ E-COMMERCE
55%
OFFICE TOOLS/ PRODUCTIVITY
56%
63%
EMAIL/COLLABORATION SOFTWARE
54%
IT DEPARTMENT
MARKETING
52%
50%
SALES
See how Cloud computing is

Fig 6.1 home page of the application

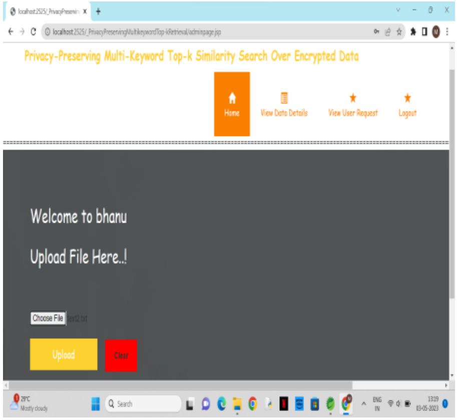

Privacy-Preserving Multi-Keyword Top-k Similarity Search Over Encrypted Data
Home
View Data Details
View User Request
Logout
Choose File
Upload
Clear

Welcome to bhanu

Upload File Here..!

Fig 6.2 upload text file

Fig 6.3 By selecting a text file from the device, owner can upload the selected fi By selecting a text file from the device, owner can upload the selected file, Fig 6.4 Once the file has been selected to upload; owner enters the title, keyword, category and expiry date for accessing the uploaded file.

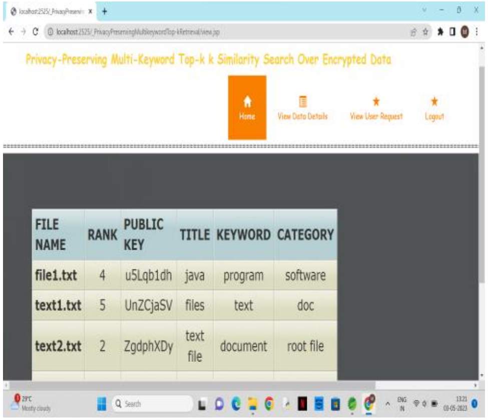

Privacy-Preserving Multi-Keyword Top-k Similarity Search Over Encrypted Data
Home
View Data Details
View User Request
Logout
Choose File
Upload
Clear

Welcome to bhanu

Upload File Here..!

,

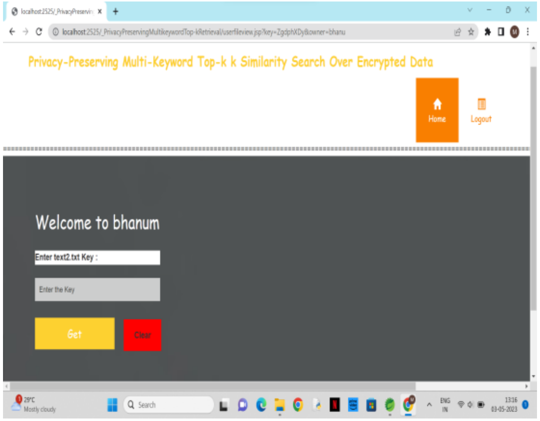
Fig 6.3 selected file to upload


Privacy-Preserving Multi-Keyword Top-k k Similarity Search Over Encrypted Data
Home
View Data Details
View User Request
Logout

FILE NAME	RANK	PUBLIC KEY	TITLE	KEYWORD	CATEGORY
file1.txt	4	u5Lqb1dh	java	program	software
text1.txt	5	UnZCjaSV	files	text	doc
text2.txt	2	ZgdphXDy	text file	document	root file

Fig 6.4 view details of uploaded files

Fig 6.5 Owner gets the request for secret key from user, Fig 6.6owner can view request id, public key, username, file name, send keys, cancel request options.

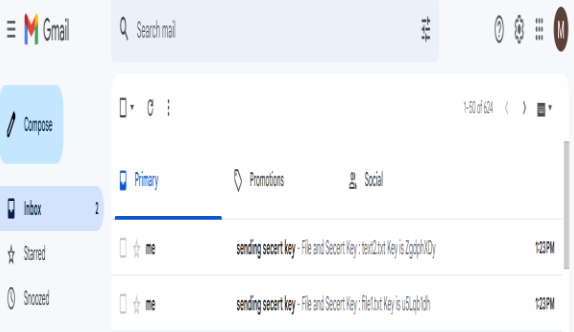

Privacy-Preserving Multi-Keyword Top-k k Similarity Search Over Encrypted Data
Home
Logout
Get
Clear

Welcome to bhanum

Enter text2.txt Key :

Enter the Key

Fig 6.5 request for secret key

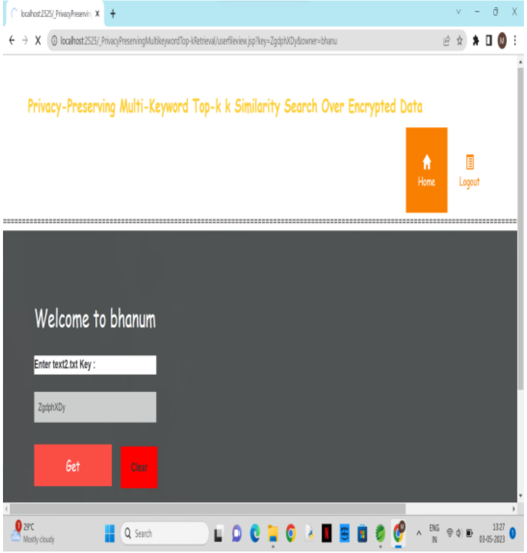


me sending secert key - File and Secert Key : text2.txt Key is ZgdphXDy 1:23PM

me sending secert key - File and Secert Key : file1.txt Key is u5Lqb1dh 1:23PM

Fig 6.6 secret key shared through mail

Fig 6.7 user can access the file by entering the secret key shared through mail by the owner.



localhost:2525/_PrivacyPreservingMultikeywordTop-kRetrieval/userfileview.jsp?key=ZgdphXDy&owner=bhanu

Privacy-Preserving Multi-Keyword Top-k k Similarity Search Over Encrypted Data

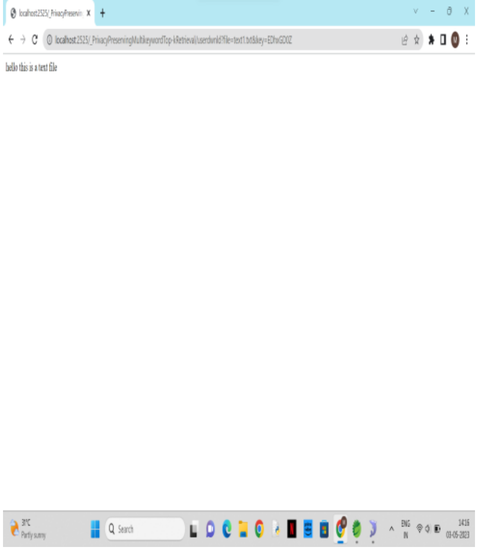
Welcome to bhanum

Enter text2.txt Key :

ZgdphXDy

Fig 6.7 user enters secret key

Fig 6.8 After entering the secret key; user can access the file which is decrypted.



localhost:2525/_PrivacyPreservingMultikeywordTop-kRetrieval/userdwnld?file=text1.txt&key=EDhvGD0Z

hello this is a text file

Fig 6.8 user views decrypted file

V. CONCLUSION

The efficiency and security of multi-keyword top-k similarity searches over encrypted data are the main goals of this research. In the beginning, we suggest the random traversal algorithm, which can execute two identical searches with different keys, the cloud server navigates over various paths on the index, and the data user obtains various results while maintaining the same high degree of query accuracy. Next, we develop the group multi-keyword top-k search strategy, which separates the dictionary into several groups and only needs to keep the top-k documents of each word group while generating an index, in order to increase the search efficiency. Next, we use the random traversal approach to obtain the RGMTS in order to safeguard the query unlink capability. This can make it more difficult for cloud servers to launch linkage attacks against two identical queries. Finally, the results demonstrate that our methods outperform state-of-the-art ones in terms of efficiency and security.

VI. FUTURE SCOPE

In the future, we plan to investigate how to support additional multi-keyword semantics (such weighted queries over encrypted data, rank order integrity checks in search results, and privacy assurances in a more robust threat model). Data users can be confident in the validity of the provided search result since we make the entire search process verifiable. To enable effective search on massive databases, we create searchable encryption that is very scalable.

VIII. REFERENCES

1. N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, “Privacy-preserving multi-keyword ranked search over encrypted cloud data,” IEEE Transactions on Parallel and Distributed Systems, vol. 25, no. 1, pp. 222–233, 2014.
2. Z. Xia, X. Wang, X. Sun, and Q. Wang, “A secure and dynamic multi-keyword ranked search scheme over encrypted cloud data,” IEEE Transactions on Parallel and Distributed Systems, vol. 27, no. 2, pp. 340–352, 2016.
3. J. Tang, Y. Cui, Q. Li, K. Ren, J. Liu, and R. Buyya, “Ensuring security and privacy preservation for cloud data services,” ACM Computing Surveys, 2016.
4. D. Boneh, G. Di Crescenzo, R. Ostrovsky, and G. Persiano, “Public key encryption with keyword search,” in Advances in CryptologyEurocrypt 2004. Springer, 2004, pp. 506–522.
5. D. X. Song, D. Wagner, and A. Perrig, “Practical techniques for searches on

encrypted data," in Security and Privacy, 2000. SP 2000. Proceedings. 2000 IEEE Symposium on, 2000, pp. 44–55.

6. Y.-C. Chang and M. Mitzenmacher, "Privacy preserving keyword searches on remote encrypted data," in Applied Cryptography and Network Security. Springer, 2005, pp. 442–455.
7. Y. H. Hwang and P. J. Lee, "Public key encryption with conjunctive keyword search and its extension to a multi-user system," in Pairing-Based Cryptography–Pairing. Springer, 2007, pp. 2–22.
8. W. Sun, B. Wang, N. Cao, M. Li, W. Lou, Y. T. Hou, and H. Li, "Privacy-preserving multi-keyword text search in the cloud supporting similarity-based ranking," in Proceedings of the 8th ACM SIGSAC Symposium on Information, ser. ASIA CCS '13. ACM, 2013, pp. 71–82.
9. N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, "Privacy-preserving multi-keyword ranked search over encrypted cloud data," IEEE Transactions on Parallel and Distributed Systems, vol. 25, no. 1, pp. 222–233, 2014.
10. [10] P. Golle, J. Staddon, and B. Waters, "Secure conjunctive keyword search over encrypted data," in Applied Cryptography and Network Security. Springer, 2004, pp. 31–45.
11. L. Ballard, S. Kamara, and F. Monrose, "Achieving efficient conjunctive keyword searches over encrypted data," in Information and Communications Security. Springer, 2005, pp. 414–426.
12. J. Baek, R. Safavi-Naini, and W. Susilo, "Public key encryption with keyword search revisited," in Computational Science and Its Applications. Springer, 2008, pp. 1249–1259.
13. D. J. Park, K. Kim, and P. J. Lee, "Public key encryption with conjunctive field keyword search," in Information security applications. Springer, 2004, pp. 73–86.
14. C. Wang, N. Cao, J. Li, K. Ren, and W. Lou, "Secure ranked keyword search over encrypted cloud data," in Distributed Computing Systems (ICDCS), IEEE 30th International Conference on, 2010, pp. 253–262.
15. B. Wang, S. Yu, W. Lou, and Y. T. Hou, "Privacy-preserving multi-keyword fuzzy search over encrypted data in the cloud," in INFOCOM, 2014 Proceedings IEEE, 2014, pp. 2112–2120.
16. M. Kuzu, M. S. Islam, and M. Kantarcioglu, "Efficient similarity search over encrypted data," in Data Engineering (ICDE), 2012 IEEE 28th International Conference on, 2012, pp. 1156–1167.
17. Q. Lv, W. Josephson, Z. Wang, M. Charikar, and K. Li, "Multiprobe lsh: Efficient indexing for high-dimensional similarity search," in Proceedings of the 33rd International Conference on Very Large Data Bases. VLDB Endowment, 2007, pp. 950–961.
18. X. Yuan, H. Cui, X. Wang, and C. Wang, "Enabling privacy assured similarity retrieval over millions of encrypted records," in European Symposium on Research in Computer Security. Springer, 2015, pp. 40–60.
19. C. Wang, N. Cao, K. Ren, and W. Lou, "Enabling secure and efficient ranked keyword search over outsourced cloud data," IEEE Transactions on Parallel and Distributed Systems, vol. 23, no. 8, pp. 1467–1479, 2012.
20. H. Li, Y. Yang, T. H. Luan, X. Liang, L. Zhou, and X. Shen, "Enabling fine-grained multi-keyword search supporting clasub-dictionaries over encrypted cloud data," IEEE Transactions on Dependable and Secure Computing, vol. 13, no. 3, pp. 312–325, MayJune 2016.
21. I. H. Witten, A. Moffat, and T. C. Bell, Managing gigabytes: compressing and indexing documents and images. Morgan Kaufmann, 1999.
22. B. Yao, F. Li, and X. Xiao, "Secure nearest neighbor revisited," in Data Engineering (ICDE), 2013 IEEE 29th International Conference on, 2013, pp. 733–744.
23. "Keyword and search engines statistics,"2016.[Online].Available :http://www. keyword-iscovery.com/keyword-stats.html