

# Multi-keyword Ranked Search over Encrypted Cloud Data

Swati Barsagade

*Department of Information Technology  
AISSMS IOIT, Pune, Maharashtra, India*

Nilima Bhujbal

*Department of Information Technology  
AISSMS IOIT, Pune, Maharashtra, India*

Kalika Kambale

*Department of Information Technology  
AISSMS IOIT, Pune, Maharashtra, India*

Swati Khilare

*Department of Information Technology  
AISSMS IOIT, Pune, Maharashtra, India  
Email- swati.khilare3@gmail.com*

**Abstract-** Multi-keyword ranked search over encrypted cloud data (MRSE) while preserving strict system-wise privacy in cloud computing paradigm. Among various multi-keyword semantics, we choose the efficient principle of “coordinate matching”, i.e., as many matches as possible, to capture the similarity between search query and data documents. Specifically, we use “inner product similarity”, i.e., the number of query keywords appearing in a document, to quantitatively evaluate the similarity of that document to the search query in “coordinate matching” principle. However, directly outsourcing data vector or query vector will violate index privacy or search privacy. To meet the challenge of supporting such multi-keyword semantic without privacy breaches, we propose a basic MRSE scheme using secure inner product computation, which is adapted from a secure  $k$ -nearest neighbor ( $kNN$ ) technique, and then improve it step by step to achieve various privacy requirements.

**Keywords –** Multi-keyword, Co-ordinate matching, Ranked Search

## I. INTRODUCTION

Cloud computing is the long dreamed vision of computing as a utility, where cloud customers can remotely store their data into the cloud so as to enjoy the on-demand high quality applications and services from a shared pool of configurable computing resources [1]. To protect data privacy and combat unsolicited accesses in cloud and beyond, sensitive data, e.g., emails, personal health records, photo albums, tax documents, financial transactions, etc., may have to be encrypted by data owners before outsourcing to commercial public cloud [2]; this, however, obsoletes the traditional data utilization service based on plaintext keyword search. The trivial solution of downloading all the data and decrypting locally is clearly impractical, due to the huge amount of bandwidth cost in cloud scale systems..

On the one hand, to meet the effective data retrieval need, large amount of documents demand cloud server to perform result relevance ranking, instead of returning undifferentiated result. Such ranked search system enables data users to find the most relevant information quickly, rather than burdensomely sorting through every match in the content collection [3]. To improve search result accuracy as well as enhance user searching experience, it is also crucial for such ranking system to support multiple keywords search, as single keyword search often yields far too coarse result. As a common practice indicated by today’s web search engines (e.g., Google search), data users may tend to provide a set of keywords instead of only one as the indicator of their search interest to retrieve the most relevant data. And each keyword in the search request is able to help narrow down the search result further. “Coordinate matching” [4], i.e., as many matches as possible, is an efficient principle among such multi-keyword

semantics to refine the result relevance, and has been widely used in the plaintext information retrieval (IR) community.

In this paper, for the first time, we define and solve the problem of multi-keyword ranked search over encrypted cloud data (MRSE) while preserving strict system-wise privacy in cloud computing paradigm[7]. Among various multi-keyword semantics, we choose the efficient principle of “coordinate matching”, i.e., as many matches as possible, to capture the similarity between search query and data documents. Specifically, we use “inner product similarity” [4], i.e., the number of query keywords appearing in a document, to quantitatively evaluate the similarity of that document to the search query in “coordinate matching” principle. During index construction, each document is associated with a binary vector as a sub index where each bit represents whether corresponding keyword is contained in the document. The search query is also described as a binary vector where each bit means whether corresponding keyword appears in this search request, so the similarity could be exactly measured by inner product of query vector with data vector. However, directly outsourcing data vector or query vector will violate index privacy or search privacy. To meet the challenge of supporting such multikeyword semantic without privacy breaches, we propose a basic MRSE scheme using secure inner product computation, which is adapted from a secure  $k$ -nearest neighbor ( $kNN$ ) technique [4], and then improve it step by step to achieve various privacy requirements in two levels of threat models. Our contributions are summarized as follows,

- 1) For the first time, we explore the problem of multikeyword ranked search over encrypted cloud data, and establish a set of strict privacy requirements for such a secure cloud data utilization system to become a reality.
- 2) We propose two MRSE schemes following the principle of “coordinate matching” while meeting different privacy requirements in two levels of threat models.
- 3) Thorough analysis investigating privacy and efficiency guarantees of proposed schemes is given, and experiment on the real-world dataset further show proposed schemes indeed introduce low overhead on computation and communication.

## II. PROPOSED ALGORITHM

### A. RSA –

RSA is an algorithm for public-key cryptography that is based on the presumed difficulty of factoring large integers, the factoring problem. RSA stands for Ron Rivest, Adi Shamir and Leonard Adleman, who first publicly described it in 1977. Clifford Cocks, an English mathematician, had developed an equivalent system in 1973, but it was classified until 1997. A user of RSA creates and then publishes the product of two large prime numbers, along with an auxiliary value, as their public key. The prime factors must be kept secret. Anyone can use the public key to encrypt a message, but with currently published methods, if the public key is large enough, only someone with knowledge of the prime factors can feasibly decode the message. Whether breaking RSA encryption is as hard as factoring is an open question known as the RSA problem.

### B. $K$ -Nearest Neighbor –

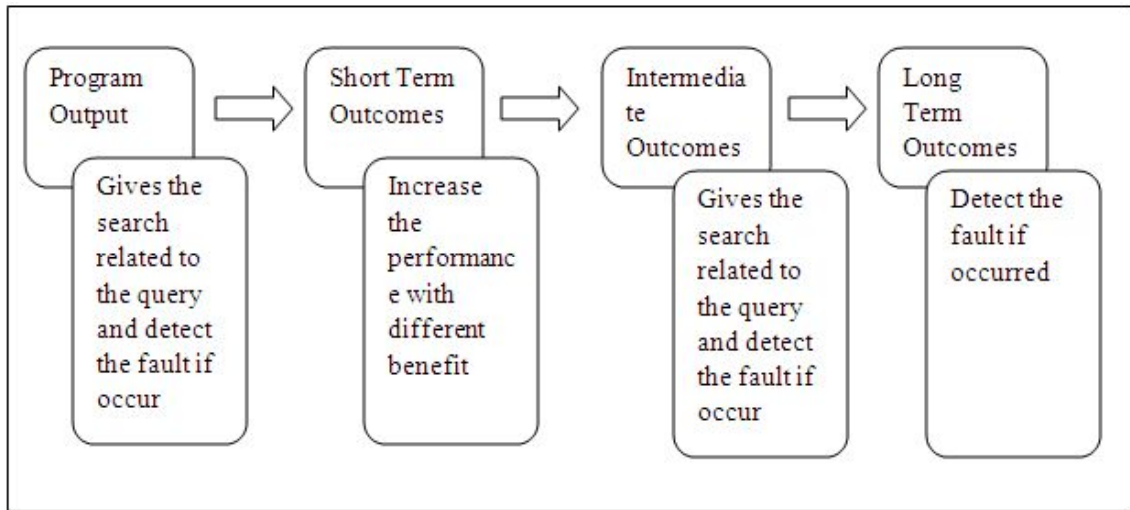
$K$ -nearest neighbor search identifies the top  $k$  nearest neighbors to the query. This technique is commonly used in predictive analytics to estimate or classify a point based on the consensus of its neighbors.  $K$ -nearest neighbor graphs are graphs in which every point is connected to its  $k$  nearest neighbors.

## III. EXPERIMENT AND RESULT

Following are steps which helps the performance measurement of the system-

### Step 1- Identify Program Outputs and Outcomes

Before writing performance measures, grantees should define their organization’s important program outputs and outcomes, paying particular attention to the causal linkages between the short-, intermediate-, and long-term outcomes. The following diagram displays program outputs and a series of related outcomes for a program intended to encourage changes in state policies on school choice. The diagram establishes the series of causes and effects that are expected for a program (i.e. what will happen first, as a result this will happen next, and so forth).



**Step 2- Identify Measurement Strategies**

Once the important outputs and outcomes are identified, grantees should identify how each can be measured. In some cases, measurement of outcomes may be difficult (e.g., organization staff may not have access to certain groups targeted for change). Although the outcome may still be relevant, performance measures cannot be easily developed. When this happens, grantees should attempt to find proxy measures or other indicators to confirm that intended outcomes of a program have occurred.

The following list identifies potential measurement strategies for each of the outputs and outcomes included in the diagram above.

Output: Gives the result with respect to query with fault tolerance manager.

Measurement strategy: Program log file and uploads the number of files required for search.

Short-term outcome: Increased performance of output.

Measurement strategy: Different decrypted files, log files, Search query.

Intermediate outcome: Increased policymaker commitment to regional economic development.

Measurement strategy: decrypted files, log files for fault detection, Search query.

Long-term outcome: Fault detection and recovery.

**Step 3- Identify Quantitative Targets for Each Output/Outcome of Interest**

Once the important outputs and outcomes and their measurement strategies are identified, grantees need to determine HOW MUCH of a particular accomplishment (for output measures) or change (for outcome measures) will constitute success. Targets should be ambitious, but achievable. The merit of a program is not always judged by the program’s ability to meet each and every target, but the extent to which progress is made towards the proposed targets. The following table includes targets for each of the program outputs and outcomes identified above.

Table 1-Experiment Result

Output/Outcomes	Target
Gives the best search related to the query and detect the fault if occur	Give atleast one result of relevant query
Increase the performance with different benefit	50% of best matching of result and fault detection
Gives the search related to the query and detect the fault if occur	Gives a good result with decryption
Detect the fault if occurred	Perform FTM with the its best

## IV. CONCLUSION

The System “Multi-keyword Ranked Search with Fault Tolerance” Manager has been implemented which work on encrypted cloud data, which provide users many specialized services so that the system can satisfied all special requirements of users. After searching the data the result has been return in the decrypted document. In Fault Tolerance, Log file describes the logged in details of user or any other person who is trying to access the account. System which gives alert sending message (SMS) from Mobile Server on user’s mobile. It describes which file property has been updated or deleted by hacker. But the best part is only data which is visible to hacker on cloud is updated, original data is not updated. So when user will get alert user could again upload data on cloud and the original data is secured and protected from hacker. Our proposed schemes introduce low overhead on both computation and communication.

## REFERENCES

- [1] L. M. Vaquero, L. Rodero-Merino, J. Caceres, and M. Lindner, “A break in the clouds: towards a cloud definition,” *ACM SIGCOMM Comput. Commun. Rev.*, vol. 39, no. 1, pp. 50–55, 2009.
- [2] S. Kamara and K. Lauter, “Cryptographic cloud storage,” in *RLCPS, January 2010, LNCS. Springer, Heidelberg*.
- [3] A. Singhal, “Modern information retrieval: A brief overview,” *IEEE Data Engineering Bulletin*, vol. 24, no. 4, pp. 35–43, 2001.
- [4] I. H. Witten, A. Moffat, and T. C. Bell, “Managing gigabytes: Compressing and indexing documents and images,” Morgan Kaufmann Publishing, San Francisco, May 1999.
- [5] D. Song, D. Wagner, and A. Perrig, “Practical techniques for searches on encrypted data,” in *Proc. of S&P*, 2000.
- [6] E.-J. Goh, “Secure indexes,” Cryptology ePrint Archive, 2003, <http://eprint.iacr.org/2003/216>. searches on remote encrypted data,” in *Proc. of ACNS*, 2005.
- [7] Y.-C. Chang and M. Mitzenmacher, “Privacy preserving keyword searches on remote encrypted data,” in *Proc. of ACNS*, 2005.
- [8] R. Curtmola, J. A. Garay, S. Kamara, and R. Ostrovsky, “Searchable symmetric encryption: improved definitions and efficient constructions,” in *Proc. of ACM CCS*, 2006.
- [9] D. Boneh, G. D. Crescenzo, R. Ostrovsky, and G. Persiano, “Public key encryption with keyword search,” in *Proc. of EUROCRYPT*, 2004.
- [10] M. Bellare, A. Boldyreva, and A. O’Neill, “Deterministic and efficiently searchable encryption,” in *Proc. of CRYPTO*, 2007.