IMPROVING FUZZY KEYWORD SEARCH OVER ENCRYPTED CLOUD STORED DATA

Dissertation submitted in fulfilment of the requirements for the Degree of

MASTER OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

By

BRAHM DATT

11600755

Supervisor

Ms. Sukhbir Kaur

School of Computer Science and Engineering

Lovely Professional University

Phagwara, Punjab (India)

August – December 2017

PAC FORM

LOVELY PROFESSIONAL UNIVERSITY
Transforming Education, Transforming India

TOPIC APPROVAL PERFORMA

School of Computer Science and Engineering

Program : P172::M.Tech. (Computer Science and Engineering) [Full Time]

COURSE CODE : CSE548 **REGULAR/BACKLOG :** Regular **GROUP NUMBER :** CSERGD0347

Supervisor Name : Sukhbir Kaur **UID :** 18571 **Designation :** Assistant Professor

Qualification : M.E. **Research Experience :** 4.5 years

SR.NO.	NAME OF STUDENT	REGISTRATION NO	BATCH	SECTION	CONTACT NUMBER
1	Brahm Datt	11600755	2016	K1637	8699613828

SPECIALIZATION AREA : Database Systems **Supervisor Signature:**

PROPOSED TOPIC : Enhanced Fuzzy Keyword Search Over Encrypted Cloud Stored Data

Qualitative Assessment of Proposed Topic by PAC		
Sr.No.	**Parameter**	**Rating (out of 10)**
1	Project Novelty: Potential of the project to create new knowledge	6.75
2	Project Feasibility: Project can be timely carried out in-house with low-cost and available resources in the University by the students.	6.75
3	Project Academic Inputs: Project topic is relevant and makes extensive use of academic inputs in UG program and serves as a culminating effort for core study area of the degree program.	6.75
4	Project Supervision: Project supervisor's is technically competent to guide students, resolve any issues, and impart necessary skills.	7.25
5	Social Applicability: Project work intends to solve a practical problem.	6.50
6	Future Scope: Project has potential to become basis of future research work, publication or patent.	7.00

PAC Committee Members		
PAC Member 1 Name: Kewal Krishan	UID: 11179	Recommended (Y/N): Yes
PAC Member 2 Name: Raj Karan Singh	UID: 14307	Recommended (Y/N): NA
PAC Member 3 Name: Sawal Tandon	UID: 14770	Recommended (Y/N): NA
PAC Member 4 Name: Dr. Pooja Gupta	UID: 19580	Recommended (Y/N): Yes
PAC Member 5 Name: Kamlesh Lakhwani	UID: 20980	Recommended (Y/N): Yes
PAC Member 6 Name: Dr.Priyanka Chawla	UID: 22046	Recommended (Y/N): NA
DAA Nominee Name: Kuldeep Kumar Kushwaha	UID: 17118	Recommended (Y/N): Yes

Final Topic Approved by PAC: Enhanced Fuzzy Keyword Search Over Encrypted Cloud Stored Data

Overall Remarks: Approved

PAC CHAIRPERSON Name: 11024::Amandeep Nagpal **Approval Date:** 04 Nov 2017

11/29/2017 2:06:25 PM

ABSTRACT

In recent year People have become more privacy conscious from developer as well as consumer perspective. People have started realizing keeping one's data secure and private not only from uncertain vulnerability attacks, security breaches but from unauthorized access like from vendors itself. Also security require constant effort, one simply can't rely on just some top notch antivirus or firewall software's. One has to keep up the security in accordance with the current pace of advancement in technology. So, best solution is to encrypt the data on user end before sending it to the any third party storage. Now, in this scenario what happen is performing regular operations like for example search is not easy as one has to decrypt all data before searching any specified keyword. This can make some user frustrate as it will take time additionally one has to specify the key for decrypting the whole table or data in the storage.

In this research, we are exploring a technique that will help user perform search over encrypted data. Also the main attraction of this technique is those users need not to specify the exact keyword to search over encrypted data. User will be able to perform wild card based searches over encrypted data and retrieve the required file instead of decrypting the whole files uploaded on the cloud.

ACKNOWLEDGEMENT

I want to display my profound appreciation to concerned individuals who helped me in this learning process.

I want to thank my tutor Ms. Sukhbir Kaur, who in this venture helped me as an educator on every single stride from the very first moment to the time of Pre-Dissertation. She additionally helped me in seeking and selecting great material to help in my research. So, I can do my work subjectively. Her proposals and helping hand dependably help to do my work smoothly and effectively.

DECLARATION

I hereby declare that the pre-dissertation proposal entitled "IMPROVING FUZZY KEYWORD SEARCH OVER ENCRYPTED CLOUD STORED DATA " submitted for the M.Tech Degree is entirely my original work and all ideas and references have been duly acknowledged. It does not contain any work for the award of any other degree or diploma.

I understand that the work presented herewith is in direct compliance with Lovely Professional University's Policy on plagiarism, intellectual property rights, and highest standards of moral and ethical conduct. Therefore, to the best of my knowledge, the content of this dissertation represents authentic and honest research effort conducted, in its entirety, by me. I am fully responsible for the contents of my dissertation work.

Sign……..

Brahm Datt

Reg.No…11600755……

SUPERVISOR'S CERTIFICATE

This is to certify that the work reported in the M.Tech Dissertation proposal entitled "**IMPROVING FUZZY KEYWORD SEARCH OVER ENCRYPTED CLOUD STORED DATA"**, submitted by **Brahm Datt** at **Lovely Professional University, Phagwara, India** is a bonafide record of his original work carried out under my supervision. This work has not been submitted elsewhere for any other degree.

Signature of Supervisor

Ms. Sukhbir Kaur

Date:

Counter Signed by:

1) **Concerned HOD:**
   HoD's Signature: _____

   HoD Name: _____

   Date: _____

2) **Neutral Examiners:**

   External Examiner

   Signature: _____

   Name: _____

   Affiliation: _____

   Date: _____

   Internal Examiner

   Signature: _____

   Name: _____

   Date: _____

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1. INTRODUCTION

In this age of digitization where everything and everyone either use, create and update digital data daily or at least has some digital presence in fact or identification or multimedia form. Yet the size of this data is growing twice per year. There is human generated data as well as machine generated data is encountering a general ten times speedier development rate than conventional business information, and machine information is expanding significantly more quickly at 50x the development rate. Now in this area of vast data cloud computing is a crucial feature.

1.1 Cloud Computing:

Cloud computing in twenty first century have very appealing advantages for organizations and end clients. Five of the primary advantages of cloud computing are:

A. Self-Benefit Provisioning: End clients can turn up figure assets for a workload on request. This kills the conventional requirement for it directors to arrange and oversee figure assets.

B. Flexibility: companies can scale up as registering needs increment and scale down again as requests diminish. This wipes out the requirement for monstrous interests in neighborhood foundation, which might possibly stay dynamic.

C. Pay Per Utilize: compute assets are measured at a granular level, empowering clients to pay just for the assets and workloads they utilize.

D. Workload Strength: cloud specialist organizations frequently execute repetitive assets to guarantee strong capacity and to keep clients' critical workloads running - regularly over different worldwide locales.

E. Relocation Adaptability: organizations can move certain workloads to or from the cloud - or to various cloud stages - as wanted or naturally for better cost reserve funds or to utilize new administrations as they develop.

With ever increasing data and need for storing this very large amount of data came the economical technology called cloud with some additional features like always available, on demand services and flexibility of storage. But storing one’s data on cloud in some other hands one simply cannot sit back and relax relying totally on their security standard. Because as the technology is changing the threat to those technologies are growing and advancing in parallel. So, it comes down to the user that they should also take some point of responsibility to make sure that their data stays safe even after cloud get compromised. So, what user can do to secure their data before outsourcing it to the cloud is they encrypt the data. This way user can secure their data from unauthorized access and other security vulnerability that may occur on data while in the cloud. But this process of encryption by the user, leads to the overhead of information access. Additionally, in distributed computing, the vendors share their stored information with vast number of clients because

of which protection of the information is not guaranteed. In this manner it is required that each individual ought to recover particular information records which they are searching for inside a session. To apply this kind of framework we need to manage seek using catchphrase or using keywords that recover the required records as opposed to recovering all the encoded documents. While plaintext seek situations, like Google, the search word method is utilized which enables clients to specifically recover the required files. Encrypted information confines client's capacity to utilize the catchphrase seek procedure and subsequently makes the plaintext look strategies to be of no utilization for cloud computing.

In the scheme proposed in[1][2], a technique called fuzzy multi keyword search over cloud data or outsourced data. By utilizing fuzzy multi keyword search the ease of use of search over encrypted data is upgraded. Clients can seek their content with conceivable values and get the desired outcome but in situations where correct catchphrase finding comes up short. This disappointment of correct keyword could be a direct result of a few spelling or typos. This is the point where fuzzy multi keyword search shows its beauty and give wanted outcomes to the client. The technique to be enhanced uses fuzzy logic system which can reduce variable values to lie in between 0 and 1.

1.2 Fuzzy Logic

Fuzzy logic is a way to deal with processing in light of “degrees of truth” as opposed to the standard thing “true or false” (1 or 0) Boolean rationale on which the computer works.

The possibility of fuzzy logic was first describe in class by Dr. Lotfi zadeh of the University of California at Berkeley in the 1960s. Dr. Zadeh was trying to process natural language. Natural language (like most different exercises in life) is not effortlessly converted into the terms of 0 and 1.so, he explored the fuzzy logic as an expected solution for his problem.

Fuzzy logic incorporates 0 and 1 as extraordinary instances of truth. Additionally incorporates the different conditions of truth in the middle, as for instance, the consequence of a correlation between two things could be not “tall” or “short” but rather some value like .05” tallness.

Fuzzy logic appears to be nearer to the way our brains work. We total information and shape various halfway truths which we total further into higher truths. A comparative sort of process is utilized as a part of neural systems, master frameworks and other computerized reasoning applications. Fuzzy logic is fundamental to the improvement of human-like capacities for ai, here and there alluded to as counterfeit general insight: the portrayal of summed up human psychological capacities in programming so that, confronted with a new errand, the ai framework could discover an answer.

1.3 Fuzzy sets

Fuzzy sets, enable components to be halfway in a set. Every component is given a level of participation in a set. This enrollment esteem can extend from 0 (not a component of the set) to 1 (an individual from the set). Unmistakably in the event that one just permitted the outrageous enrollment estimations of 0 and 1, this would really be equivalent to fresh sets. A participation capacity is the connection between the estimations of a component and its level of enrollment in a set. In this the sets (or classes) are defined numbers that are high negative, negative medium, negative little, close to zero, positive little, positive medium, and positive huge.

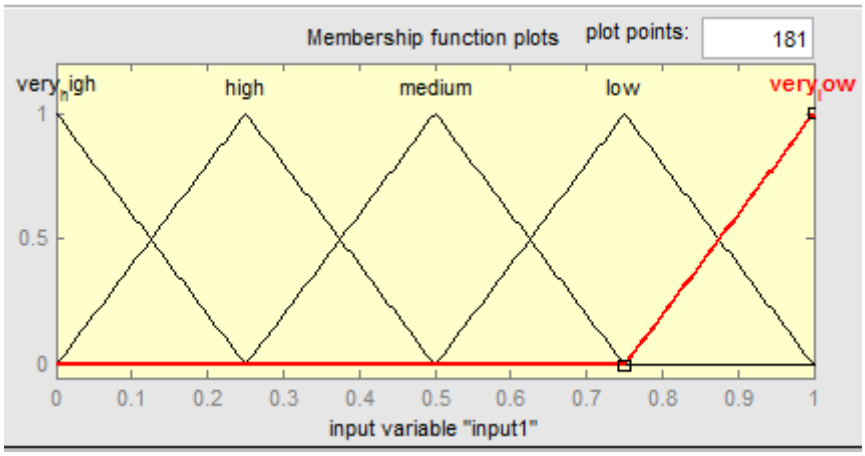

Figure 1-1 Fuzzy Set With Very High, Low, Medium, Very Low, High Values.

Fuzzy sets are fitting for example characterization on the grounds that a given signal or example may in certainty have incomplete participation in a wide range of classes. Several applications like hand writing recognition, hand printed character recognition, and voice recognition. We will use fuzzy sets to improve the search and to correct misspelled keyword.

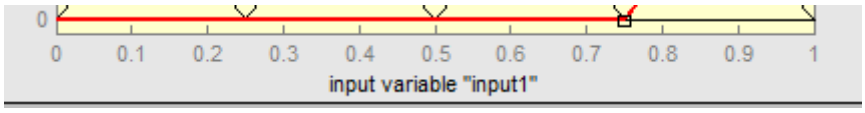

Figure 1-2 Fuzzy Crisp Input Values.

1.4 Fuzzy Keyword Search On Encrypted Data

A fuzzy keyword search as explained above use to tackle real world problem with variable outcome to lie in between 0,1 and thus reduces overhead for matching or connecting the outcome with in respond to user given inputs.

A working architecture of fuzzy keyword search taken from the same paper is shown in the figure1 as proposed in the paper [3].In simple words here in this scheme an index is created of every file that is being stored on the cloud and after that the file is encrypted and stored in the cloud .also file information which used to search particular file or document are stored in the form of n-grams. Now whenever a user want to search something like panda file a n-gram for panda file keyword is get generated and comparing it with the stored encrypted n-gram process get started . And whenever a match found corresponding file is decrypted and sent to the user. Also this technique was improved for the misspelled keyword search where the search performed by user will look something like:

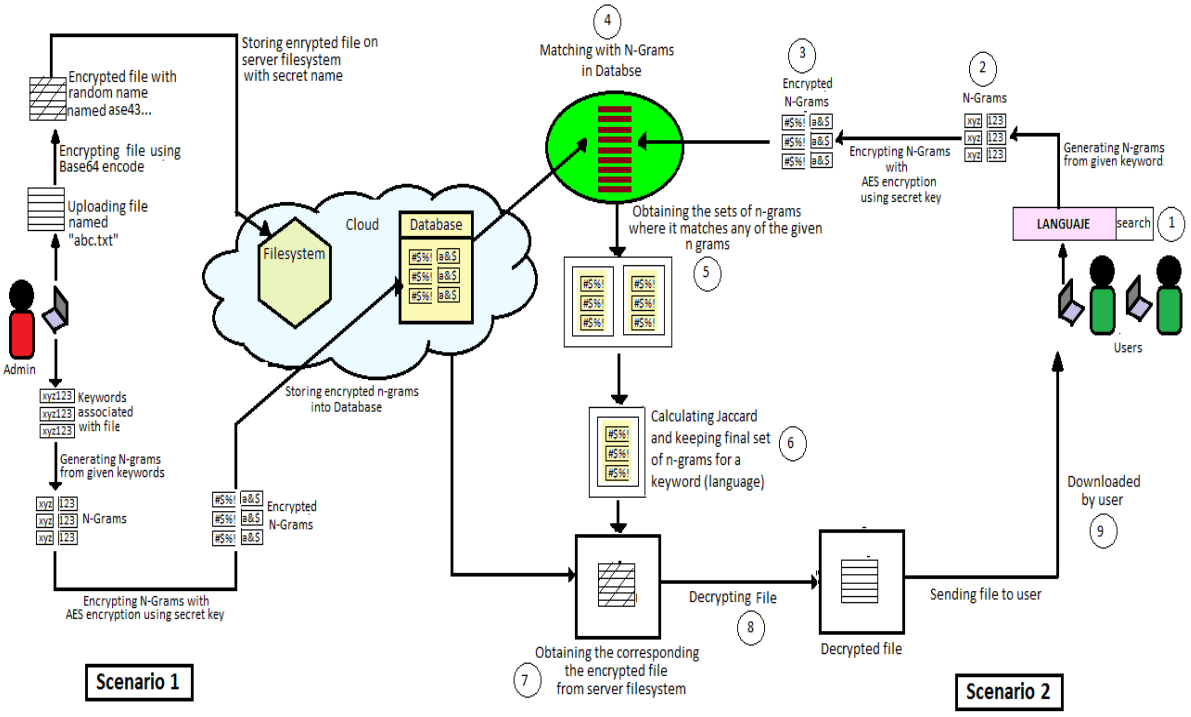

Figure 1-3 Architecture Of Fuzzy Keyword Search

- User want to search keyword panda
- And accidently or intentionally misspelled it as pnnda
- Data in the database is in encrypted form
- And then for search pnnda the keyword panda is matched and corresponding result will be shown.

Now in the proposed scheme this searching technique will get an enhancement which can be said as improvement of the fuzzy keyword search. The enhancement will be to add wild card search in the method.

2. LITERATURE SURVEY

A novel verifiable and dynamic fuzzy keyword search scheme over encrypted data in cloud computing [4]

This paper talks about overcoming the pitfalls of exact keyword searching on encrypted data. Not only that but also talks about the correctness and authenticity of the result. In this paper an uc (universal composability) technique by kurosawa et al. Called sse scheme to detect the cheating behavior of the serve as mentioned. To further explain the concept that search verification whether it's coming from trusted server or the server is compromised.

Additionally it also talks about the dynamic update of the data along with search feature.

- A wild card based search approach is followed to construct the fuzzy set.
- Also the verification of result in accordance with the server is considered for which the rsa accumulator is used to authenticate the server.
- Furthermore a vdfs* is proposed for dynamic update of file system for ease of insertion, deletion and update operation.
- The proposed vdfs is confirmed to be uc-secure.
- Uc-secure (the security of a protocol proven in a standalone setting is preserved under composition if it is secure in the universal composable framework).

Privacy preserving synonym based fuzzy multi- keyword ranked search over encrypted cloud data [5]

This paper explores the one step further development of the fuzzy search over encrypted data by introducing the multi-keyword and ranked search. Thus it makes the encrypted data search closer to the normal text like search. Also this paper explores the synonym based search which uses the file keyword generated with the extension of synonyms.

Additionally it introduces the index generation for ranked search and improve the efficiency of the index generation. This paper enhance the usual tree based dynamic tree index to be used over the fuzzy search in encrypted data.

The result time in the proposed search algorithm drastically improves the result response time. Their result shows the 90% faster search and reduction in index update.

In contrast it explores synonym based search plus ranking them for faster access.

Toward efficient multi-keyword fuzzy search accuracy improvement [6]

This paper explains and overcome some previous technique 0used for search over encrypted data. Like

- Use of bi-gram for misspelled words
- The problem with the bigram was that it is too complicated for more than one misspelled keyword.
- As it increase high and very high inconsistency for Euclidean distance calculation for matching misspelled word.
- Uni-gram based technique is used to reduce distance for Euclidean calculation. For one letter misspelled as well as other letter errors.
- Additionally use of stemming algorithm for root word search. That is search keyword with same root.
- Constructed a keyword file based on the weight of the keyword. That is the files relevant to keyword has greater chance to appear.

Individuals read writings. The writings comprise of sentences and furthermore sentences comprise of words. Individuals can comprehend phonetic structures and their implications effectively, however machines are not sufficiently fruitful on characteristic understanding. In this way, we show to machines as how to interpret this type of data. This is the primary idea; words are essential, and significant components to speak in a sentence. We know that occasionally word bunches give a bigger number of advantages than just a single word while clarifying the significance.

Here is our sentence “I read a book about the historical backdrop of India.”

The machine needs to get the importance of the sentence by isolating it into little pieces. In what manner would it be a good idea for it?

- It can respect words one by one. This is unigram; each word is a gram.

“I”, “read”, “a”, “book”, “about”, “the”, “history”, “of”, “India”

- It can respect words two at any given moment. This is bigram; every two contiguous words make a bigram.

“I read”, “read an”, “a book”, “book about”, “about the”, “the history”, “history of”, “of India”

Euclidean distance is likely harder to articulate than it is to figure. Euclidean distance used for the separation between two focuses. These focuses can be in various dimensional space and are drawn to various types of directions. In one-dimensional space, the focuses are simply on a straight number line. In two-dimensional space, the directions are given as

focuses on the x-and y-co-ordinates, and in three-dimensional space, x-, y-and z-co-ordinates are utilized.

- One-dimensional

Subtract one point on the number line from another; the request of the subtraction doesn't make a difference. For instance, one number is 5 and the other is - 6. Subtracting 5 from - 6 measures up to - 11.

Figure the total estimation of the distinction. To ascertain the outright esteem, square the number. For this case, - 11 squared equivalents 121.

Ascertain the square foundation of that number to wrap up the outright esteem. For this illustration, the square base of 121 is 11. The separation between the two focuses is 11.

- Two-dimensional

Subtract the x-and y-directions of the main point from the x-and y-directions of the second point. For instance, the directions of the primary point are (2, 4) and the directions of the second point are (- 3, 8). Subtracting the principal x-facilitate of 2 from the second x-organize of - 3 brings about - 5. Subtracting the primary y-organize of 4 from the second y-facilitate of 8 squares with 4.

Square the distinction of the x-arranges and furthermore square the distinction of the y-facilitates. For this case, the distinction of the x-organizes is - 5, and - 5 squared is 25, and the distinction of the y-arranges is 4, and 4 squared is 16.

Include the squares together, and after that take the square foundation of that whole to discover the separation. For this illustration, 25 added to 16 is 41, and the square base of 41 is 6.403. (This is the Pythagorean Theorem at work; you are finding the estimation of the hypotenuse that keeps running from the aggregate length communicated in x by the aggregate width communicated in y.)

Fuzzy keyword search over probabilistic xml data [7]

This paper explores the possibility of the fuzzy keyword search over the probabilistic xml data. The probabilistic xml data was first introduced by Andrew which consists of the label tree with ordinary and distributed nodes. In simple terms the data can be said pretty scattered that is not related or from different sources, which makes it hard to process.

What was interesting in this paper was a pruning technique is used to result refinement.

Also it was said to be the first of its kind which done study on probabilistic xml data of its own unique type for searching in document data.

Chinese-keyword fuzzy search and extraction over encrypted patent documents [8]

Chinese keyword explores the fuzzy keyword search over encrypted data for searching the keyword in Chinese words.

It explores some previous fuzzy search technique like

- Keac (automatic keyword extraction) and
- Pat –tree.
- It tells that similar keyword search is more efficient if normal matching fails. But for that to work we need to generate the similar keyword.

Mainly focuses on the improvement for Chinese auto keyword generation using pin-yin technique. But it certainly gave the new or widened the perspective for searching using similar keyword

A novel dynamic ranked fuzzy keyword search over cloud encrypted data [9]

This paper aimed to maintain the accuracy of the result in regard of the keyword inputted.

- It construct a pointer vector for ranked keyword search by using the user feedback.
- Also they show that their fuzzy set construction is efficient using mathematical model.
- Not just the one input keyword is compared for the result but also the fuzzy set create a keyword set to be matched for the result.

An effective fuzzy keyword search scheme in cloud computing [10]

Well this paper uses the bloom filter for reducing the cost search over encrypted data.

Bloom filter uses individual hash functions to map every element to a random no uniform over the range of data.

So it reduces the number of compares to be done and speed up the search process.

Privacy-preserving ranked fuzzy keyword search over encrypted cloud data[11]

This one takes the different approach for searching data over the cloud in encrypted form. But it certainly enhances the knowledge of the field. It explains that there are three approaches for searching encrypted data:

1. Index based
2. Searching encrypted data and
3. Using secret sharing

This paper proposes technique comprised by two previous technique:

- Fuzzy keyword search and
- Ranked keyword search over encrypted data

A solution which can support privacy protection and fuzzy search quickly under cloud computing environment[12]

In this paper we see that k-gram technique and its shortcoming were identified and omitted.

As in k-gram technique not the setting k value to right amount leads to drastically poor result. So, a rough fuzzy set is proposed to cover the keyword correctness.

What they actually done was that proposed to check the inputted keyword correctness for search operation to be further continue or not.

A privacy preserved full text retrieval algorithm over encrypted data for cloud storage applications[13]

This paper sets the full text retrieval very direct rules of privacy maintenance for retrieving the data from the cloud. It creates the index on an algorithm called bloom filter which helps in search. This paper proposed a technique which calculates the similarity in user keyword and encrypted document by measuring the uncertainty between query and index and result selective matching index only.

2.1 Bloom Filter:

Suppose one want to make an account on fbook, now one require to add a catchy username, and if entered name got a message, “username is as of now taken”. Then you include your favorite date along the username, but still no good fortune. Furthermore you can include your college or class roll no your birth date, still got username is as of now taken.

Be that as it may, it’s wondering that how rapidly fbook check accessibility of username via looking a huge number of username enlisted with it. There are numerous approaches to carry out this activity straight pursuit, would be terrible

Bloom filter is an information structure that can carry out this activity.

For understanding bloom algorithm, one should realize what is hashing. A hash work takes info and yields an extraordinary identifier of settled length which is utilized for proof of information.

2.2 Bloom Filter Working:

A bloom filter is a space-effective probabilistic information structure that is utilized to test whether a component is an individual from a set. For instance, checking accessibility of username is set enrollment issue, where the set is the rundown of all enlisted username. The value we pay for effectiveness is that it is probabilistic in nature that implies, there may be some false positive outcomes. False positive means, it may tell that given username is as of now taken all things considered it's most certainly not.

2.3 Intriguing Properties Of Bloom Filters:

Below are some important aspects of the Bloom filter algorithm:-

- A bloom filter of a predefined set size can handle huge number of data parts which is not possible with hash table only.
- Including a new data part never result failure. But the false positive rate increments relentlessly as components are included until the point that all bits in the bloom filter are set to 1, and soon thereafter all inquiries yield a positive outcome.
- Bloom filter never create false negative outcome, i.e., disclosing to you that a username doesn't exist when it really exists.
- Erasing components from filter isn't conceivable in light of the fact that, in the event that we erase a solitary component by clearing bits at lists produced by k hash capacities, it may cause erasure of couple of different other components which yielded the same hash function.

2.4 Working Of Bloom Filter:

Working of Bloom filter is illustrated using an array. An unfilled bloom filter with limited or predefined size can be set using array as shown in the example which will help in understanding the working of Bloom Filter:-

Table 2-1 Bloom Filter Array

0	0	0	0	0	0	0	0	0	0
1	2	3	4	5	6	7	8	9	10

We require k number of hash function to ascertain the hashes for a given information. When we need to include a thing in the bloom array, the bits at k files indexes h1(x), h2(x) … hk(x) are set, where files are computed utilizing hash function.

Illustration – suppose we need to enter “hello” in the channel, we are utilizing 3 hash function and a bit exhibit of length 10, all set to 0 at first. To start with we'll figure the hashes as following:

H1(“hello”) % 10 = 1

H2(“hello”) % 10 = 4

H3(“hello”) % 10 = 7

Note: these numbers used are figurative for example.

Presently we will set the bits at files 1, 4 and 7 to 1

Table 2-2 Bloom Filter After Insertion Array

1	0	0	1	0	0	1	0	0	0
1	2	3	4	5	6	7	8	9	10

Again we need to enter “hulk”, also we'll ascertain hashes using the hash function for each single alphabet in the word “hulk”as

H1(“hulk”) % 10 = 3

H2(“hulk”) % 10 = 5

H3(“hulk”) % 10 = 4

Set the bits at files 3, 5 and 4 to 1

Table 2-3 Bloom Filter After Second Insertion Array

1	0	1	0	1	0	1	0	0	0
1	2	3	4	5	6	7	8	9	10

Presently on the off chance that we need to check “hello” is available in filter or not. We'll do a similar procedure yet this time backward request. We ascertain individual hashes utilizing h1, h2 and h3 and check if all these files are set to 1 in the bit cluster of bloom. On the off chance that every one of the bits are set then we can state that “hello” is likely present. On the off chance that any of the bit at these files are 0 then “hello” is certainly not present.

2.5 False Positive In Bloom Filters:

The inquiry is the reason we said “most likely present”, why this vulnerability. How about we comprehend this with an illustration. Assume we need to check whether “rat” is available or not. We'll ascertain hashes utilizing h1, h2 and h3

H1(“rat”) % 10 = 1

H2(“rat”) % 10 = 3

H3(“rat”) % 10 = 7

On the off chance that we check the bloom filter, bits at these records are set to 1 yet we realize that “cat” was never added to the bloom filter. Bit at record 1 and 7 was set when we included “hello” and bit 3 was set we included “hulk”.

Table 2-4 Check Bloom Filter Array

1	0	1	0	1	0	1	0	0	0
1	2	3	4	5	6	7	8	9	10

Since bits at computed files are as of now set by some other values, bloom channel incorrectly assert that “rat” is available and producing a false positive outcome. Contingent upon the application, it could be tremendous drawback or moderately alright.

We can control the likelihood of getting a false positive by controlling the measure of the bloom channel. More space implies less false positives. On the off chance that we need diminish likelihood of false positive outcome, we need to utilize more number of hash capacities and bigger piece exhibit. This would include inertness moreover of thing and checking participation.

2.6 Jaccard's Coefficient:

Its measure of similarity and it basically calculate measure of dissimilarities of asymmetric data values which may or may not be in 0, 1 form that is digital form. The jaccard is a measure of closeness for the two arrangements of information, with a range from 0% to 100%. The higher the rate, the more related the two will be. In spite of the fact that it's anything but difficult to execute, it is greatly touchy to little sizes of groups and may give wrong outcomes, particularly with little groups or informational indexes with missing values.

The equation to discover the index is:

Jaccard index = (the number in the two sets)/ (the number in either set) * 100

2.7 Problem Statement

Our purpose is to design an efficient and fuzzy keyword search system based on cloud storage which also support wild card search. More specifically, the design tasks include:

1. Automatic fuzzy keyword extraction index generation.
2. Automatic wild card based index generation.
3. Constructing the keyword fuzzy set, which should be of small size and fast building period.
4. Keyword fuzzy search, which supports a rapid encrypted file searching using an efficient index structure.
5. System usability, which costs low level communication bandwidth and computation complexity to complete all the above tasks.
6. Data security, which prevent the leakage of sensitive files and keyword information.

3. RESEARCH WORK

3.1 Objective

Based on stored indexed keyword making user to be able to search using wild cards sets our objective is intend to build an efficient and effective fuzzy keyword search.

1. To explore new mechanism for constructing efficient keyword search scheme.
2. To build the fuzzy keyword index by the user while uploading or by the admin of data at outsourced space.
3. Also build appropriate index for wild card searching of all upload file.
4. Search for the authorized user by computing the search query and comparing the same with the pre calculated multi keyword index.
5. Be able to use wild card to search instead of whole text keywords for finding file.
6. Construct the corresponding n-grams for the user given wild card query.
7. Based on the fuzzy keyword computation result all possible encrypted file identifiers.
8. Compare time and efficiency cost for the scheme.

3.2 Methodology

Developing a practical project using php,my sql and some helping library to show the reasonability and working and other efficiency factors.

1. Build a website
2. User and admin data share environment
3. Store some data using database
4. Feature of server or cloud data upload.
5. Feature of encrypt data before uploading or after uploading.
6. Transparent user encrypted data retrieval.
7. Transparent data search process.
8. Make data index generation automatic
9. User can use wild card search for uploaded file with extension of file.

Algorithms to be worked with

1. Edit distance
2. Jaccard index
3. Bloom filter

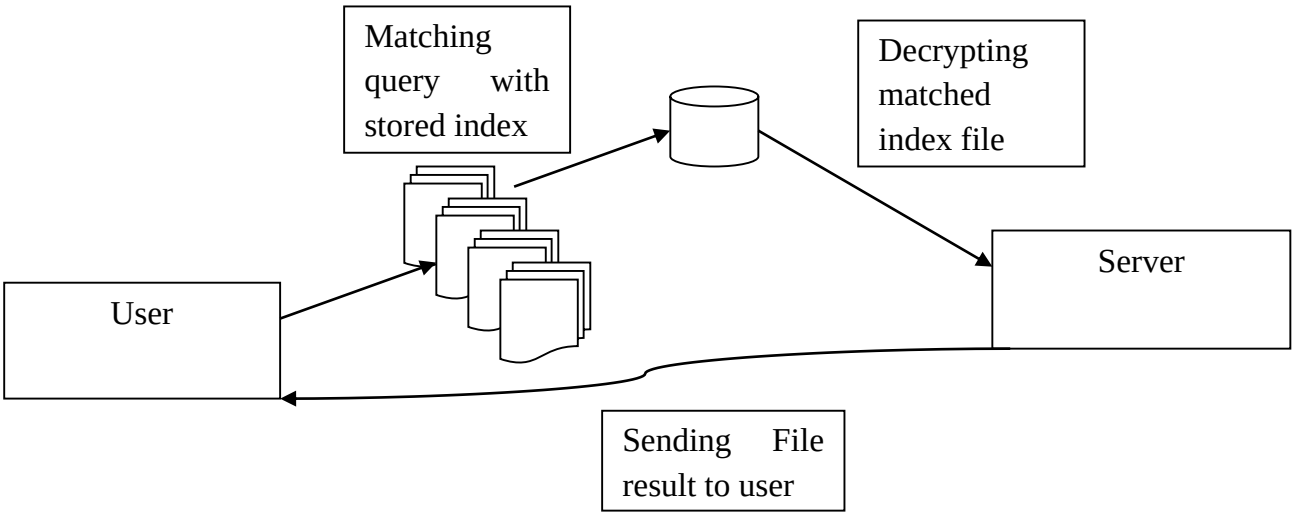

Figure 3-1 First Iterative Working Model

3.3 Design

- Design and build a user registration interface.
- Design user file upload interface.
- Whenever user upload file and encrypt it create an index of file for search
- Create wild card index for searching encrypted file
- Basic structure of the first iterative model on which the research project will be build is shown in

4. FUTURE SCOPE AND CONCLUSION

4.1 Scope

In this research an effective approach to solve the problem of fuzzy-keyword search over encrypted cloud data. The search results can be achieved when authorized cloud customers input the fuzzy matching keywords. This system also provide user to use wild card while fetching the file using search. The proposed research will analyzes the performance of our schemes in detail, including search efficiency, search accuracy, by the experiment on real-world dataset. Thus resulted scheme may prove better solution for cloud vendor as an access method for users who have largely stored encrypted data on the cloud.

4.2 Conclusion

In this research the fuzzy keyword search on encrypted technique will be enhanced which can enable users to search data using wild card which may include file search using extention name only or using one word or pair of words.

REFERENCES

[1] M. Tiwari, R. Mahajan, S. Ahuja, S. Rawat, and V. Mittal, "FUZZY KEYWORD SEARCH OVER ENCRYPTED DATA IN CLOUD COMPUTING," vol. 4, no. 6, pp. 15–20, 2016.

[2] A. Meharwade and G. A. Patil, "Efficient Keyword Search over Encrypted Cloud Data," *Phys. Procedia*, vol. 78, no. December 2015, pp. 139–145, 2016.

[3] T. Balamuralikrishna, C. Anuradha, and N. Raghavendrasai, "FUZZY KEYWORD SEARCH OVER ENCRYPTED DATA IN CLOUD COMPUTING," vol. 3, pp. 86–88, 2011.

[4] X. Zhu and G. Wang, "A Novel Verifiable and Dynamic Fuzzy Keyword Search Scheme over Encrypted Data in Cloud Computing," 2016.

[5] S. A. Mittal, "Privacy Preserving Synonym Based Fuzzy Multi- Keyword Ranked Search Over Encrypted Cloud Data," pp. 1187–1194, 2016.

[6] Z. Fu, X. Wu, C. Guan, X. Sun, and K. Ren, "Toward Efficient Multi-Keyword Fuzzy Search Accuracy Improvement," vol. 11, no. 12, pp. 2706–2716, 2016.

[7] Y. Zhao, G. Wang, and Y. Yuan, "Fuzzy Keyword Search over Probabilistic XML Data," pp. 2523–2527, 2015.

[8] W. Ding, Y. Liu, and J. Zhang, "Chinese-keyword Fuzzy Search and Extraction over Encrypted Patent Documents," 2009.

[9] W. Jie and W. Yong, "A Novel Dynamic Ranked Fuzzy Keyword Search Over Cloud Encrypted Data," 2014.

[10] H. Tuo, "An Effective Fuzzy keyword Search Scheme in Cloud Computing," vol. 1, 2013.

[11] Q. Xu, H. Shen, Y. Sang, H. Tian, I. Technology, and I. Engineering, "Privacy-Preserving Ranked Fuzzy Keyword Search over Encrypted Cloud Data," 2013.

[12] L. Xue, R. Wuling, and J. Guoxin, "A Solution Which Can Support Privacy Protection and Fuzzy Search Quickly under Cloud Computing Environment," pp. 43–46, 2014.

[13] "A privacy-preserved full text retrieval algorithm over encrypted data for cloud storage applications," *J.Parallel Distrib. Comput.*