**AN EFFICIENT SEARCH METHOD USING FEATURES TO MATCH JOINT KEYWORDS ON ENCRYPTED CLOUD DATA**

**ABSTRACT**

With the continuous improvement of the security of cloud storage, more users upload private data to the cloud. However, a large number of encrypted data using independent keywords to create indexes not only directly increase the storage overhead, but also lead to the decline of search efficiency. Therefore, this paper proposes an efficient search method using features to match joint keywords (FMJK) on encrypted cloud data. This method proposes that each *d* keywords are randomly selected from the non-duplicated keywords, which are extracted from the documents of the data owner, to generate a joint keyword, and all joint keywords form a keyword dictionary. Each joint keyword is matched with the feature of the document and the query keyword respectively, and the result obtained by the former is regarded as a dimension of the document index, while the result obtained by the latter is regarded as a dimension of the query trapdoor. Finally, the BM25 algorithm is used to calculate the inner product of the document index and the trapdoor, and then sort them and the top k results are returned. Theoretical analysis and experimental results show that the proposed method is more feasible and more effective than the compared schemes.

**EXISTING SYSTEM**

In most of the existing cipher-text sorting retrieval methods, KNN (K Nearest Neighbour) technology is used to create indexes supporting cipher-text retrieval. In the process of massive data encryption search, most of the search encryption schemes have high time complexity and large storage space, which are closely related to the encrypted key, the document index and query request dimensions. Reducing high dimensional data encryption is a solution to improve search efficiency. Some researchers try to study how to enrich the flexibility of retrieval, however they still cannot meet the retrieval requirements of a large number of data, and they cannot sort and filter useful data for authorized users. Therefore, in the face of different user needs, it is urgent to find a scheme that can not only guarantee privacy, but also improve retrieval efficiency and ensure query accuracy.

**PROPOSED SYSTEM**

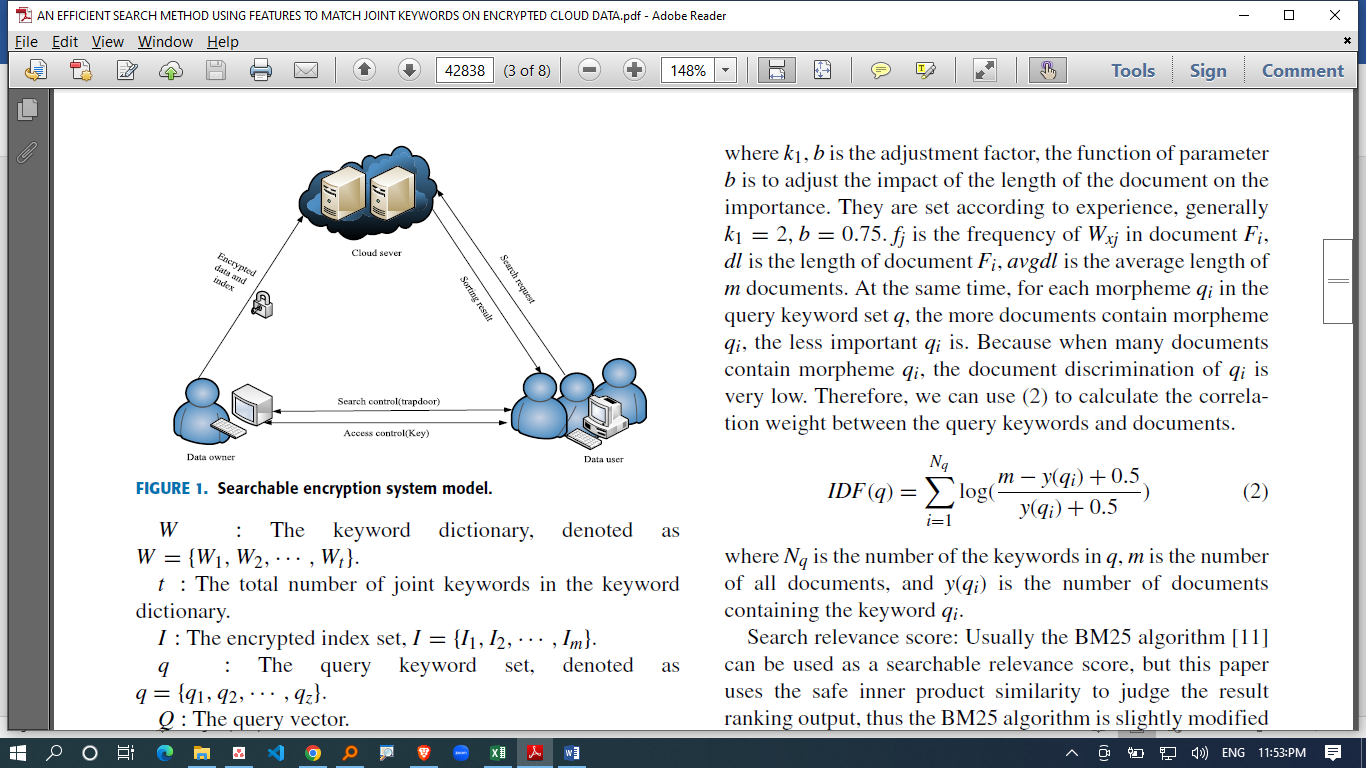
In this paper, we propose an efficient search method using features to match joint keywords (FMJK) on encrypted cloud data based on the MRSE (Privacy-Preserving Multi-Keyword Ranked Search over Encrypted Cloud Data) scheme. First, it is necessary to extract the features of each document to accurately express its theme. Then each d keywords are randomly selected from the non-duplicated keywords, which are extracted from the documents of the data owner, to generate a joint keyword, and all joint keywords form a keyword dictionary and then the features of each document are matched with the joint keywords to create an index. The authorized user enters the query keywords to match the joint keywords to create a trapdoor, and finally calculates the safe inner product of the trapdoor and the index to return the top k results. The contributions of this article are as follows:

(1) Each d randomly selected keywords form a joint keyword, which matches with the feature of the document to be mapped to one dimension of the index, which reduces the dimension size of the key, the index and the trapdoor, simplifies the matrix operation during encryption, and improves search efficiency.

(2) Through the improved BM25 algorithm to calculate the inner product of the document index and the trapdoor, which not only sorts quickly but also ensures query accuracy.

(3) The randomness of joint keywords and the encryption process of expanding and splitting ensure privacy protection. The rest of this paper is organized as follows. Section II discusses the related work of searchable technology in cloud computing environment. Section III introduces the system model attack model and research objectives of the searchable encryption process, as well as the explanation of the symbols in this paper. Section IV presents the FMJK scheme in detail. Section V analyses the FMJK scheme theoretically and experimentally.

**SYSTEM ARCHITECTURE**



**HARDWARE REQUIREMENTS**

|  |  |  |
| --- | --- | --- |
| MINIMUM (Required for Execution) | | MY SYSTEM (Development) |
| System | Pentium IV 2.2 GHz | i3 Processor 5th Gen |
| Hard Disk | 20 Gb | 500 Gb |
| Ram | 1 Gb | 4 Gb |

**SOFTWARE REQUIREMENTS**

|  |  |
| --- | --- |
| Operating System | Windows 10/11 |
| Development Software | Python 3.10 |
| Programming Language | Python |
| Domain | Cloud Computing |
| Integrated Development Environment (IDE) | Visual Studio Code |
| Front End Technologies | HTML5, CSS3, Java Script |
| Back End Technologies or Framework | Django |
| Database Language | SQL |
| Database (RDBMS) | MySQL |
| Database Software | WAMP or XAMPP Server |
| Web Server or Deployment Server | Django Application Development Server |
| Design/Modelling | Rational Rose |