Scholar Publishing Group
Machine Learning Theory and Practice
https://doi.org/10.38007/ML.2025.050108
ISSN 2790-0983 Vol. 5, Issue 1: 76-86

Compiling Declarative Privacy Policies into Runtime Enforcement for Cloud and Web Infrastructure

Chen-Wei Chang

Independent Researcher, WA 98011, USA

Keywords: Declarative Privacy Policy; Cloud infrastructure; Searchable encryption; Dynamic range query; Privacy Computing Protocol

Abstract: With the popularization of cloud computing and web infrastructure, the demand for cloud data storage and processing has surged. However, data privacy issues (such as internal theft and data leakage caused by external attacks, accounting for 25% of cloud environment threats) have a significant impact on personal, enterprise, and national security. Efficiently compiling declarative privacy policies, such as access control and data usage restrictions, into executable policies is a key challenge. This study combines searchable encryption (SSE/PEKS), secret sharing, index optimization, and computation protocol design to construct a policy compilation framework: proposes a false positive free fuzzy keyword search scheme, and combines Shamir secret sharing to achieve multi-user, multi-keyword, dynamic update, and verification functions, which is more efficient than symmetric encryption schemes and ensures 100% accuracy; Design a low-level index structure, combined with point/range intersection determination technology, to construct a dynamic multi-dimensional range search system that improves search and update efficiency while ensuring single dimensional privacy; Improve the privacy data calculation function, design protocols for mean, variance, correlation, etc., and support complex analysis needs in policy implementation; In addition, extending the discrete logarithm problem to semigroup and proposing a solution algorithm to enhance the security of cryptographic primitives. Future research will focus on the Top-k problem of fuzzy search (returning the most relevant top k results), dynamically updated forward/backward privacy protection (preventing historical policies from leaking new data or subsequent queries from accessing deleted data), and exploring the discrete logarithm problem on the half loop to improve the efficiency and security of privacy policy enforcement.

1. Introduction

The rapid development of mobile Internet, big data, supercomputing, sensor networks and brain science has pushed machine learning (ML) into an unprecedented accelerated development stage, promoting technological breakthroughs in medical, financial, autonomous driving and other fields. However, its entire life cycle (data preprocessing, model training, model reasoning) is facing increasingly complex security and privacy challenges - traditional technologies (such as differential privacy, basic security multi-party computing) are difficult to balance security, efficiency and scalability, especially in resource constrained scenarios such as cloud computing and edge devices. Although existing research proposes solutions to local problems, there are still key shortcomings: security feature extraction schemes in the data preprocessing stage may lack end-to-end privacy

protection due to high computational costs or inability to seamlessly integrate with federated learning frameworks; Distributed solutions during the model training phase (such as federated learning) or sacrificing efficiency and robustness (such as vulnerability to poisoning attacks), or excessively sacrificing performance through cryptographic protocols; In the model inference stage, there are common issues with the security protocols of Convolutional Neural Networks (CNN) and Transformer based Graph Neural Networks (GNN), such as high communication costs and insufficient support for complex operations (such as large-sized convolution kernels and nonlinear activation functions).

This article aims to construct a privacy protection framework that covers the entire lifecycle of machine learning, with a focus on addressing three core issues: designing a secure and efficient feature extraction scheme suitable for cloud assisted environments (compatible with horizontal federated learning), proposing an anti-attack gradient aggregation strategy to enhance the robustness and efficiency of distributed training, and developing a low-cost secure inference protocol for modern neural architectures such as CNN and Transformer GNN (supporting linear and nonlinear operations). Specifically, a single cloud assisted feature extraction framework called SeiFS is proposed for horizontal federated learning scenarios, which integrates obfuscation circuits, unintentional transmission, and secret sharing techniques. By balancing binary tree encoding and secure feature selection mechanisms, it achieves seamless integration with existing federated learning protocols while ensuring user data privacy; For the distributed training phase, a PEFL protocol combining additive homomorphic encryption and robust gradient aggregation strategy was designed. By automatically filtering malicious user submitted poisoning gradients and optimizing aggregation rules based on gradient sign and numerical correlation, the reliability of model training was improved. At the same time, a lightweight version of PEFLimd was introduced, which significantly reduced training latency by reducing redundant calculations and merging communication rounds. Its performance advantages were verified through convergence analysis; In the model inference stage, a dedicated secure inference framework CryptoGT for Transformer GNN was developed, which includes a homomorphic encryption convolution evaluation scheme (using a fast convolution algorithm to convert convolution operations into element wise multiplication and addition, avoiding ciphertext rotation, supporting large-sized convolution kernels and large stride lengths) and a high-order polynomial security evaluation protocol based on vector unintentional linear evaluation (VOLE) (for nonlinear functions such as Softmax, LayerNorm, GeLU, etc., complex calculations are completed through a single protocol call, significantly reducing communication complexity). All schemes have been verified through formal security proofs (semi honest models) and real dataset experiments: SeiFS achieves 1.13 times higher feature extraction efficiency than baseline federated learning in cloud scenarios; PEFL/PEFLimd reduces training time by up to 10.49 times while filtering 95% of the poisoning gradient; CryptoGT reduces communication overhead by 44.61 times compared to existing solutions in GNN inference tasks.

2. Correlation theory

2.1. Cryptography Basic Tools and Security Definitions

A finite field is an important foundational tool in fields such as mathematics, cryptography, and coding. It is defined as a field containing only a finite number of elements (also known as a Galois field), with an order of prime numbers or powers of prime numbers (such as p or p^n). Its characteristic is prime number, and for prime number p, there exists a prime field Z_p composed of modulo p integers; For the power p^n of prime numbers, a finite field $F\{p^n\}$ can be constructed by modulo an n-th irreducible polynomial in the polynomial ring $Z_p[x]$. Negligible function is used to

describe attack advantage, defined as for any positive polynomial p (•), there exists an integer N, and when the parameter A>N, the function value is less than 1/p (A), and its sum and product with the polynomial are still negligible functions. A hash function is an irreversible function that maps any length message to a fixed length digest. It has properties such as unidirectionality (a known digest cannot be used to infer a message), weak collision resistance (a known message cannot find another message with the same digest), and strong collision resistance (two different messages cannot find the same digest).

The secret sharing scheme is the core tool of distributed security protocols, consisting of access structures (authorized and unauthorized sets) and allocation schemes (secret distribution and reconstruction). Shamir's (t, n) threshold scheme is implemented through polynomial interpolation: a t-1 degree polynomial is generated during secret distribution to calculate the shares of each participant; Use t shares to reconstruct polynomials during merging to recover secrets. The Paillier (1999) and ElGamal (1985) public key cryptography is a typical homomorphic encryption scheme: the Paillier system is based on the problem of composite modulus difficulty, has additive homomorphism, selects large prime numbers p and q for key generation, calculates modulus N=pq and generator g, randomly selects r for encryption, and the ciphertext is $c=g^{m} r^{n} \bmod N^{2}$; The ElGamal system is based on the discrete logarithm difficulty problem and has multiplicative homomorphism. The key generation selects the cyclic group G and the generator g, randomly selects the private key x, the public key $y=g^{x}$, randomly selects r during encryption, and the ciphertext is ($A=g^{r}$, $B=m \cdot y^{r}$).

In terms of security definition, semantic security (IND-CPA) refers to the inability of an adversary to distinguish between two ciphertexts of the same length; Indifferentiation under keyword attack (IND-CKA) refers to the inability of an adversary to distinguish the corresponding keywords in ciphertext without knowing the trapdoor; Indifferentiation under internal keyword guessing attack (IND-IKGA) refers to the inability of an adversary to infer keyword information when they have a trapdoor; Choose security to prove protocol security through simulations of the real world and the ideal world. If the adversary's advantage in distinguishing between the two can be ignored, then the solution is secure.

2.2. Searchable encryption schemes

Searchable encryption technology is mainly divided into two categories: searchable symmetric encryption (SSE) and searchable asymmetric encryption (PEKS). PEKS is based on public key encryption algorithm and consists of three parts: data owner, data user, and cloud platform. The data owner encrypts documents and indexes with the public key and uploads them to the cloud server. The data user generates a search trapdoor through the private key, and the server tests whether the secret contains keywords and returns the result without decrypting it. Its technical implementation is widely based on various encryption foundations: early solutions based on Public Key Infrastructure (PKI) required secure channel transmission trapdoors, and subsequent research gradually eliminated this limitation; The identity based encryption (IBE) scheme treats keywords as identities and supports anonymous encryption to enhance security; Based on Attribute Encryption (ABE), fine-grained access control is achieved by associating ciphertext and private key with attributes; Based on predicate encryption (PE), search is achieved by matching predicate and ciphertext properties, supporting complex operations such as disjunction and concatenation. In addition, the Certificate Free Encryption (CLE) scheme solves the problem of key custody and supports Proxy Re Encryption (PRE), which allows the principal to delegate search permissions to the principal through re encryption without disclosing the private key. Searchable Symmetric Encryption (SSE) consists of data owners, cloud platforms, and data users: data owners encrypt data using symmetric

algorithms and generate secure indexes, users share keys to generate search trapdoors, and servers perform searches and return matching encryption results. SSE continues to expand its functionality, covering single/multiple keyword search, Boolean search (supporting "AND", "OR", and "NOT" operations), sorted search (returning results by keyword frequency or weight), semantic search (supporting content aware matching), fuzzy keyword search (handling spelling errors or uncertain keywords), and more. Some solutions also have verifiability (preventing server tampering with results) and dynamic update functionality (supporting the addition or deletion of encrypted data) to meet practical needs in different scenarios.

3. Research method

3.1. A Non False Positive Multi User Verifiable Fuzzy Search Scheme Based on Shamir Secret Sharing

The Internet of Things combines information sensing devices with networks to achieve data perception, collection, and sharing. Its large-scale deployment benefits from the explosive growth of smart devices and the development of wireless communication technology. In the 6G era, the application of the Internet of Things has expanded to fields such as medical automation, intelligent transportation, and intelligent agriculture (such as wearable devices supporting real-time remote medical monitoring, and the combination of drones and artificial intelligence to achieve agricultural automation management). To efficiently utilize massive amounts of data, IoT applications often outsource data to cloud servers, where users store, analyze, and process data through the cloud. However, the collaboration between cloud computing and the Internet of Things brings security challenges: partially trusted cloud servers and malicious eavesdroppers may lead to privacy breaches, so users need to store data in encrypted form, which creates a demand for ciphertext search. Among them, fuzzy keyword search is particularly important because users are often uncertain about the spelling of keywords. Its core includes underlying matching techniques (such as fuzzy dictionaries, locally sensitive hashing, Bloom filters, vector encoding, etc.) and privacy protection mechanisms (symmetric/asymmetric encryption). However, existing solutions still face four major challenges: first, accuracy issues. Efficient fuzzy search can easily generate "false positives" (such as when users search for "female" (fuzzy keyword "fnn * * *"), the server may return a mismatched "false"), and solutions based on local sensitive hashing (FWSR), AliuHash, and count Bloom filters (HLZ) or prime number indivisibility (LPP+) still have accuracy defects; Secondly, the adaptability to multi-user scenarios is poor. Most solutions are based on symmetric encryption and require shared keys, while the key protection capability of smart devices is limited. Frequent exchanges increase the risk of leakage and are difficult to apply to multi-user scenarios such as hospital authorized doctors searching medical data; The third issue is insufficient verifiability. Existing solutions assume that cloud servers are honest but curious, but in reality, they may return incorrect results. Some verification mechanisms, such as Merkle hash trees, homomorphic message authentication codes, and RSA accumulators, may not support fuzzy search or have high overhead; Fourthly, the support for dynamic updates is limited, and solutions that support data addition, deletion, and modification (such as ZLC+) are difficult to balance multi keyword search, verifiability, and other functions while maintaining high efficiency.

In response to the above issues, this section proposes a false positive free fuzzy keyword matching technique and designs a multi-user fuzzy keyword search system (MFKS). This system combines the Shamir secret sharing scheme, supports multi keyword fuzzy search, dynamic updates, and result verification, achieving complete accuracy while maintaining comparable efficiency to symmetric encryption schemes. The system model includes four types of entities: data providers (IoT devices/sensors, responsible for encrypting files, extracting keywords, and building secret

indexes), data users (authorized users, generating query vectors through encrypted identity tags), cloud platforms (including multiple data servers, storing encrypted files and indexes, and executing searches in parallel), and verification centers (generating system parameters, matching keywords, and verifying the correctness of results). The data flow includes: data providers uploading encrypted files and indexes (steps ② and ③), users sending query requests (steps ④ and ⑤), cloud servers performing searches and returning results (steps ⑥⑧), and verification centers verifying and returning the final files (steps ⑦⑨) (as shown in Figure 1).

Table 1. Functional Comparison of Multi-User Fuzzy Keyword Search Schemes

Scheme	Multi-Keyword	Multi-User	Verifiable	Dynamic Update	False-Positive Free
FWSR[47]	✓	✗	✗	✗	✗
HLZ[61]	✓	✗	✗	✗	✗
LPP+[92]	✓	✗	✗	✗	✗
TMW+[136]	✓	✗	✓	✗	✗
LZm+[90]	✗	✓	✓	✗	✓
ZLC+[176]	✗	✓	✗	✓	✗
MFKS System	✓	✓	✓	✓	✓

The attack model assumes that the verification center is honest, and at most t servers in the cloud platform may collude to recover keywords or send incorrect results; The data provider and user are "honest but curious" (complying with the protocol but may eavesdrop on communication), and the system allows them to eavesdrop on communication between a single participant (authentication center, user, or provider) and up to t servers. Security analysis shows that the system can resist keyword selection attacks and internal keyword guessing attacks; Experimental verification shows that it achieves complete accuracy and maintains high efficiency.

3.2. Vector and Range Determination (VSRD) Fuzzy Keyword Search Scheme

To solve the problem of false positives in fuzzy keyword search, this chapter proposes a fuzzy keyword search scheme based on vector and range determination (Vector Sum Range Decision, VSRD). This scheme achieves precise matching by encoding keywords into fixed length vectors and utilizing the range determination of vector sums, fundamentally eliminating false positives while supporting efficient multi keyword search. Existing fuzzy search schemes, such as Bloom filters and prime factorization schemes, are prone to false positives due to probabilistic matching or computational accuracy issues. The VSRD scheme solves this core problem by redesigning the keyword encoding and matching mechanism: firstly, characters are divided into three categories - exact characters (26 English letters), padding characters ("#" for fixed vector length), and wildcard characters ("*" for uncertain letters); Then randomly select integer encoding for the 26 letters, set the padding character encoding to be greater than the maximum letter encoding and less than a specific threshold, and set the wildcard encoding to be the sum of all character encodings (ensuring it matches all characters). On this basis, the index vector and the query vector are formed by taking the opposite number of keyword encoding. By calculating the sum of the two vectors (prime number P), if each term of the sum vector is either 0 (exact match) or within the filled character encoding interval (wildcard or filled character match), it is determined that the keyword matches. This method is essentially a vector subtraction operation, avoiding probabilistic matching and computational accuracy issues. The VSRD scheme includes four core algorithms: setting the

algorithm to select a large prime number P and a fixed vector length L, randomly generating 26 letter integer encoding, and defining padding characters and wildcard encoding; The index extraction algorithm fills the index keywords (including exact characters and padding characters) to length L, encodes them, and takes the opposite number to generate an index vector; The query extraction algorithm fills the query keywords (including exact characters and wildcard characters) to length L, encodes them, and generates a query vector; The matching algorithm calculates the sum (modulus P) of the index vector and the query vector. If each item of the sum vector satisfies the matching condition (0 or fills the character interval), it returns a match (1), otherwise it returns a mismatch (0).

3.3. Multi user fuzzy keyword search system

The multi-user fuzzy keyword search system (MFKS) proposed in this chapter combines the Vector and Range Decision (VSRD) fuzzy search scheme and Shamir secret sharing scheme to construct an efficient and secure system that supports multi-user, multi-keyword search, dynamic updates, and result verification. The system model includes four types of entities: data providers (responsible for encrypting files, extracting keywords, and building indexes), data users (generating query vectors through encrypted identity tags), cloud platforms (including multiple data servers that store encrypted files and indexes and perform searches in parallel), and verification centers (generating system parameters, matching keywords, and verifying the correctness of results), covering six core algorithms: initialization, index construction, query construction, search, verification, and update.

During the initialization phase, the verification center runs the VSRD setting algorithm to generate system parameters (such as large prime numbers, character encoding functions, etc.), and distributes secret parameters to data providers and users through a secure channel. In the index construction stage, the data provider uses the VSRD index extraction algorithm to encode keywords into index vectors, which are then split into multiple secret shares through Shamir secret sharing and distributed to the corresponding cloud servers; During dynamic updates, data providers generate new index vectors and replace old shares to achieve file addition/deletion. During the query construction phase, data users use the VSRD query extraction algorithm to encode search keywords into query vectors, which are then split into secret shares and sent to the cloud server along with encrypted identity tags. During the search phase, the cloud server executes the VSRD matching algorithm in parallel (calculating the sum modulus of the index vector and query vector), and returns the results to the validation center; The verification center verifies the correctness of the results through interpolation polynomials. If at least one server returns a correct result, the search is deemed valid and the encrypted file is returned to the user.

Figure 1 shows an example of a system in a two user scenario: different users distinguish queries based on unique identity tags, the cloud server performs matching based on secret shares, and the verification center integrates the results and transfers the correct files.In terms of security, the MFKS system has proven its reliability through three types of security games: IND-CKA security ensures that up to t collusion servers cannot determine the specific keywords corresponding to the index vector; IND-IKGA security ensures that t collusion servers cannot recognize the specific keywords of the query vector; Under the assumption of discrete logarithm (DHI), t-security can resist attacks from up to t malicious servers and avoid the validation center outputting incorrect results. In summary, the MFKS system provides a reliable data search solution for cloud assisted IoT scenarios through multi-user support, dynamic updates, result validation, and strict security proofs, while maintaining zero false positive fuzzy search.

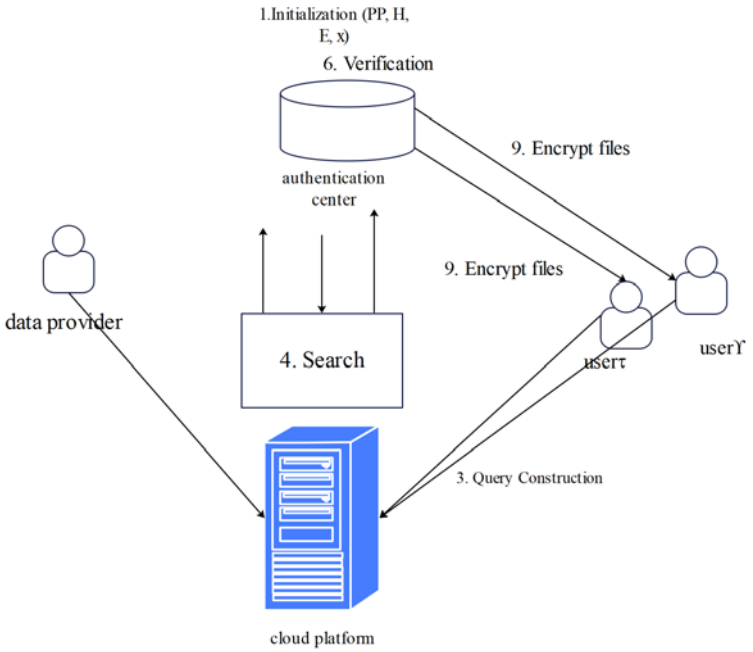

Figure 1. Example of MFKS system with two users

4. Results and discussion

4.1. Efficient dynamic multi-dimensional range query system supporting single dimensional privacy protection

With the widespread application of DBaaS in fields such as electronic healthcare, encrypted data queries are facing challenges due to high computational and storage costs. The traditional searchable encryption scheme mainly supports equal search, which is difficult to meet the demand for expression queries containing attribute keywords and attribute values (such as multi-dimensional range queries in electronic health records) in practical needs. Existing research attempts to implement multi-dimensional range queries through public key encryption or order preserving encryption, but there are issues with high computational costs or leakage of plaintext order; Some solutions decompose multidimensional ranges into single dimensions, but this leads to privacy leakage in single dimensions. Although Chen et al. (ZLG+) and Sun et al. (SZZ+) proposed multi-dimensional range query schemes that support single dimensional privacy based on i-trees and iMinMax trees, and Yang et al. (YGL+) designed a dynamic query scheme that combines i-trees and reversible matrix encryption, it requires multiple interactions to increase communication complexity; However, Zuo et al.'s (ZSL+) serverless interactive dynamic scheme is limited to single dimensional queries, and the tree based indexing structure leads to low update efficiency. Therefore, achieving efficient dynamic multi-dimensional range queries while protecting single dimensional privacy remains an urgent challenge to be solved. A new range query matching method is proposed as the underlying solution to address the above issues, and a three-level index structure (layered by attribute keywords, data ranges, and data values) is introduced to achieve efficient dynamic updates. Subsequently, a dynamic multidimensional range query system (MRQ) is designed. The system combines a new point intersection and interval intersection determination scheme with a three-level indexing structure, which is more efficient than multi-dimensional range query schemes based on tree indexing and symmetric encryption. It also supports complex query types such as ">a" and "<b". Experimental evaluation shows that the MRQ system improves index generation, query

generation, search, and update speed by 1-1.53 times, 1.82-8.2 times, 1.12-3.24 times, and 6.58 times, respectively, and reduces communication overhead by 96%. The MRQ system model consists of four entities: the data owner is responsible for encrypting data records, building secret indexes, and uploading them to the cloud platform; The cloud platform consists of two honest but curious servers that store encrypted files and indexes, collaborate to perform searches, and return results; The verification center generates system parameters and distributes secret parameters, interpolates intermediate results during the search process to determine matching files; Data users generate query vectors and send secret queries. The attack model assumes that the verification center is honest, cloud servers do not collude and comply with protocols but are curious, and data owners and users are honest but curious (possibly eavesdropping on communication between a single entity and a server). The system needs to ensure data privacy (servers cannot infer plaintext from ciphertext), index privacy (users cannot infer indexes from eavesdropping), query privacy (servers cannot infer user queries from secret queries), and single dimensional privacy (servers cannot obtain single dimensional range matching results from intermediate results), in order to achieve efficient multi-dimensional range queries and multi-dimensional privacy protection under attack models.

4.2. Model experiment

This chapter proposes an efficient dynamic multi-dimensional range query system that supports privacy protection, with the core being the combination of data comparison schemes and dynamic indexing structures. In terms of data comparison, point intersection, interval intersection, and range matching are achieved through vector interval determination: the symbols ">", "<", and "=" are encoded as numerical values (such as "=" being 0, "<" being α, and ">" being β), the index and query data are encoded as vectors, and the vector sum is checked through modular operations to determine whether it falls within a specified range (such as all 0, [1, p/2), or (p/2, p)) to determine matching. Combining the Shamir threshold scheme, the secret is shared into two parts to ensure privacy protected decision-making. The dynamic multidimensional range query system consists of four entities: data owner, cloud platform (two servers), verification center, and data user. The data owner encrypts the data and constructs a three-level index structure (layered by attribute keywords, data range, and data values), while the verification center initializes system parameters and distributes secrets; When querying, the user generates a query vector, and the server collaborates with the verification center to execute matching and judgment algorithms (such as privacy based point intersection judgment PPID and interval intersection judgment PIID), returning data that meets the conditions. The system supports dynamic updates and efficiently adds or deletes files through addition and subtraction of decimal file identifiers (such as splitting binary file identifiers into decimal vectors and updating them through modular operations). In terms of privacy protection, it is ensured that the server cannot infer single dimensional matches through intermediate results, and data owners and users cannot infer privacy information through eavesdropping communication, ultimately achieving efficient dynamic multi-dimensional range queries while protecting single dimensional privacy..

4.3. Effect analysis

The performance evaluation of the MRQ Distributed Access Control (DAC) system was conducted on a laptop equipped with an AMD Ryzen 7 4700U processor (2.0 GHz) and 16GB of memory. All entities (data owner, user, verification center, and two servers) were deployed on the same device, and the experiment was run in a Python environment. The dataset is derived from the UCI machine learning library and contains 101766 instances and 66 attributes (23 numerical and 43

non-numerical; categorical attributes such as gender are split into binary indicators). The system parameter settings include a maximum data value of 443867222, modulus p=887734453 (satisfying p>2 x maximum value), and constants a=2393547976 and b=6917430,26. 67 random values were selected for the experiment (24 for indexing titles and 43 for non-numeric attributes), and the decimal part was converted to an integer by multiplying by 100. All tests were run 10000 times and averaged. Performance indicators show that index generation combined with title/interval/data class construction and secret sharing of PIID/PID scheme results in a linear increase in computational and storage overhead with dimension (w), number of intervals (m), and number of records (d). For example, when w=4, m=1000, and d=100000, the computation time is 13.629 seconds and the storage occupies 1103KB; when w=10, m=400, and d=100000, the time drops to 3.551 seconds and the storage is 1094KB. The query generation is based on the PIID extraction algorithm, with an average time of 0.2583 seconds and linearly increases with the dimension w (such as only 3.25 milliseconds when w=10). The search operation covers the matching, judgment, and interpolation of PIID/PID, and the cost is also linearly related to w, m, and d. In typical scenarios (w=4, m=1000, d=100000), it takes 211.795 milliseconds. The update operation (add/delete) only needs to match the title index, avoiding inter region scanning. Its cost increases linearly with the update frequency (such as 450 updates taking 31.9 seconds), and its efficiency is significantly better than search. Compared with existing systems (YG+, ZLG+), MRQ performs well in multiple indicators: index generation speed is 33% faster than YG+and 1-1.5 times faster than ZLG+(N=80/100); The query generation efficiency is 1.8 times that of YG+and 28.2-30 times that of ZLG+, respectively; The search speed is 2.79-2.88 times faster than YG+and 1.12-3.24 times faster than ZLG+; The update operation is 6.58 times faster than YG+. In terms of communication overhead, MRQ only requires 0.038-0.056KB when d=100000, which is 96% lower than YG+'s 1.37-1.84KB. The storage efficiency is also better than the tree index structure of YG+/ZLG+(dependency vector dense nodes). MRQ provides an efficient solution for secure high-dimensional range queries in cloud environments through modular design and linear scalability.

5. Conclusion

With the popularization of cloud technology, the demand for users to encrypt data and store it in the cloud is increasing day by day. How to achieve efficient queries and calculations while ensuring data privacy has become a research focus. This article focuses on the research of fuzzy keyword search, multi-dimensional range query, and calculation functions for privacy data in cloud environments. The following achievements have been made: firstly, a new scheme based on Shamir secret sharing is proposed to address the problem of false positives in fuzzy keyword search and the functional limitations of symmetric encryption systems. It eliminates false positives while achieving multi-user, multi-keyword, dynamic update, and verification functions, which is more efficient than traditional symmetric encryption schemes and has 100% accuracy; Secondly, to address the issues of single dimensional privacy leakage and low dynamic update efficiency in multi-dimensional range search, a low-level index structure is designed, combined with point intersection and range intersection determination techniques, to construct a dynamic multi-dimensional range search system that improves search and update efficiency while ensuring single dimensional privacy; Thirdly, in terms of privacy data calculation, design protocols such as mean, variance, correlation, comparison, and equivalence testing to improve the system's computing capabilities; In addition, the discrete logarithm problem is extended from finite groups to semigroups, and algorithms are proposed to solve the discrete logarithm and its variants in semigroups, providing theoretical support for privacy protection mechanisms. Security and performance analysis indicate that the proposed system combines efficiency and practicality. Future research will focus on three directions:

optimizing fuzzy search result return strategies (such as Top-k problems), enhancing forward and backward privacy protection in dynamic update processes, and exploring security applications of discrete logarithm problems in semi loop environments to further enhance the functionality and security of privacy data processing systems.

References

[1] *Yang D, Liu X. Collaborative Algorithm for User Trust and Data Security Based on Blockchain and Machine Learning[J]. Procedia Computer Science, 2025, 262: 757-765.*

[2] *Llamas, J. M., Vranckaert, K., Preuveneers, D., & Joosen, W. (2025). Balancing Security and Privacy: Web Bot Detection, Privacy Challenges, and Regulatory Compliance under the GDPR and AI Act. Open Research Europe, 5, 76.*

[3] *Wang, Y., Cai, C., **ao, Z., & Lam, P. E. (2025). LLM Access Shield: Domain-Specific LLM Framework for Privacy Policy Compliance. arxiv preprint arxiv:2505.17145.*

[4] *Zhu, Z. (2025). Cutting-Edge Challenges and Solutions for the Integration of VectLu, C. (2025). The Application of Point Cloud Data Registration Algorithm Optimization in Smart City Infrastructure. European Journal of Engineering and Technologies, 1(1), 39-45.*

[5] *Silva, C., Felisberto, J., Barraca, J. P., & Salvador, P. (2025). ASAP 2.0: Autonomous & proactive detection of malicious applications for privacy quantification in 6G network services. Computer Communications, 237, 108145.*

[6] *Pan, H. (2025). Development and Optimization of Social Network Systems on Machine Learning. European Journal of AI, Computing & Informatics, 1(2), 73-79.*

[7] *Tang X, Wu X, Bao W. Intelligent Prediction-Inventory-Scheduling Closed-Loop Nearshore Supply Chain Decision System[J]. Advances in Management and Intelligent Technologies, 2025, 1(4).*

[8] *Prabowo, S., Putrada, A. G., Oktaviani, I. D., Abdurohman, M., Janssen, M., Nuha, H. H., & Sutikno, S. (2025). Privacy-Preserving Tools and Technologies: Government Adoption and Challenges. Ieee Access.*

[9] *Wang, C. (2025). Exploration of Optimization Paths Based on Data Modeling in Financial Investment Decision-Making. European Journal of Business, Economics & Management, 1(3), 17-23.*

[10] *Xu, H. (2025). Research on the Implementation Path of Resource Optimization and Sustainable Development of Supply Chain. International Journal of Humanities and Social Science, 1(2), 12-18.*

[11] *Ali, M., Arunasalam, A., & Farrukh, H. (2025, May). Understanding Users' Security and Privacy Concerns and Attitudes Towards Conversational AI Platforms. In 2025 IEEE Symposium on Security and Privacy (SP) (pp. 298-316). IEEE.*

[12] *Zhu, Z. (2025). Application of Database Performance Optimization Technology in Large-Scale AI Infrastructure. European Journal of Engineering and Technologies, 1(1), 60-67.*

[13] *Wei, X. (2025). Practical Application of Data Analysis Technology in Startup Company Investment Evaluation. Economics and Management Innovation, 2(4), 33-38.*

[14] *Deng, H. W., Salek, M. S., Rahman, M., Chowdhury, M., Shue, M., & Apon, A. W. (2025). Leveraging public cloud infrastructure for real-time connected vehicle speed advisory at a signalized corridor. International Journal of Transportation Science and Technology, 17, 131-147.*

[15] *Li, W. (2025). Research on Optimization of M&A Financial Due Diligence Process Based on Data Analysis. Journal of Computer, Signal, and System Research, 2(5), 115-121.*

[16] Huang, J. (2025). Research on Resource Prediction and Load Balancing Strategies Based on Big Data in Cloud Computing Platform. Artificial Intelligence and Digital Technology, 2(1), 49-55.

[17] Xu, H. (2025). Supply Chain Digital Transformation and Standardized Processes Enhance Operational Efficiency. Journal of Computer, Signal, and System Research, 2(5), 101-107.

[18] Li W. Building a Credit Risk Data Management and Analysis System for Financial Markets Based on Blockchain Data Storage and Encryption Technology[C]//2025 3rd International Conference on Data Science and Network Security (ICDSNS). IEEE, 2025: 1-7.