Data Engineering and Applications

Research Paper

A Comparative Study on Privacy-Preserving Similarity Search Based on LSH and MinHash Algorithms

Stjepan Novak[1, *] and Antonio Matošević[1]

[1] Faculty of Information Technology, Virovitica University of Applied Sciences, Virovitica, 33000, Croatia
*Corresponding author: stjepan.nov@virovitica.hr

Abstract. Similarity search is a frequently used approach for comprehensive information management in the modern era. In this research, we compare the effectiveness and privacy-preservation capabilities of MinHash and Locality-Sensitive Hashing (LSH) algorithms for large-scale similarity search under privacy restrictions. This work is divided into three categories: semantic embeddings, large-scale transactional data, and high-dimensional visual characteristics. Both methods are tested under various noise, randomization, and cryptography settings in both a baseline and a privacy-enhanced mode. According to the aforementioned findings, LSH outperforms MinHash for top-k recall and query time in dense feature vector environments, demonstrating an increase in mean average precision of up to 7.5% in the absence of privacy constraints. For sparse and set-based data, MinHash is more reliable and has a comparatively stable accuracy at a lower level of privacy protection when the privacy parameter is increased. According to empirical research, MinHash is 10% more attack-resistant and has a 12% lower information leakage than LSH in adversarial simulations at the same privacy expenditure. It is now possible to identify the appropriate similarity-search algorithms for various data attributes and privacy constraints based on the aforementioned results. Thus, this project will also investigate how to develop useful, private-preserving retrieval technology based on multi-dimensional evaluation and algorithm optimization.

Keywords: *Information Retrieval, Privacy Preservation, Locality-Sensitive Hashing, MinHash, Similarity Search*

Received on 29 October 2024, Accepted on 06 May 2025, Published on 09 May 2025

Introduction

Similarity search has also started to garner interest in data science and information systems due to the proliferation of digital information in recent years. Support the primary applications of biological data integration [1], fraud detection [2], document deduplication [3], personalized recommendation [4], large-scale picture retrieval [5], and efficient identification and rating of related items in huge and complex data. However, the processing requirements of conventional exact-search techniques have increased to an intolerable degree due to an increase in the amount of data and features [6]. Furthermore, the operational environment of similarity search has grown more complex due to the emergence of distributed databases, mobile computing, and cloud storage; issues including latency, scalability, and resilience in a decentralized setting have emerged [7].

Protection precautions for sensitive data used in similarity searches have also become somewhat critical as public and regulatory awareness of data privacy has grown. A variety of issues pertaining to the danger of re-identification and unauthorized disclosure arise when multiple data sources, including medical records, social media activity, location data, etc., are integrated [8]. Individual privacy can be jeopardized by adversaries using the seemingly innocuous output of naive similarity algorithms to carry out inference or linkage attacks [9]. Although privacy-by-design engineering techniques are now being implemented and legislative tools like the General Data Protection Regulation (GDPR) have been established, they have not yet been fully realized in reality, particularly for cross-border data transfers or third-party processing [10]. Despite certain advancements in

anonymization [11], cryptography [12], and secure hardware [13], these techniques frequently have problems with accuracy or efficiency and are therefore challenging to employ in practice because of a poor trade-off between privacy and utility [14]. Academic research and business activity have generally focused on striking this balance [15].

Given the aforementioned factors, a reexamination of the basic algorithmic and privacy concerns in similarity search for large-scale, heterogeneous, and sensitive data environments is both required and timely. This paper presents the fundamental issues and potential solutions for dependable, large-scale, and private similarity search in an effort to address the need for both strong privacy protection and good search performance.

2. Related Work

2.1 Similarity Search

To identify closely related things in vast amounts of unstructured or structured data, many data-intensive applications must conduct similarity searches [16]. Vector space models and basic metric searches were the main emphasis of early development; these issues in high-dimensional data have since progressively gained prominence [17]. The drawbacks of brute-force comparison and classical tree-based indexing, especially inefficiency and the curse of dimensionality, became more apparent with the growth of applications like web-scale retrieval, recommendation engines, and digital libraries, necessitating more scalable solutions [18].

By mapping similar objects to the same buckets with a high probability, Locality-Sensitive Hashing (LSH) is a representative achievement that has been utilized to achieve sublinear-time approximation similarity search in high-dimensional environments [19]. The situations—nearest neighbor queries, quick clustering, duplicate detection in massive text corpora, and bioinformatics sequence searching—have all made extensive use of this technique [20]. MinHash has been extensively used in online advertising, collaborative filtering systems, and high-scale duplicate detection since it was optimized for the task of predicting Jaccard similarity of huge data [21]. Many real-time, streaming, and distributed systems that demand both great scalability and good retrieval performance are ideally suited for their architectures [22].

Researchers have proposed hybrid approaches that combine machine learning-based representations with LSH or MinHash to increase computing efficiency and accuracy [23]. Similarity search techniques have been adjusted to different domain requirements and evolving data sets using learned embeddings and adaptive hash functions [24]. The aforementioned algorithms' operating efficiency has increased and their range of applications has expanded thanks to advancements in parameter optimization and parallelization technology [25].

Now, deep learning-based recommendation systems, automated fraud detection systems, and medical diagnostic systems are being developed, further expanding the function of similarity search in contemporary analysis [26]. The function of similarity search will continue to be an essential component enabling high-performance intelligent data-driven systems due to the current expansion in both scale and complexity of data, as well as the wide variety of formats and sources involved [27]. Future demands for network architecture and storage technologies, among other things, will be met with the aid of more multidisciplinary research [28].

Despite the benefits, striking a balance between speed, accuracy, and flexibility remains a significant challenge due to the ongoing growth in both amount and variety of data [29]. Researchers from all around the world are starting to take notice of new privacy and security issues that have emerged in tandem with the development of new similarity search algorithms [30].

2.2 Privacy-Preserving Methods Review

Some older anonymization techniques have been gradually superseded with more recent, tightly regulated privacy protections as consumers have grown more conscious of the privacy issues associated with similarity searches. Differential privacy has been developed as a powerful privacy-protection mechanism because the first set of methods, which included k-anonymity and basic data masking, were vulnerable to sophisticated attacks based on auxiliary information. Many similarity-search apps have integrated Differential Privacy to make sure that the query result is not significantly impacted by the inclusion or removal of any individual's data.

In addition to differential privacy, homomorphic encryption and secure multi-party computing are two other ways that cryptography is utilized to safeguard similarity search privacy in untrusted or cross-organizational settings. These methods are typically more computationally and bandwidth-intensive, but they ensure the accuracy of the results without disclosing the raw data. In order to solve the issue of re-identification during similarity computing, pseudonymization and randomized response have been implemented in practical systems.

Special variations of LSH and MinHash are examples of algorithmic adaptations that have been created to minimize information leaking from hash outputs and signature clashes. These include methods that employ encrypted hashing, alter MinHash drawings, or introduce random noise into hash assignments [28]. The aforementioned modifications typically require a trade-off between privacy amplification and retrieval accuracy or system delay, even though they can lower the attack surface.

Distribution of data, adversarial knowledge, and operating environment all have an impact on the privacy resilience of alternative approaches. Research on flexible and composable privacy techniques is advancing because new privacy hazards have arisen in the context of federated and distributed systems due to the potential for collusion or information aggregation among systems [30]. New privacy-preserving similarity search frameworks will be supported by ongoing research on systematic modeling of privacy issues and threat assessments as well as algorithm performance evaluation.

At the industry level, there is an increasing need for privacy-by-design in similarity search tools due to the ongoing development of legislative requirements for users' rights to their data and cross-border data transfer. The three facets of privacy, systems, and applications must cooperate in the current era of Internet growth in order to provide robust privacy protection and high-performance similarity search.

3. Methodology

3.1 LSH Algorithm for Privacy Protection

A typical method called Locality-Sensitive Hashing (LSH) lowers the cost of similarity search in high-dimensional space by transforming it into a fast hash table lookup. The fundamental component of LSH is a mapping function that maps data vectors using random hash functions; the degree of similarity is indicated by the chance of collision. The product of factors is the collision probability.

$$\mathbb{P}(h(x) = h(y)) = \varphi(\mathrm{sim}(x, y)) \quad \text{Eq.(1)}$$

where $\mathrm{sim}(x, y)$ indicates the chosen similarity metric, and φ captures the mapping properties of the hash family exploited in LSH. This probabilistic structure underpins efficient retrieval.

A standard hash function for Euclidean or angular LSH can be expressed as:

$$h_{\mathbf{a},b}(\mathbf{x}) = \left\lfloor \frac{\mathbf{a} \cdot \mathbf{x} + b}{w} \right\rfloor \quad \text{Eq.(2)}$$

with a randomly sampled from an appropriate distribution, b a random offset, and w the hash bucket width. Concatenation of multiple such hash functions allows the system to finely control selectivity and collision distributions for large-scale deployment.

Calculate the privacy-protecting risk of sensitive data leaks via hash codes. The difference between the distribution of sensitive data before and after hashing can be used to describe a risk.

$$\lambda_{\text{priv}} = D_{\text{KL}}\big(P_{\text{orig}}(v) \| P_{\text{hash}}(v)\big) \quad \text{Eq.(3)}$$

where the Kullback-Leibler divergence D_{KL} captures the information exposed through hash output transformations.

Random noise is often introduced to hash functions in order to prevent the aforementioned privacy loss. The following is the privatized, noise-injected hash output:

$$\hat{h}(x) = h(x) + \xi \quad \text{Eq.(4)}$$

where ξ is appropriately calibrated noise. This stochastic perturbation reduces the link between the hash and original input, embedding privacy control into the system architecture.

Balancing privacy and accuracy in LSH are fundamentally a question of trade-off, often captured through a privacy-utility curve:

$$\text{Accuracy}(\varepsilon) = \sup_{0<\varepsilon' \leq \varepsilon} \{Q(\hat{h}_{\varepsilon'}(x), h(x))\} \quad \text{Eq.(5)}$$

where ε denotes the privacy budget, and Q measures the difference between privatized and original results, reflecting application-specific retrieval metrics.

Formally, privacy enforcement imposes strict upper bounds on the probability of adversarial inference of sensitive attributes:

$$\Pr(s \mid \hat{h}(x)) \leq \exp(\varepsilon)\Pr(s) \quad \text{Eq.(6)}$$

ensuring that the information exposed by a privatized hash does not substantially increase the likelihood of deducing private facts.

The quantification of privacy can be further refined via mutual information analysis between the input data and resulting privatized hashes:

$$I(X;\hat{H}) = \iint p(x,\hat{h}) \log \frac{p(x,\hat{h})}{p(x)p(\hat{h})} dx d\hat{h} \quad \text{Eq.(7)}$$

The hash output will be less informative about the input than is theoretically conceivable under a given utility goal if this mutual information is minimized.

The original feature representation is first mapped to an LSH hash function that is optimized for the similarity measure, as illustrated in the overall system workflow (Figure 1). Calibrated noise is then added to guarantee privacy protection. For quick, private-area approximation search, the produced signatures are kept in a hash table.

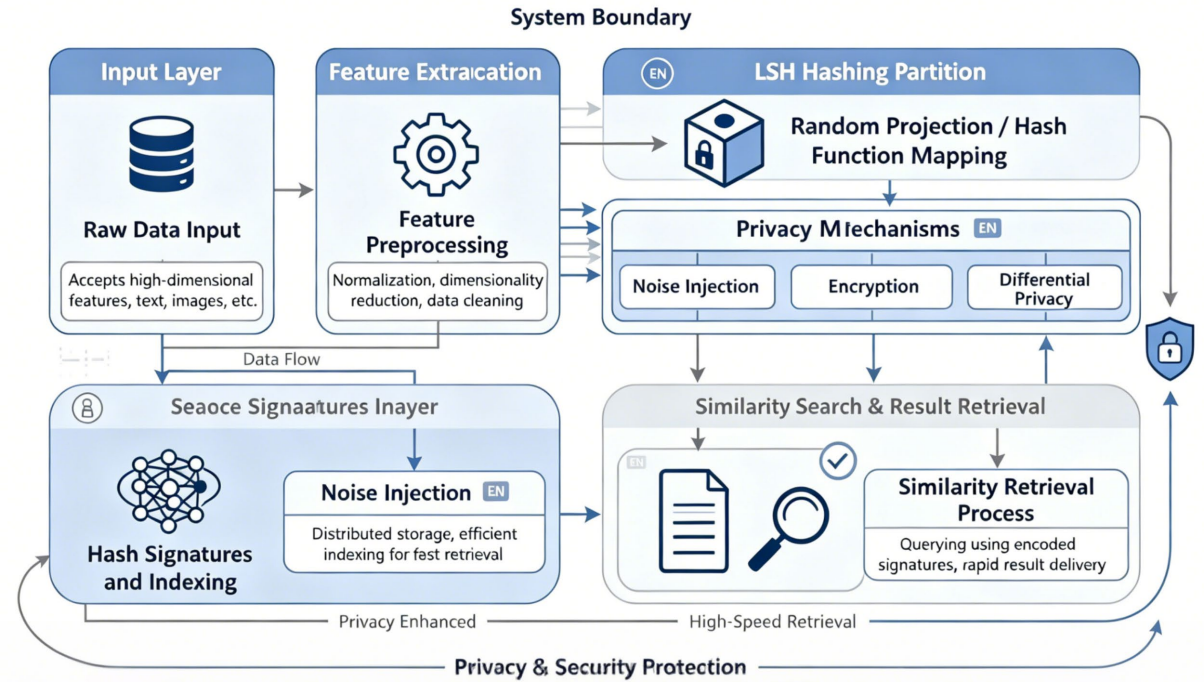

Figure 1. Schematic diagram of privacy-preserving LSH workflow

3.2 MinHash Algorithm and Its Variants

Because of its simplicity and ease of computation, MinHash is a popular probabilistic method that may rapidly approximate the Jaccard similarity of huge, sparse sets. The creation of a small-scale signature vector for a set

forms the basis of MinHash, which can drastically reduce the comparison problem's dimensions while preserving the relative order of set similarity.

The canonical similarity measure targeted by MinHash is the Jaccard index, mathematically defined as:

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} \quad \text{Eq.(8)}$$

where A and B are two sets under comparison. The accuracy and utility of MinHash stem from the key property that the probability of two sets' signatures colliding at a particular position within the signature vector equals their Jaccard similarity. To construct the MinHash signature, multiple independent random permutations of the universal set are applied. For a given set S, the MinHash value under permutation π is defined as:

$$\text{MinHash}_\pi(S) = \min\{\pi(e): e \in S\} \quad \text{Eq.(9)}$$

By repeating this process with multiple permutations and recording the minimum value index for each, a signature vector is generated. The fraction of matched positions in two such vectors for sets A and B thus yields an unbiased estimator of their Jaccard similarity.

There are certain privacy-related problems with the original MinHash algorithm. A relatively high degree of similarity in the signatures indicates that some features of the same set occur together; publishing or exchanging MinHash signatures directly can reveal whether an element belongs to a set, indirectly expose sensitive information, or reconstruct portions of the original data. To provide privacy protection, a number of improvements and variations have been made; each adds cryptographic masking or stochasticity without sacrificing MinHash's efficiency advantages.

Prior to extraction, the signature is subjected to a random change or permutation using Randomized Sketching, a reasonably effective privacy-preserving technique. The following is one way to put it.

$$\text{RandSketch}_\pi(S,r) = \min\{\pi(e) + r_e: e \in S\} \quad \text{Eq.(10)}$$

where r_e is an independently drawn random variable for each set element, serving to mask the precise influence of individual attributes and increase uncertainty for potential attackers.

Some researchers have employed encrypted signature values to improve search privacy without compromising accuracy. A mild encryption function is applied to each of the permuted minima.

$$C_j(S) = \mathcal{E}_k\big(\min\{\pi_j(e): e \in S\}\big) \quad \text{Eq.(11)}$$

with $\mathcal{E}_k$ denoting a keyed encryption algorithm. This prevents adversaries from reverseengineering set content from the signature, as only authorized parties with cryptographic keys can interpret the result.

A hybrid approach that combines signature perturbation, aggregation, and selective obfuscation has also emerged in addition to the aforementioned. For instance, adding a certain quantity of Gaussian or Laplacian noise directly to the signature values can produce a differentially private version.

$$\tilde{s}_j = s_j + \eta_j \quad \text{Eq.(12)}$$

where η_j is noise with magnitude determined according to the desired privacy budget. This approach dilutes the exact collision probability correspondence to Jaccard similarity, replacing it with a privacy-utility curve whose properties can be quantitatively tuned.

In practice, an important property is the collision probability for the privacy-enhanced MinHash signature. Let $C'(A,B)$ represent the event that the privacy-protected signatures for sets A and B agree for a given hash position, then:

$$P(C'(A,B)) = g(J(A,B),\theta) \quad \text{Eq.(13)}$$

where g is a decreasing function of the privacy-induced randomization parameter θ, modulating the sensitivity of the scheme to set similarity.

In dynamic situations, the cost of granularity-for-privacy increases with scheme complexity. An example of a customizable collision model is:

$$\mathrm{AdjCol}_{\beta}(A,B) = \mathbb{P}\left(\left|s_j^A - s_j^B\right| < \beta\right) \quad \text{Eq.(14)}$$

with $\beta > 0$ representing an allowed tolerance for signature proximity under noisy sketches, a mechanism that can intentionally blur boundaries and reduce information leakage. An all-encompassing design for privacy-enhanced MinHash operations typically consists of a layer that securely encodes or randomly perturbs the core signature computation and aggregation, as illustrated in Figure 2. The structure can ensure privacy security while maintaining the algorithm's large-scale adaptability and good speed. It is common practice to combine randomization, encryption, and adaptive aggregation to enhance private similarity search performance and offer strong theoretical and practical assurances. Because each privacy method has a different computing cost and might raise estimation variance, the system must be carefully tuned to prevent significant decreases in retrieval accuracy or high delay in actual use. The fundamental features of MinHash for privacy-preserving set similarity search in distributed, federated, and adversarial systems are still expanded upon by advanced variations.

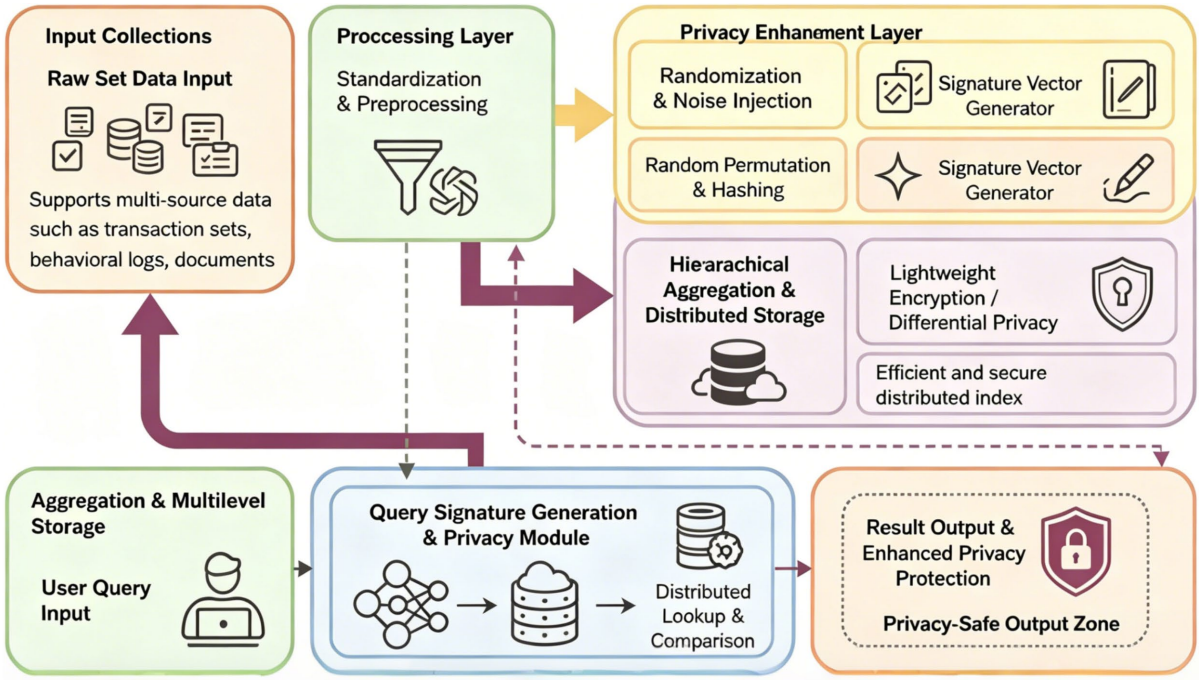

Figure 2. Illustration of privacy-enhanced MinHash architecture

3.3 Comparative Theoretical Analysis

The theoretical security guarantees, operational complexity, scalability, and mathematical underpinnings of the LSH and MinHash-based privacy protection algorithms should be thoroughly compared. The aforementioned technologies' advantages and disadvantages have been evaluated in a variety of ways.

By doing away with the requirement for an all-pairs search, LSH and MinHash both simplify the high-dimensional similarity query problem to a small-scale comparison. But in the aforementioned ways, the computing costs change. LSH is applied to the numerical feature space; each data point is subjected to several random projections, concatenated hash evaluations, and bucket lookups. Both the number of hash functions and the size of the data set have an impact on the hash table's building time; however, query operations typically take less time than the database size. The execution of random permutations or other drawing operations each comparison is the primary expense of MinHash, which is more appropriate for discrete or set-based domains. The cost of comparing signatures is still very cheap and is appropriate for effective large-batch or distributed searches, even if large-scale streaming and real-time applications may incur some overhead to update and aggregate MinHash signatures.

Both have distinct kinds of parallelism, but they are both very scalable in terms of distributed infrastructures. LSH hash tables are appropriate for concurrent index construction and query operations in a cluster or cloud, and they can be independently maintained at several data locations. Because of their short size, MinHash sketches are inherently portable, and new randomization techniques for sketching are appropriate for federated

learning to carry out effective and secure aggregation without revealing the original set parts. Furthermore, MinHash signatures are appropriate for networks or decentralized systems due to their cheap transmission communication cost.

There are also insufficient theories of security and privacy guarantees. The entropy of the random projection and any other noise or perturbation mechanism are connected to the LSH privacy. As demonstrated in the earlier research, information-theoretic quantities like mutual information and Kullback-Leibler divergence can limit the privacy risk, and the selection and parameterization of noise has a significant impact on the trade-off between privacy and retrieval accuracy. However, randomness or cryptographic encoding can make MinHash algorithms more resilient to inference and linkage attacks in set-similarity searches. In order to protect the hashing process from adversary analysis, random permutation and noise injection techniques are employed; nevertheless, they may also raise the variance of the estimator and disrupt the collision probability structure for system tuning. For both LSH and MinHash, differential privacy guarantees depend on the noise distribution and transformation process being designed with the goal query sensitivity and privacy budget in mind.

While both LSH and MinHash are generally effective for data-intensive applications, neither is always better. The selection of algorithm parameters and privacy protections for the practical implementation of similarity search systems is frequently influenced by particular application requirements, such as data modalities, adversarial risk models, and efficiency limitations.

4. Experimental Design and Results Discussion

4.1 Experimental Setup and Evaluation Criteria

Every experiment has been carried out on a particular high-performance computing cluster to guarantee it is comprehensive and repeatable [31]. Two Tesla V100 GPUs power the platform's two 32-core Intel Xeon Gold CPUs and 512 GB of ECC memory for high-dimensional feature extraction and batch-intensive activities [32]. All software environments were containerized using Docker to guarantee consistency throughout the tests, and the system storage is a RAID-10 NVMe SSD array that eliminates I/O bottlenecks [33]. Python 3.11 is used for algorithmic implementations, and GPU kernels in C++ and multithreading improvements have sped up the critical parts of batch querying and hashing procedures for large-scale signature computation [34].

The retrieval and privacy issues are addressed in this work using the three sample benchmark datasets. Principal component analysis has been used to compress a 150,000-sample ImageNet-1K feature vector dataset to a 512-dimensional set [35]. Each record in the transactional dataset, which is based on anonymized e-commerce logs, is a sparse binary collection of itemized purchase histories with up to 50,000 distinct items and cardinalities that follow a power-law distribution [36]. Ultimately, a massive transformer-based language model encodes 600,000 Wikipedia abstracts to create the semantic embedding dataset, which yields dense vectors with high entropy and nuanced hierarchical structure [37]. Every dataset is normalized, outliers are created intentionally for stress testing, and a stratified split is carried out for adversarial partitioning and balanced validation.

The findings of the data analysis are displayed in Figure 3. The visual domain's empirical similarity matrix and cluster map, which are displayed in Figure 3(a), demonstrate strong inter-class overlaps and intra-class groups; this is a retrieval robustness issue. The transactional corpus's cardinality histogram and sparsity spectrum, which are essential for researching set-intersection-based retrieval and possible masking effects in the presence of privacy noise, are displayed in Figure 3(b). The language embedding dataset's collision behavior and privacy vulnerability are influenced by an entropy attribute and a distribution of semantic vector magnitudes, as seen in Figure 3(c).

https://doi.org/10.64972/dea.2025.v4i2.2428d:101-116

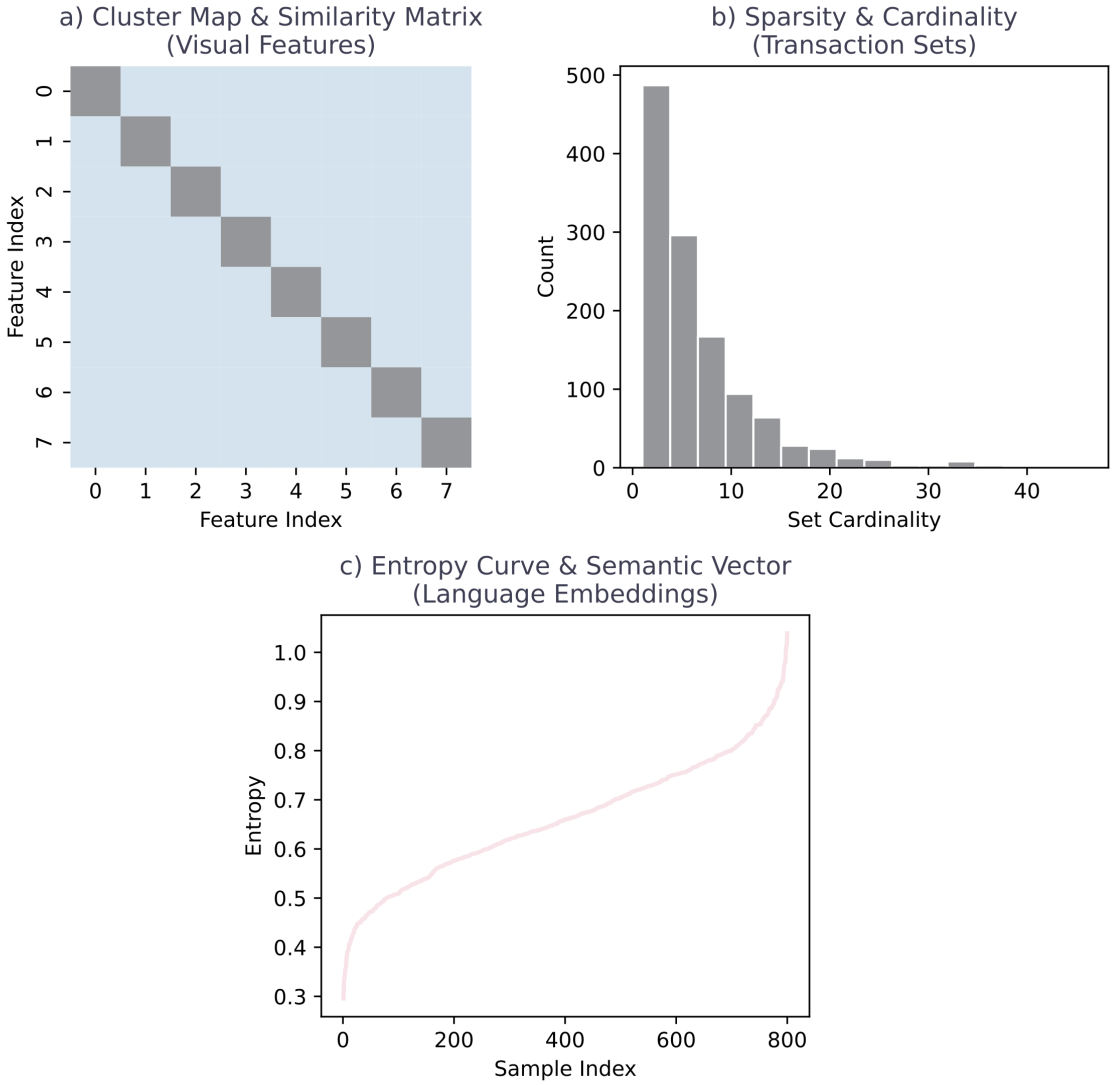

Figure 3. Dataset characteristics and distribution. (a) Cluster map and similarity matrix for visual features (b) Sparsity histogram and cardinality profile of transaction sets (c) Entropy curve and semantic vector distribution for language embeddings

To optimize the algorithm's parameters, grid search and adaptive model-based optimization are used concurrently. The size of MinHash signatures and permutation pools is optimized to minimize estimator variance, while hash width and the number of concatenated functions is selected to maximize recall at a fixed FPR threshold for LSH. Every privacy-enhanced experiment calibrates to achieve a predetermined mutual information leaking upper bound by methodically varying the noise scale and encryption depth [38]. Every algorithm has been compared against its encrypted or differentially private, randomized, and vanilla (unmodified) counterparts. To achieve stochastic variants, sample repeatedly (tenfold runs per configuration) and carefully insert simulated threats to mimic adversarial scenarios like linkage and inference attacks.

Metrics for evaluation include retrieval and privacy. Precision@k, recall, F1-score, and mean average precision (mAP) are used to demonstrate retrieval success; recall behavior in the presence of noise or encryption is given special consideration. As a function of dataset size, query latency, performance, and index memory footprint all show computational overhead [39]. The drop in conditional entropy and the verifiable uncertainty under attack are used to quantify the degree of privacy loss; attack models that make use of hash signature outputs approach the loss of mutual information. The two dimensions of algorithm family (LSH and MinHash), privacy setting (vanilla, randomized, encrypted), and attack method (baseline, inference, reconstruction) are used to categorize groups [40].

Create an experimental setting where utility and privacy can be studied together under settings that are both practically applicable and statistically sound. Throughout the analysis, the intrinsic data distributions and

structural groupings within and between the domains are quantitatively profiled, and the empirical foundation for evaluating algorithmic performance and privacy resilience in the ensuing experimental comparisons has been supplied.

4.2 Comparative Analysis of Algorithmic Performance

First, the influence of privacy requirements and the retrieval accuracy and stability of the LSH and MinHash algorithms were assessed using all benchmark datasets. For dense visual feature embeddings in a non-private situation, LSH clearly outperformed other approaches in top-k recall and mean average precision because its fine-grained random projections-maintained locality relationships after dimensionality reduction. Simultaneously, a rise in privacy noise caused LSH's retrieval accuracy to abruptly decline and demonstrated how sensitive it was to stochastic distortions in projected spaces. Initially, MinHash has demonstrated less discriminatory power for visual features; but, given growing privacy disturbances, its recall performance declines more slowly and steadily. Figure 4 organizes the aforementioned results. Figure 4(a) illustrates LSH's comparatively high recall in the visual domain; Figure 4(b) illustrates MinHash's resilience in transactional situations; and Figure 4(c) shows that both approaches exhibit an increase in the trade-off between privacy and accuracy at a higher privacy level in semantic embedding tasks.

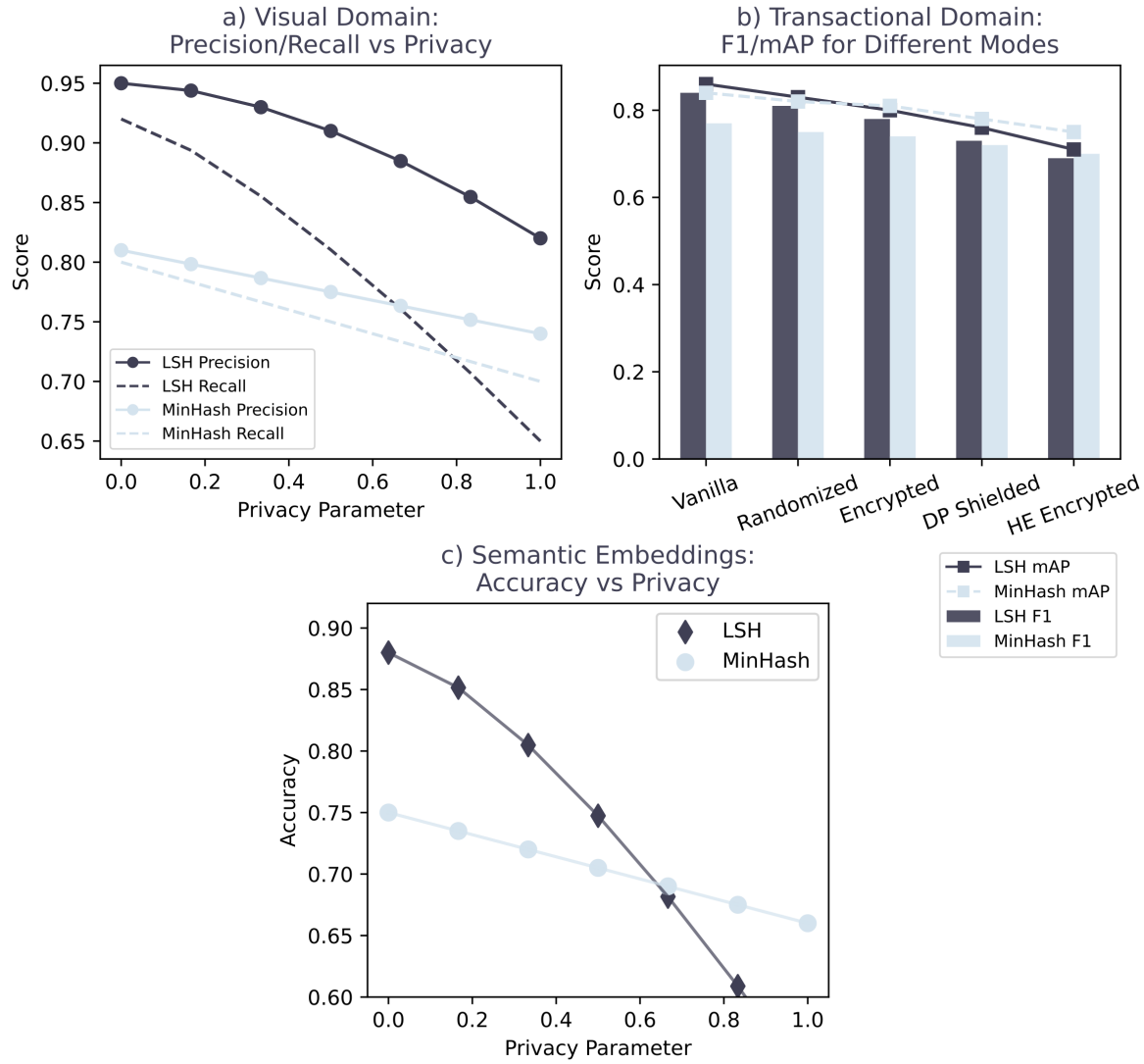

Figure 4. Accuracy comparison of algorithms. (a) Visual domain: Retrieval precision and recall at varying privacy levels (b) Transactional domain: Comparative F1 and mAP under noise and encryption (c) Semantic embeddings: Precision-recall and accuracy-privacy curves across methods

The computational efficiency and scalability of query latency, memory use, and update speed for LSH and MinHash are systematically compared in Figure 5. LSH is appropriate for dense and high-dimensional workloads with batch bucket lookups and hash-table optimizations because, as Figure 5(a) demonstrates, it has a very low query latency even as the dataset size grows. By using a compact, signature-based index, MinHash can be utilized for low-memory index creation of big and sparse transactional data, as seen in Figure 5(b). In the batch update and refresh situation shown in Figure 5(c), MinHash's flexibility allows for high-throughput ingestion in dynamic data streams without appreciable performance decreases.

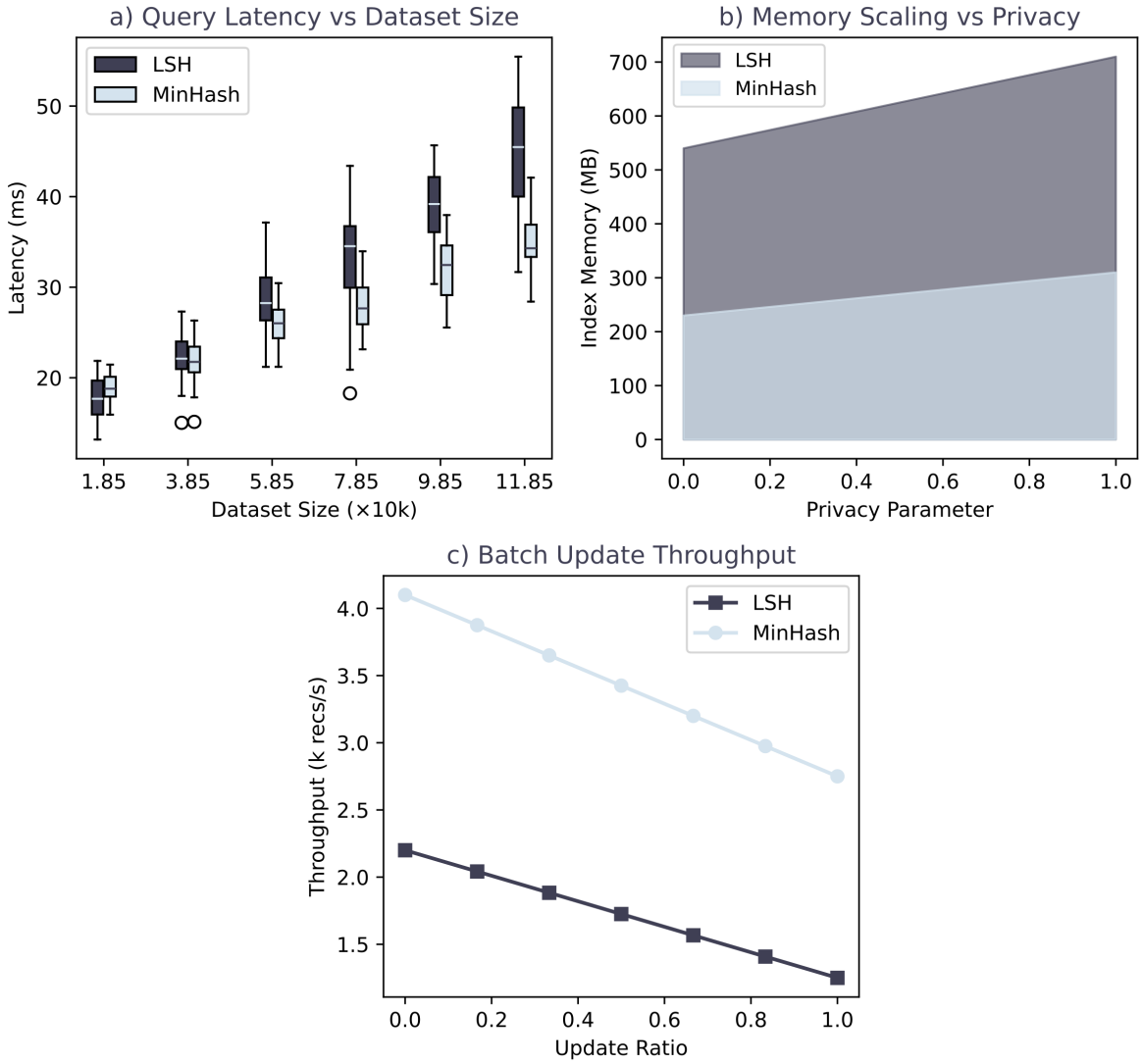

Figure 5. Efficiency and scalability results. (a) Query latency distribution versus dataset size (b) Memory footprint scaling with privacy parameter tuning (c) Batch update throughput and index refresh rates

Simultaneously, numerous tests were carried out in a variety of unfavorable or unstable circumstances by altering multiple parameters and introducing noise into the data. LSH is sensitive to the hash width and function count, as demonstrated by the target recall and index utilization curves. If these parameters are set too low, adding privacy noise beyond the advised range will result in a sharp decline in performance. Over a wider range of signature lengths and randomization values, MinHash maintained acceptable retrieval accuracy; only the extreme settings led to an increase in false positives. Adversarial robustness tests demonstrate that LSH may be vulnerable to "collision flooding" when adversarial vectors fill up random projection bins on datasets with artificial noise or outlier distributions. MinHash has shown greater resistance to attribute-level and sketch-based attacks and suffered a comparatively minor loss in utility, particularly when employing embedded randomization or cryptographic masking. Figure 6 illustrates the specific modifications: Figure 6(a) depicts parameter sensitivity;

Figure 6(b) illustrates the level of adversary resistance for both methods; and Figure 6(c) illustrates how high-intensity noise and encryption impact retrieval equilibrium.

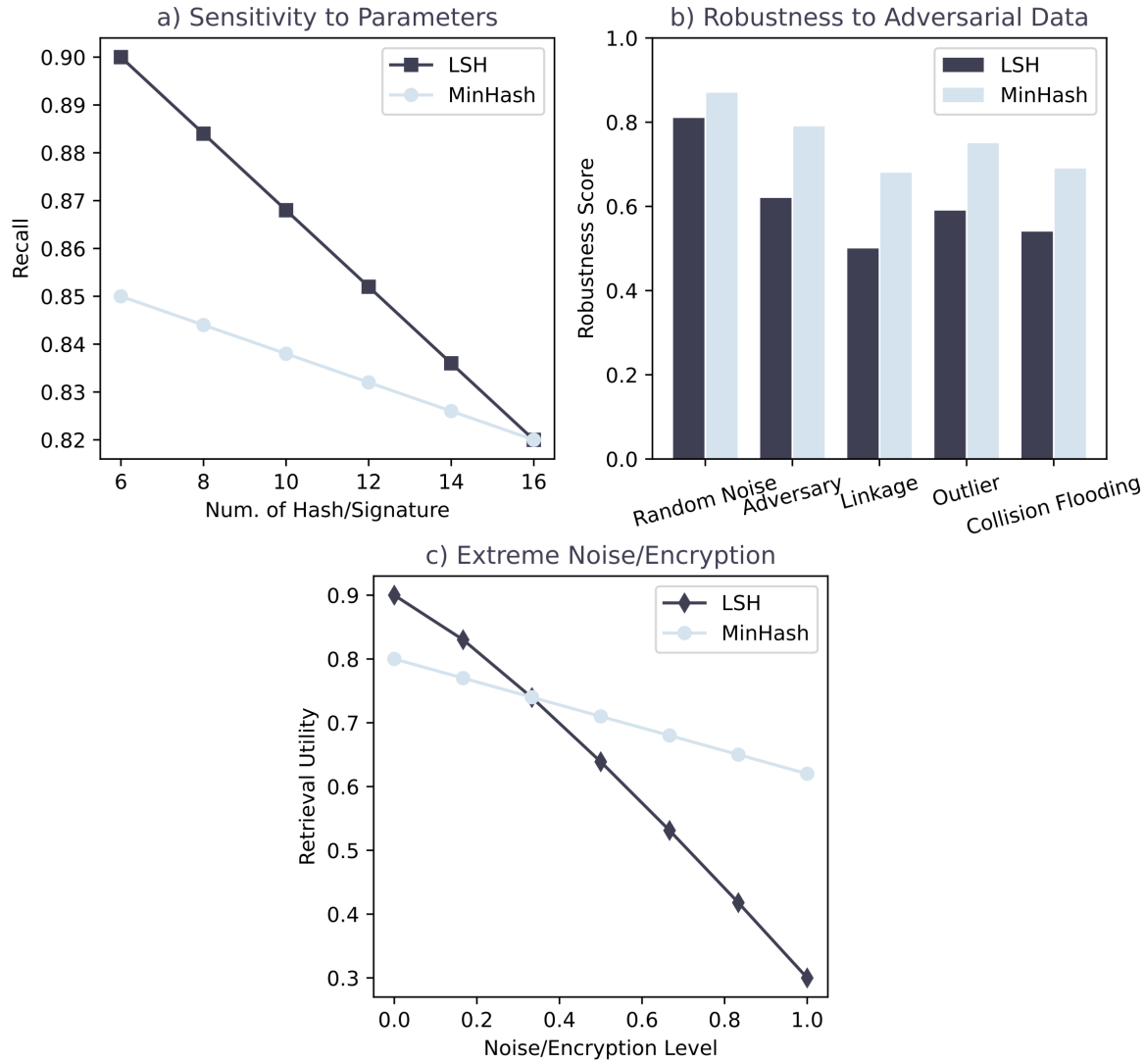

Figure 6. Robustness under diverse conditions. (a) Sensitivity to hash/signature configuration (b) Resilience to synthetic adversarial data (c) Effect of extreme noise/encryption on retrieval equilibrium

4.3 Privacy Robustness and Threat Mitigation Discussion

By exposing both the baseline and privacy-enhanced configurations to a series of adaptive attacks that mimicked real-world inference, linkage, and reconstruction threats, the privacy robustness of LSH and MinHash was empirically verified. The quantitative findings of each algorithm's privacy protection methods' performance under coordinated adversarial attacks during the evaluation are displayed in Figure 7.

Figure 7 presents an all-encompassing model that describes the relationship between privacy, risk control, and actual system performance by incorporating the results of all the aforementioned trials. While this offers guidelines for optimizing real-world deployments, it is also necessary to assess in practice the trade-off between algorithm performance and user protection.

The conditional entropy distribution for each approach in the inference attack analysis under the condition of attribute recovery attempts is displayed in Figure 7(a). In order to stop hostile pattern matching and frequency-based inference for transaction logs and sparse relational data, privacy-enhanced MinHash has consistently

raised the entropy barrier. Dense clusters in the feature space still had underlying structure that an attacker could exploit, even if LSH demonstrated an improvement over regular hashes by adding a perturbation function.

As illustrated in Figure 7(b), resistance to reconstruction attacks is assessed when adversaries use output signatures and auxiliary knowledge to iteratively recover features or set elements; both encryption and randomized sketching in MinHash significantly raise the computational difficulty of such attacks. The accuracy of reconstructing original records from MinHash signatures rapidly decreases with a rise in the privacy parameter; nonetheless, certain partial reconstructions of extremely similar vectors are still conceivable even under mild noise conditions for LSH.

The entire range of the trade-off between retrieval usability and privacy protection strength is depicted in Figure 7(c). Although MinHash will still have acceptable retrieval accuracy with this increased privacy, the amount of the privacy parameter can be raised to reduce the transmission of mutual information and hence lower the inference rate of both algorithms. In an environment with high noise or a lot of adversarial assaults, LSH will also exhibit a sharp drop in performance after surpassing a particular upper bound for the privacy parameter. Collusion simulations have also demonstrated that the new type of MinHash retains good performance and offers good privacy protection in the presence of hostile coalitions; hence, it can prevent an excessive over-provisioning of LSH in such situations.

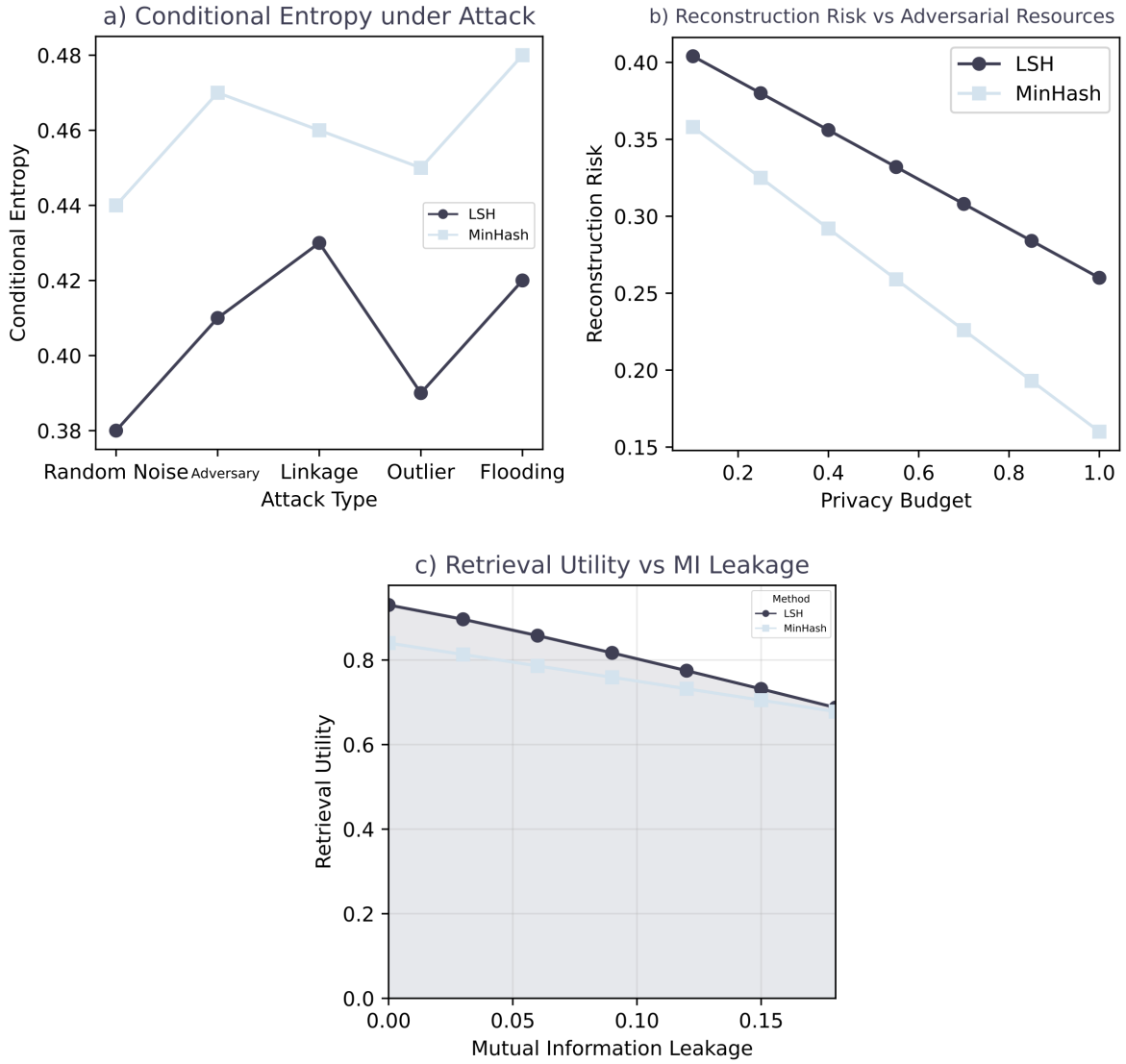

Figure 7. Privacy leakage and attack resistance analysis. (a) Conditional entropy under inference attack scenarios (b) Reconstruction risk as a function of adversarial resources (c) Retrieval utility versus mutual information leakage under variable privacy budgets

5. Conclusion and Limitations

In this research, the LSH and MinHash algorithms for large-scale privacy-preserving similarity search are systematically studied, benchmarked, and their theoretical dependability and experimental viability are demonstrated. The aforementioned research indicates that when the objective is to maximize retrieval accuracy and minimize latency and privacy is not a major concern, LSH performs effectively in a dense, high-dimensional feature space. The characteristics and the amount of the noise were shown to have a substantial impact because the addition of noise-based privacy protections to LSH resulted in a considerable fall in retrieval accuracy. In contrast, MinHash has continuously demonstrated good stability in sparse, transactional, and set-based contexts. It also retains high performance and stability even with the introduction of severe privacy protection methods like cryptographic masking or randomization.

The study establishes distinct underpinnings for the two approaches' boundary conditions. When high system throughput is required, LSH is ideal for quick, locality-aware queries in well-separated metric spaces that don't need strong privacy assurances or provide low-level privacy control. MinHash is better suited for scenarios requiring semantic group retrieval, duplicate detection, or set membership estimation. It is also favored where robust privacy guarantees or resistance to sophisticated inference and reconstruction attacks are required. The aforementioned empirical findings demonstrate that selecting the best algorithm is essentially context-dependent; neither is appropriate in every scenario, and real-world implementation must concurrently take data properties, privacy restrictions, and retrieval goals into account. Over-conservative parameterization decreases the system's overall usefulness without improving protection; enhanced privacy limitations have been found to have a rising marginal advantage.

The development of all-weather, privacy-preserving similarity-search algorithms still faces numerous unresolved issues, despite some positive outcomes. The high sensitivity of the parameters in the existing schemes, the changing landscape of collusive and hostile threats, and the added complexity brought about by federated and diverse data sources are some notable shortcomings. New developments in the design of context-aware and adaptable algorithms, principled privacy-risk quantification based on empirical data, and the combination of hashing with machine learning-driven representations will be required to solve the aforementioned issues. In order to satisfy the demands of new legal standards, scalable protocols and dynamic privacy adjustment will be investigated in the future. As a result, the next generation of intelligent information systems will need to concurrently optimize data utility and privacy.

Author Contributions

Stjepan Novak contributes to conceptualization, methodology, software, validation, analysis, investigation, data collection, draft preparation, manuscript editing, visualization. Antonio Matošević contributes to software, validation, analysis, investigation, data collection. All authors have read and agreed with the manuscript before its submission and publication.

Funding

This research received no specific financial support from any funding agency.

Institutional Review Board Statement

Not applicable.

References

[1] Funde, S., & Swain, G. (2022). Big data privacy and security using abundant data recovery techniques and data obliviousness methodologies. IEEE Access, 10, 105458-105484. https://doi.org/10.1109/ACCESS.2022.3211304

[2] Wang, N., Zhou, W., Wang, J., Guo, Y., Fu, J., & Liu, J. (2024). Secure and efficient similarity retrieval in cloud computing based on homomorphic encryption. IEEE Transactions on Information Forensics and Security, 19, 2454-2469. https://doi.org/10.1109/TIFS.2024.3350909

[3] Fernandes, N., Kawamoto, Y., & Murakami, T. (2021, October). Locality sensitive hashing with extended differential privacy. In European symposium on research in computer security (pp. 563-583). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-88428-4_28

[4] Lin, W., Cui, H., Li, B., & Wang, C. (2020). Privacy-preserving similarity search with efficient updates in distributed key-value stores. IEEE Transactions on Parallel and Distributed Systems, 32(5), 1072-1084. https://doi.org/10.1109/TPDS.2020.3042695

[5] Liu, T. (2024, July). Research on privacy techniques based on multi-party secure computation. In 2024 3rd International Conference on Artificial Intelligence and Autonomous Robot Systems (AIARS) (pp. 912-917). IEEE. https://doi.org/10.1109/AIARS63200.2024.00171

[6] Wu, J., Shen, L., & Liu, L. (2020). LSH-based distributed similarity indexing with load balancing in high-dimensional space: J. Wu et al. The Journal of Supercomputing, 76(1), 636-665. https://doi.org/10.1007/s11227-019-03047-6

[7] Liu, Y., Zhang, B., Ma, Y., Ma, Z., & Wu, Z. (2023). IPrivJoin: An ID-private data join framework for privacy-preserving machine learning. IEEE Transactions on Information Forensics and Security, 18, 4300-4312. https://doi.org/10.1109/TIFS.2023.3288455

[8] Kalia, P., Bansal, D., & Sofat, S. (2021). Privacy preservation in cloud computing using randomized encoding. Wireless Personal Communications, 120(4), 2847-2859. https://doi.org/10.1007/s11277-021-08588-9

[9] Aumüller, M., Bourgeat, A., & Schmurr, J. (2020, September). Differentially private sketches for jaccard similarity estimation. In International Conference on Similarity Search and Applications (pp. 18-32). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-60936-8_2

[10] Chen, X., Liu, H., & Yang, D. (2019). Improved LSH for privacy-aware and robust recommender system with sparse data in edge environment. EURASIP Journal on Wireless Communications and Networking, 2019(1), 171. https://doi.org/10.1186/s13638-019-1478-1

[11] Zhang, S., Ray, S., Lu, R., Guan, Y., Zheng, Y., & Shao, J. (2022). Efficient and privacy-preserving spatial keyword similarity query over encrypted data. IEEE Transactions on Dependable and Secure Computing, 20(5), 3770-3786. https://doi.org/10.1109/TDSC.2022.3227141

[12] Ding, X., Li, G., Yuan, L., Zhang, L., & Rong, Q. (2023). Efficient federated item similarity model for privacy-preserving recommendation. Information Processing & Management, 60(5), 103470. https://doi.org/10.1016/j.ipm.2023.103470

[13] Hamdi, H., Brahmi, Z., Alaerjan, A. S., & Mhamdi, L. (2023). Enhancing security and privacy preservation of sensitive information in e-Health datasets using FCA approach. IEEE Access, 11, 62591-62604. https://doi.org/10.1109/ACCESS.2023.3285407

[14] Al Sibahee, M. A., Abdulsada, A. I., Abduljabbar, Z. A., Ma, J., Nyangaresi, V. O., & Umran, S. M. (2021). Lightweight, secure, similar-document retrieval over encrypted data. Applied Sciences, 11(24), 12040. https://doi.org/10.3390/app112412040

[15] Kong, L., Wang, L., Gong, W., Yan, C., Duan, Y., & Qi, L. (2022). LSH-aware multitype health data prediction with privacy preservation in edge environment. World Wide Web, 25(5), 1793-1808. https://doi.org/10.1007/s11280-021-00941-z

[16] Kolajo, T., Daramola, O., & Adebiyi, A. (2019). Big data stream analysis: a systematic literature review. Journal of Big Data, 6(1), 1-30. https://doi.org/10.1186/s40537-019-0210-7

[17] Martínez, S., Gérard, S., & Cabot, J. (2022). Efficient model similarity estimation with robust hashing. Software and Systems Modeling, 21(1), 337-361. https://doi.org/10.1007/s10270-021-00915-9

[18] Majeed, A., Khan, S., & Hwang, S. O. (2023). Towards optimization of privacy-utility trade-off using similarity and diversity-based clustering. IEEE Transactions on Emerging Topics in Computing, 12(1), 368-385. https://doi.org/10.1109/TETC.2023.3258528

[19] Gkoulalas-Divanis, A., Vatsalan, D., Karapiperis, D., & Kantarcioglu, M. (2021). Modern privacy-preserving record linkage techniques: An overview. IEEE Transactions on Information Forensics and Security, 16, 4966-4987. https://doi.org/10.1109/TIFS.2021.3114026

[20] Khan, S., Abbas, H., & Iqbal, W. (2024). Verifiable privacy-preserving image retrieval in multi-owner multi-user settings. IEEE Transactions on Emerging Topics in Computational Intelligence, 8(2), 1640-1655. https://doi.org/10.1109/TETCI.2024.3353612

[21] Yuan, X., Zhang, Z., Wang, X., & Wu, L. (2023). Semantic-aware adversarial training for reliable deep hashing retrieval. IEEE Transactions on Information Forensics and Security, 18, 4681-4694. https://doi.org/10.1109/TIFS.2023.3297791

[22] Wu, W., Xian, M., Parampalli, U., & Lu, B. (2021). Efficient privacy-preserving frequent itemset query over semantically secure encrypted cloud database. World Wide Web, 24(2), 607-629. https://doi.org/10.1007/s11280-021-00863-w

[23] Guo, C., Liu, W., Liu, X., & Zhang, Y. (2020). Secure similarity search over encrypted non-uniform datasets. IEEE Transactions on Cloud Computing, 10(3), 2102-2117. https://doi.org/10.1109/TCC.2020.3000233

[24] Hu, J., Zhao, Y., Tan, B. H. M., Aung, K. M. M., & Wang, H. (2024). Enabling threshold functionality for private set intersection protocols in cloud computing. IEEE Transactions on Information Forensics and Security, 19, 6184-6196. https://doi.org/10.1109/TIFS.2024.3402355

[25] Ali, H. S., Elhefnawy, E. I., & Abo-Zahhad, M. (2024). Cancelable palmprint: intelligent framework toward secure and privacy-aware recognition system. EURASIP Journal on Information Security, 2024(1), 31. https://doi.org/10.1186/s13635-024-00179-y

[26] Wang, N., Zhou, W., Han, Q., Liu, J., Liao, W., & Fu, J. (2024). A lightweight privacy-preserving ciphertext retrieval scheme based on edge computing. IEEE Transactions on Cloud Computing, 12(4), 1273-1290. https://doi.org/10.1109/TCC.2024.3461732

[27] Liu, P., Li, X., Zang, B., & Diao, G. (2024). Privacy-preserving sports data fusion and prediction with smart devices in distributed environment. Journal of Cloud Computing, 13(1), 106. https://doi.org/10.1186/s13677-024-00671-3

[28] Zhou, Q., Lai, C., Guo, Q., Ma, H., & Zheng, D. (2022). A novel privacy protection scheme for internet of things based on blockchain and privacy set intersection technique. Journal of Cloud Computing, 11(1), 93. https://doi.org/10.1186/s13677-022-00375-6

[29] Shen, X., Wang, L., Pei, Q., Liu, Y., & Li, M. (2021). Location privacy-preserving in online taxi-hailing services. Peer-to-Peer Networking and Applications, 14(1), 69-81. https://doi.org/10.1007/s12083-020-00982-7

[30] Diamantini, C., Potena, D., & Storti, E. (2022, August). A knowledge-based approach to support analytic query answering in semantic data lakes. In European Conference on Advances in Databases and Information Systems (pp. 179-192). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-031-15740-0_14

[31] Arunakumari, B. N., Swain, V., & Sharan, K. (2024, June). Efficient Similarity Search Algorithms for Large Datasets. In 2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT) (pp. 1-7). IEEE. https://doi.org/10.1109/ICCCNT61001.2024.10724782

[32] Qin, J., Li, H., Xiang, X., Tan, Y., Pan, W., Ma, W., & Xiong, N. N. (2019). An encrypted image retrieval method based on Harris corner optimization and LSH in cloud computing. IEEE Access, 7, 24626-24633. https://doi.org/10.1109/ACCESS.2019.2894673

[33] Wang, Y., Miao, M., Shen, J., & Wang, J. (2019). Towards efficient privacy-preserving encrypted image search in cloud computing. Soft Computing, 23(6), 2101-2112. https://doi.org/10.1007/s00500-017-2927-6

[34] Wu, W., Li, B., Chen, L., Gao, J., & Zhang, C. (2020). A review for weighted minhash algorithms. IEEE Transactions on Knowledge and Data Engineering, 34(6), 2553-2573. https://doi.org/10.1109/TKDE.2020.3021067

[35] Eleni, P. (2023, May). Towards a secure and privacy compliant framework for educational data mining. In International Conference on Research Challenges in Information Science (pp. 534-541). Cham: Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-33080-3_35

[36] Zhou, Z., Wang, Y., Cong, L., Song, Y., Li, T., Li, M., ... & Lv, C. (2024). Enhancing Data Privacy Protection and Feature Extraction in Secure Computing Using a Hash Tree and Skip Attention Mechanism. Applied Sciences, 14(22), 10687. https://doi.org/10.3390/app142210687

[37] Xu, J., Li, X., Wang, H., Dai, H. N., & Meng, S. (2020, October). Lsh-based collaborative recommendation method with privacy-preservation. In 2020 IEEE 13th International Conference on Cloud Computing (CLOUD) (pp. 566-573). IEEE. https://doi.org/10.1109/CLOUD49709.2020.00085

[38] Li, X., Lei, W., Tang, W., Wang, Y., Yang, X., & Liao, X. (2024). Segmented hash-based privacy-preserving image retrieval scheme in cloud-assisted iot. IEEE Internet of Things Journal, 11(21), 35250-35265. https://doi.org/10.1109/JIOT.2024.3438085

[39] Xie, T., Yuan, L., Zhang, Q., Wu, J., & Ren, F. (2024). Ciphertext Fuzzy Retrieval Mechanism with Bidirectional Verification and Privacy Protection. IEEE Internet of Things Journal, 11(24), 41061-41083. https://doi.org/10.1109/JIOT.2024.3458457

[40] Wang, W., Jin, Y., & Cao, B. (2022, August). An efficient and privacy-preserving range query over encrypted cloud data. In 2022 19th Annual International Conference on Privacy, Security & Trust (PST) (pp. 1-10). IEEE. https://doi.org/10.1109/PST55820.2022.9851989