Turkish Journal of Computer and Mathematics Education Vol.12 No.13 (2021), 5913-5921

Research Article

A Study on Privacy Preserving Technology Using Min-Hash Based Similarity Measurement Method

Dr.Ho-Kyung Yang [a], Dr.Hyun-Jong Cha [b], Dr.You-Jin Song [c*]

[a] Division of Information Technology Education, Sunmoon University, KOREA
[b] Department of Multimedia Science, Chungwoon University, KOREA
[c] Department of Information Management, Dongguk University, KOREA
Corresponding Author: You-Jin Song, **email:** song@dongguk.ac.kr

Abstract: Recently, Traditional Methods of measuring the similarity have been time-consuming and costly as the size and area of data increase. Border proposed a Min-Hash efficiently estimates the similarity between two signatures represented two-sets as the connotated form. Min-Hash is widely used in plagiarism prevention, graph and image analysis, and genetic analysis. However, raw data is encrypted but exposure to keys due to frequent use of keys for decryption poses security challenges. In particular, exposure to data about users at large sites such as Facebook and Amazon causes serious damage. More recently, studies of new fourth-generation encryption technologies that can protect user-related data without using the keys needed for encryption have drawn attention. Also, data clustering technology that uses encryption is drawing attention. Thus, among the various clustering methods, this paper presents model using Rushell and Rao similarity for preserving privacy by using RSA homomorphic encryption and estimates efficiently it by using Min-Hash.

Keywords: Similarity Measurement, Jaccard Similarity, MinHash, Homomorphic Encryption, private MinHash, privacy preserving

1. Introduction

Clustering in data mining is a method of classifying objects with similar characteristics into the same class **(Min and Heo, 2014; Lee, 2019)**. There are several methods for determining the similarity of objects, depending on the type of attribute value that the object has. Among them, Jaccard Similarity is a typical method for determining the similarity of objects whose attribute values can be shown based on a set. Jacquard similarity is a method of measuring similarity by relatively evaluating the intersection between different sets. Jacquard similarity is in various fields such as collaborative filtering **(Chum, Philbin and Zisserman, 2008)**, Group Technology **(Seifoddini, 1989; Yin and Yasuda, 2005)**, enterprise decision making through Enterprise Grid **(Rahman, Hassan and Buyya, 2010; Niwattanakul, singthongchai, Naenudorn and Wanapu, 2013)**, search engine **(Niwattanakul, singthongchai, Naenudorn and Wanapu, 2013)**, keyword comparison **(Bank and Cole, 2008)**, etc. It is used in **(Lee, 2017; Jung, 2020)**.

As areas of representation of data become diverse and storageable, an era is approaching in which numerous objects that exist in the real world can be expressed as data **(Hahm and Chen, 2020)**. As a result, applying similarities to objects to traditional methods can only afford the amount of data and computational time. Min-Hash **(Seifoddini, 1989)**, proposed by Broder, is a kind of locality Sensitive Hashing (LSH) technique that allows the form of a set to be connotated, such as a signature, and gives an approximate estimate of the similarity of the sets. Min-Hash is applied in plagiarism prevention **(Seifoddini, 1989; Yin and Yasuda, 2005)**, graphs **(Rahman, Hassan and Buyya, 2010; Niwattanakul, Singthongchai, Naenudorn and Wanapu, 2013)**, and image analysis **(Bank and Cole, 2008; Broder, 1997)**, and genetic analysis **(Border, 2000)**, and is used as an efficient way to measure similarities between data in many areas that can be represented by data.

Min-Hash represents the smallest value when two sets of elements are recorded in a particular hash function and is a method that can be used to approximate similarities. The smallest value from Min-Hash is described as Min-Hash Value and the Min-Hash Value for Set A is expressed as hmin(A). The probability that the two sets have the same Min-Hash Value as the two sets of Jaccard similarity.

While encryption has been essential to data security in recent years, traditional encryption technology does not fully protect user-related data due to the exposure of keys due to frequent use of keys. In particular, exposure to data about users at large sites such as Facebook and Amazon causes serious damage. Existing encryption only showed

the cryptogram as a string indistinguishable from the random number, and it was impossible to perform a meaningful operation on the cryptogram itself. Thus, the homomorphic encryption proposed by Rivest, Addleman and Dertouzous (1978) was first proposed a method to perform several operations without a decode key, even when the plain text is encrypted. In addition, Gentry (2009) designed a fullly homomorphic encryption based on the challenges of number theory and lattice theory, enabling computers to perform all computations of addition, subtraction, multiplication, and division of ciphertext. Recently, it has been applied not only to search and statistical analyses, but also to highly complex computations such as machine learning and image processing in HEAN, and was selected as one of the top 10 Emering Technology in the 2011 MIT Technical Review **(Teixeira, Silva and Meira, 2012)**. In data clustering, K-means clustering technology that utilizes this technology to apply encryption has recently drawn attention **(Aksakalli and Welke, 2016)**.

"Homorphic" in homomorphic encryption refers to the idea of preserving the computation between two algebraic structures of the same type, which is commonly addressed in mathematics. In other words, a homomorphic encryption is a cryptographic system that preserves certain operations, such as addition and multiplication, with responding elements of plaintext to elements of ciphertext **(Teixeira, Silva and Meira, 2012)**. A homomorphic encryption that can only perform some operations of the same type data is called a "partial homomorphic encryption". It is commonly consist of additive homomorphic encryption and multiplicative homomorphic encryption.

Therefore, in this paper, efficient similarity measurement through Min-Hash can be applied to homomorphic encryption, which is one of the four generations of encryption technology, to maintain privacy. Present a degree analysis model. The structure of this study is as follows. Chapter 2 introduces background knowledge and related research, and Chapter 3 introduces Private Min-Hash to which the same type encryption technology as conventional Min-Hash is applied. Chapter 4 describes experiments and results to demonstrate the effectiveness of the method proposed in this paper. Chapter 5 describes the problems and conclusions that need to be resolved in the future.

2.Related Works

In this section, we introduce basic concepts of Similar measurements, Min-Hash, Hormorphic Encryption, and Customer Analysis System, which are core technologies needed for thesis.

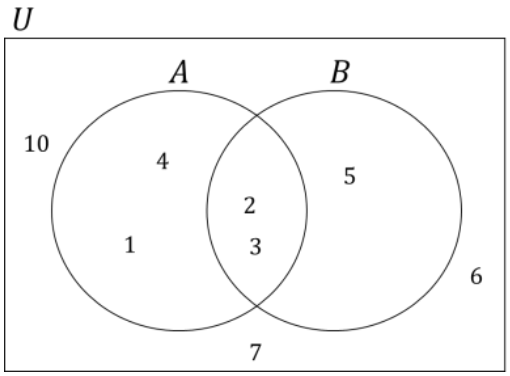
2.1.Similar measurements

Jaccard Similarity is a method of measuring the similarity of two objects represented in a set by calculating the relative magnitude of the intersection with respect to the size of the union of both sets. It can show the degree of similarity (Jaccard and Welke, 2016).

$$Jaccard(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

For example, when two sets A = {1,2,3,4} and B = {2,3,5} exist, the jacquard similarity between sets A and B is shown in Figure 1.

Figure.1 Jaccard Similarity



$Jaccard(A,B) = \frac{2}{5}$

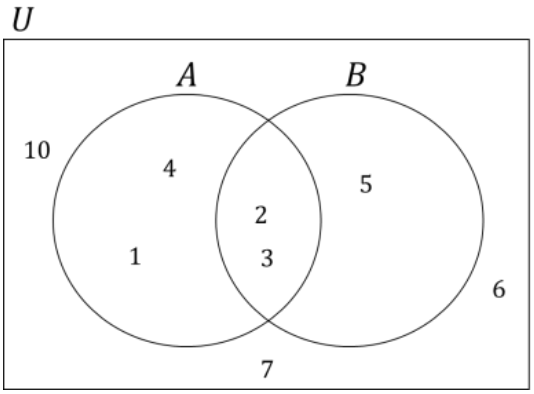
Russell-Lao similarity is similar to jacquard similarity. The method of measuring two object similarity differs from jacquard similarity in that it is the size of the complete set rather than the size of the union of the two sets.

Calculate the relative magnitude of the intersections relative to represent the similarity of both sets (Russell and Rao, 1940).

$$Rusrao(A,B) = \frac{|A \cap B|}{|U|}$$

For example, when two sets A = {1,2,3,4} and B = {2,3,5} exist, the Russell-Lao similarity of sets A and B is the same as in Figure 2.

Figure.2 Russell and Rao Similarity



$$RusRao(A,B) = \frac{2}{8}$$

The method often used to measure similarity between Jacquard-like and Russell-Lao-like is similar to Jacquard, but methods using Russell-Lao-like similarity are also being studied **(Sneath and Sokal, 1973)**. There are various methods for measuring the similarity between the two sets, as in **(Choi, Cha and Tappert, 2010)**, and the similarity can be measured in various ways depending on the situation. In this paper, I would like to calculate the similarity between two objects based on the similarity between Russell and Lao.

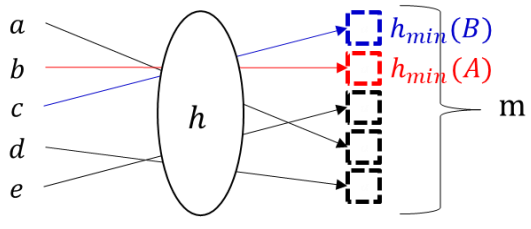
2.2.Min-Hash

Min-Hash is a method that shows the smallest result value when two sets of elements are put into a specific hash function in history, and can be used to approximate the similarity. The smallest result value that comes out via Min-Hash is expressed as Min-Hash Value, and the Min-Hash Value of the set A is shown in *hmin* (A). The probability that the two sets of comparison targets A and B have the same Min-Hash Value is the same as the two sets of jacquard similarity **(Chum, Philbin and Zisserman, 2008; Lee, Ke and Isard, 2010; Koslicki and Zabeiti, 2019)**.

$$Pr[\, h_{min}(A) = h_{min}(B)\,] = Jaccard(A,B)$$

The hash function h that is typically used to obtain the Min-Hash Value of a set of m elements is $ax + b \bmod p$. a and b are arbitrary natural numbers, and p is the smallest prime number greater than or equal to m.

Figure.3 Min-Hash



$$h : ax + b \,\%\, p$$

$$h_{min}(A) : \min\,(h(x)) \quad (x \in \mathrm{A})$$

$$h_{min}(B) : \min\,(h(x)) \quad (x \in \mathrm{B})$$

Figure 3 shows the contents of Min-Hash.

When there are n Min-Hash Values for a set A via n Min-Hash, it can be expressed in the form of a vector and expressed by Min-Hash Signature. A SigA with a Min-Hash Signature in the set A can be expressed as:

$$Sig_A = \left[h_{min_1}(\mathrm{A}), h_{min_2}(\mathrm{A}), \cdots h_{min_n}(\mathrm{A}) \right]$$

When the kth Min-Hash Value is the same value in the Min-Hash Signature of the sets *A* and *B*, the weight of similarity can be expressed as follows.

$$\widehat{Jaccard}_k(A,B) \begin{cases} 1 & h_{min_k}(A) = h_{min_k}(B) \\ 0 & h_{min_k}(A) \neq h_{min_k}(B) \end{cases}$$

Therefore, *n* hash functions can be used to calculate the similarity of the generated Min-Hash Signatures of *A* and *B* to approximate the similarity of sets *A* and *B*.

$$Jaccard(A,B) \fallingdotseq \frac{1}{k}\sum_{k=1}^{n} \widehat{Jaccard}_k(A,B)$$

2.3.Homomorphic Encryption

Homomorphic, which means homomorphism in homomorphic encryption, comes from homomorphism, which is often used in mathematics, and refers to an event (map) that maintains operations between two units of the same type. In other words, homomorphic encryption is a password system that saves specific operations such as addition and multiplication with the idea of associating the elements of plaintext space with the elements of cryptographic space. Among the homomorphic encryptions, the password that can be executed by only some operations is called partial homomorphic encryption. Elgamal passwords on a finite body can only be multiplied. Multiplicative homomorphic encryption **(Tsiounis and Yung, 1998)**. There is additive homomorphic encryption **(Pan, Sun and Fang, 2011)** that protects only the addition.

2.4.Customer Analysis System

When designing models that utilize similarities such as customer Gunjipfa, recommender systems have raised security issues for many developers in relation to customer information. Therefore, research has been conducted to address potential security issues when leveraging user information **(Ramos, 2003; Han, Kamber and Pei, 2011; Broder, 1997)**.

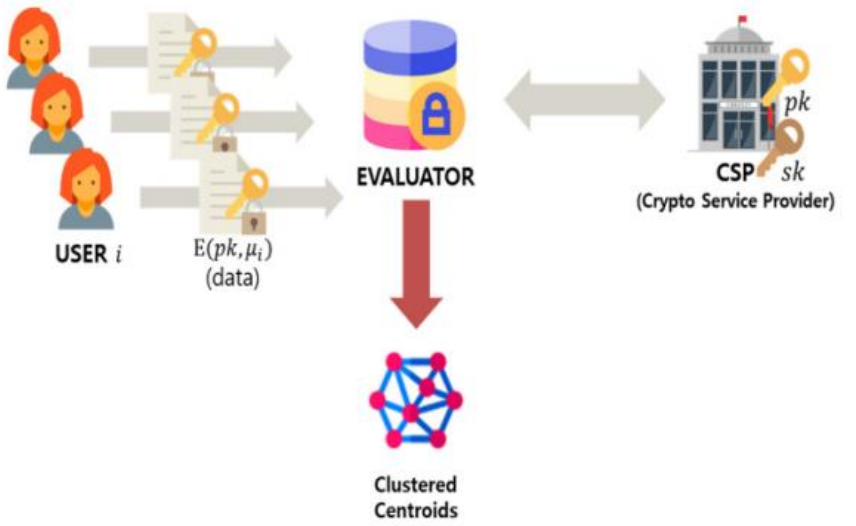
Most research directions are largely in the use of information between users, or through a third trust institution. Therefore, before using this information to calculate the similarity between users, the relevant data must always be converted into a format that is either encrypted or unknown **(Broder, 1997; Murthy, 2012; Syropoulos, 2000)**.

Algggan, Gambs and Kermarrec (2011) proposed a method that can calculate the degree of similarity in a state where the value of the result is converted into a format that cannot be known by using the differential privacy policy. In particular, utilizing the calculations of Laplace, Scalar Product, and Cosine Similarity **(Alaggan, Gambs and Kermarrec, 2011)**, added Laplacian Noise based on the calculation method of homologous cryptography. Therefore, it was possible to calculate the similarity with the encrypted data. After that, Wong and Kim(2014) proposed by et al., The similarity of data, which is the form of the proposed specific binary vector, was applied to homomorphic encryption

In addition, recently, research on applying Fully Homomorphic Encryption to k-means clustering technology has become active, and it is expanding further in fields such as market analysis and medical research (Alaggan, Gambs and Kermarrec, 2011; Jeong, Kim and Lee, 2018; Almutairi, Coenen and Dures, 2017).

Figure 4 shows the concept of k-means clustering technology.

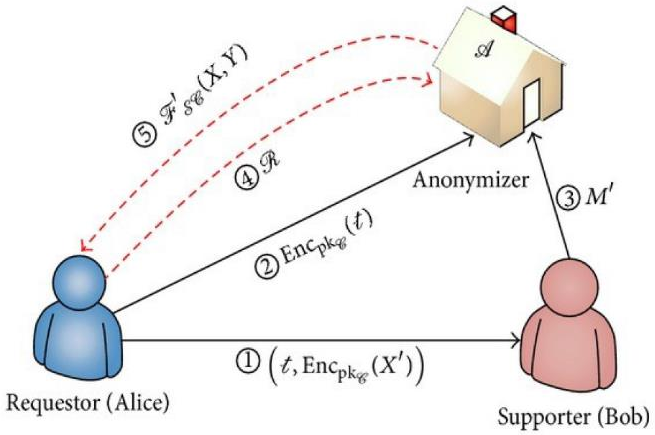
Figure.4 Privacy-preserving K-means Clustering



3.Proposed Private MinHash

3.1.Basic Structure of the System

Figure.5 Homomorphic Encryption Scheme



Russell-Lao similarity comparison system, which basically applies multiplication encryption, was constructed as shown in Figure 5.

- Anonymizer : The user is performing the requested calculation, but the user's information cannot be grasped.

- Anonymizer : The user is performing the requested calculation, but the user's information cannot be grasped.

- Supporter : Requestor provides information to Annoymizer for the requested information.

Basically, Annonymizer performs operations on homomorphic encrypted data, and Alice (Requestor) requests a query. The person who provides the data corresponding to the query is made up of Bob (Supporter). For example, assuming Alice has data *{2,3}* and Bob has *{1,2}*, Alice has any decimal value to convert her data to a Binary Set. Determine t. Then, from the complete data, only the data that you have that is unlikely to change to 10,000 tons is processed by t-1. After that, the t value and data are encrypted with the public key provided by Anonymizer. Once the encryption is complete, Anonymizer will be provided with the encrypted t-value, and Bob will provide the t-value and Alice's encrypted data. Bob is similar in method to Alice, but when converting his data to a Binary Set, he processes the data he has with t2 and the other data with t. Finally, after representing the encrypted data M via multiplication with the encrypted data received from Alice, the requested query without providing information about Alice and Bob's data to Anonymizer. Provides M` with randomly shuffled M data so that can be executed.

Figure 6 shows the requester and supporter dataset.

Figure.6 Alice & Bob Dataset

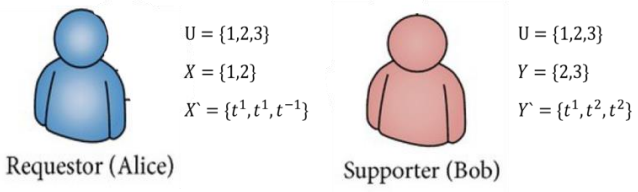


Figure.7 Anonymizer

Sig`Alice \ Sig`Bob	t^1(hasn't)	t^2(has)
t^1(has)	t^2	t^3
t^{-1}(hasn't)	t^0	t^1

t^1, t^2 : mismatch
t^0 : negative match
t^3 : positive match

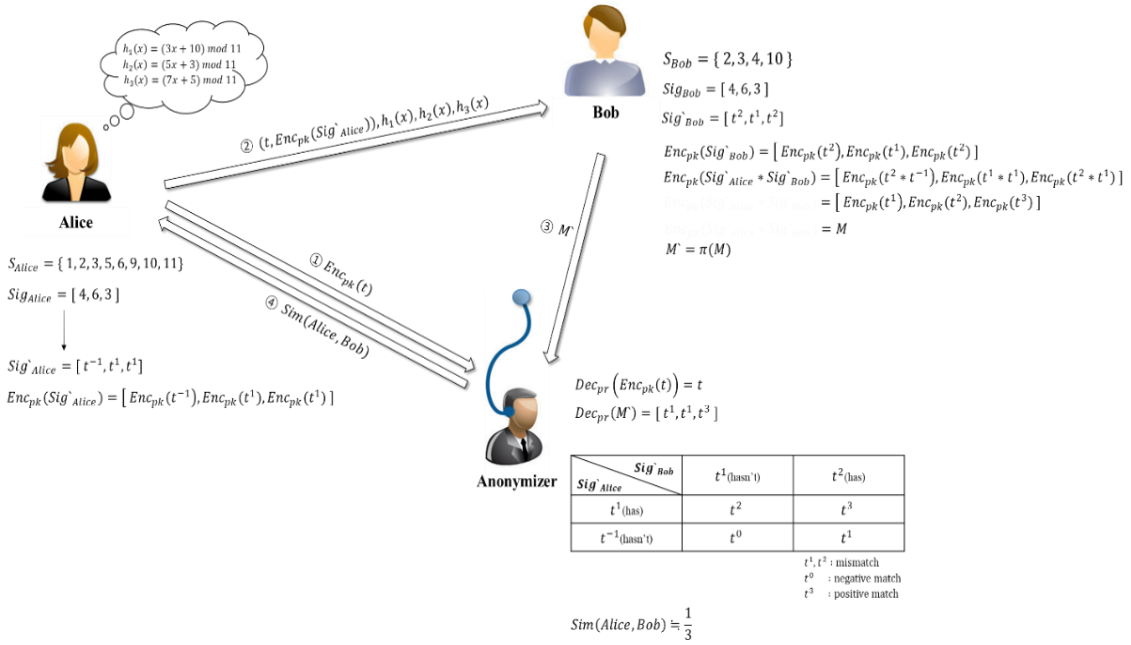
$$Enc_{pk\&}(M`) \Rightarrow \{t^1, t^2, t^3\} \Rightarrow Sim(X`,Y`) = \frac{1}{3} \Rightarrow Sim`(X`,Y`) = \frac{1}{3} + Lap(\Delta F/\varepsilon)$$

After receiving the encrypted t-value and M` from Alice and Bob, Anonymizer calculates the similarity as shown in Figure 7, which attempts to decrypt its own private key and converts it into plaintext data. Then, the data to be transmitted to Alice generates an arbitrary noise value according to the confidentiality (Sensitivity) of the data, inserts it into the calculated similarity value, and transmits it. By generating and adding random noise values, no one can know exactly about the actual similarity between Alice and Bob's data while the similarity is calculated. In this paper, Min-Hash is utilized in the process of (Han, Kamber and Pei, 2011) introduced, and the procedure for making the similarity calculation efficient is applied, and the calculated similarity value is an approximate value. Therefore, it is possible to immediately convey the calculated value to Alice without adding a random noise value.

3.2.Similarity Measurement Model

In this paper, we utilize Min-Hash in (Syropoulos, 2000) utilizing the data in the set to apply efficient similarity measurement of a large set. The overall configuration procedure consists of four stages as shown in Figure 8.

Figure.8 Similarity Measurement Model



Step 1: First, Alice sets an arbitrary minority t value. Then, an arbitrary hash function is defined to obtain a complete set of Min-Hash Values. If Alice has the specified hash function Min-Hash Value, it converts it to t, otherwise it converts it to t^{-1}. Alice who generated k hash functions will have a signature $Enc_{pk}(SigAlice)$ indicating *k* Min-Hash Values. Anonymizer is encrypted with the provided public key and tells Bob the *t*-value and signature $Enc_{pk}(t, Sig`_{Alice})$ like a hash function. Anonymizer provides an encrypted *t*-value.

Step 2: Bob creates a signature via the hash function provided in ②, and then creates and encrypts the signature in the same way as Alice. It then multiplies the values of the two encrypted signatures to generate a new signature *M*. When passing to Anonymizer, it provides data *M'* that randomly shuffles the data belonging to *M* to execute the query when it does not provide the information of *M* data.

Step 3: ③ Anonymizer decrypts the *M'* provided to Bob with his own private key to obtain $Dec_{pr}(M')$. Anonymizer saves the data corresponding to t3 in the data in the Score set.

$Dec_{pr}(M') = [\ t^1, t^1, t^3\ ]$

$Score = \{\ x \mid x \in Dec_{pr}(M`)\ and\ x = t^3\ \}$

The Boolean Table shown in Figure 9 determines the type of similarity for the four variables. The element that both objects have in common is a positive match. In this paper, this variable is represented by t^3. Elements that have only one of both objects are represented by t^1 and t^2 in mismatch. Finally, elements that both objects do not have are represented by t^0 in a negative match.

Figure.9 Boolean Table

$Sig`_{Alice}$ \ $Sig`_{Bob}$	t^1(hasn't)	t^2(has)
t^1(has)	t^2	t^3
t^{-1}(hasn't)	t^0	t^1

t^1, t^2 : mismatch
t^0 : negative match
t^3 : positive match

Alice and Bob similarity Sim (Alice, Bob) calculated through the signature transmitted by Anonymizer is calculated via the number of four variables Alice and Bob similarity is calculated as follows.

$$Sim(Alice, Bob) = \frac{|Score|}{|Dec_{pr}(M`)|}$$

Step 4 : Finally, Alice is provided with an approximation of the similarity calculated in ④ to Anonymizer.

4.Conclusion

In this paper, we introduced a customer segmentation model for measuring the similarity of data in smart devices that maintain privacy by utilizing homomorphic encryption and Min-Hash. Through experiments, Gunzipfa introduced in this paper was found to be more efficient in terms of speed, although it is not superior to clustering using the previous encryption in terms of clustering quality. Therefore, it can be applied to a model for real-time similar analysis that ensures customer privacy.

Acknowledgments

This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2019R1F1A1056507).

References

A. Min, H., & Heo, J. (2014). A Clustering Scheme Considering the Structural Similarity of Metadata in Smartphone Sensing System. The Journal of The Institute of Internet, Broadcasting and Communication, 14(6), 229-234.
B. Lee, J. (2019). A Study on Research Trend Analysis and Topic Class Prediction of Digital Transformation using Text Mining. International journal of advanced smart convergence, 8(2), 183-190.
C. Seifoddini, H. (1989). A note on the similarity coefficient method and the problem of improper machine assignment in group technology applications. The international journal of production research, 27(7), 1161-1165.
D. Yin, Y., & Yasuda, K. (2005). Similarity coefficient methods applied to the cell formation problem: a comparative investigation. Computers & industrial engineering, 48(3), 471-489.
E. Rahman, M., Hassan, M. R., & Buyya, R. (2010). Jaccard index based availability prediction in enterprise grids. Procedia Computer Science, 1(1), 2707-2716.
F. Niwattanakul, S., Singthongchai, J., Naenudorn, E., & Wanapu, S. (2013, March). Using of Jaccard coefficient for keywords similarity. In Proceedings of the international multiconference of engineers and computer scientists (Vol. 1, No. 6, pp. 380-384).
G. Bank, J., & Cole, B. (2008). Calculating the jaccard similarity coefficient with map reduce for entity pairs in wikipedia. Wikipedia Similarity Team, 1-18.
H. Lee, S. (2017). A Study on the Need of the Usable Security in the Corelation between IT Security and User Experience. International Journal of Internet, Broadcasting and Communication, 9(4), 14-18.
I. Jung, S. M. (2020). Image Watermarking Algorithm using Spatial Encryption. The Journal of the Convergence on Culture Technology, 6(1), 485-488.
J. Hahm, S., & Chen, L. (2020). The Role of Professors' Intellectual Stimulation for Intellectual Growth among Chinese Students Who Study in Korea: The Moderating Effect of Growth Need Strength. International Journal of Advanced Culture Technology, 8(3), 45-53.
K. Broder, A. Z. (1997, June). On the resemblance and containment of documents. In Proceedings. Compression and Complexity of SEQUENCES 1997 (Cat. No. 97TB100171) (pp. 21-29). IEEE.
L. Broder, A. Z. (2000, June). Identifying and filtering near-duplicate documents. In Annual Symposium on Combinatorial Pattern Matching (pp. 1-10). Springer, Berlin, Heidelberg.
M. Teixeira, C. H., Silva, A., & Meira Jr, W. (2012). Min-hash fingerprints for graph kernels: A trade-off among accuracy, efficiency, and compression. Journal of Information and Data Management, 3(3), 227-242.
N. Aksakalli, C. G., & Welke, P. (2016). Minhashing for Graph Similarity Computation. Proceedings of the 3rd CSCUBS.
O. Jaccard, P. (1908). Nouvelles recherches sur la distribution florale. Bull. Soc. Vaud. Sci. Nat., 44, 223-270.
P. Russell, P. F., & Rao, T. R. (1940). On habitat and association of species of anopheline larvae in south-eastern Madras. Journal of the Malaria Institute of India, 3(1).
Q. Sneath, P. H., & Sokal, R. R. (1973). Numerical taxonomy. The principles and practice of numerical classification, 12(5), 190-199.
R. Choi, S. S., Cha, S. H., & Tappert, C. C. (2010). A survey of binary similarity and distance measures. Journal of Systemics, Cybernetics and Informatics, 8(1), 43-48.
S. Chum, O., Philbin, J., & Zisserman, A. (2008, September). Near Duplicate Image Detection: min-Hash and tf-idf Weighting. In BMVC (Vol. 810, pp. 812-815).
T. Lee, D. C., Ke, Q., & Isard, M. (2010, September). Partition min-hash for partial duplicate image discovery. In European Conference on Computer Vision (pp. 648-662). Springer, Berlin, Heidelberg.
U. Koslicki, D., & Zabeti, H. (2019). Improving minhash via the containment index with applications to metagenomic analysis. Applied Mathematics and Computation, 354, 206-215.
V. Tsiounis, Y., & Yung, M. (1998, February). On the security of ElGamal based encryption. In International Workshop on Public Key Cryptography (pp. 117-134). Springer, Berlin, Heidelberg.
W. Pan, M., Sun, J., & Fang, Y. (2011). Purging the back-room dealing: Secure spectrum auction leveraging paillier cryptosystem. IEEE Journal on Selected Areas in Communications, 29(4), 866-876.
X. Ramos, J. (2003, December). Using tf-idf to determine word relevance in document queries. In Proceedings of the first instructional conference on machine learning (Vol. 242, pp. 133-142).
Y. Han, J., Kamber, M., & Pei, J. (2011). Data mining concepts and techniques third edition. The Morgan Kaufmann Series in Data Management Systems, 83-124.
Z. Broder, A. Z. (1997, June). On the resemblance and containment of documents. In Proceedings. Compression and Complexity of SEQUENCES 1997 (Cat. No. 97TB100171) (pp. 21-29). IEEE.
AA. Murthy, J. V. R. (2012). Clustering based on cosine similarity measure, International journal of engineering science & advanced technology, 2(3) 508-512.

BB. Syropoulos, A. (2000, August). Mathematics of multisets. In Workshop on Membrane Computing (pp. 347-358). Springer, Berlin, Heidelberg.
CC. Wong, K. S., & Kim, M. H. (2014). Preserving differential privacy for similarity measurement in smart environments. The Scientific World Journal, 2014.
DD. Alaggan, M., Gambs, S., & Kermarrec, A. M. (2011, December). Private similarity computation in distributed systems: from cryptography to differential privacy. In International Conference On Principles Of Distributed Systems (pp. 357-377). Springer, Berlin, Heidelberg.
EE. Jeong, Y., Kim, J. S., & Lee, D. H. (2018). Privacy-Preserving k-means Clustering of Encrypted Data. Journal of the Korea Institute of Information Security & Cryptology, 28(6), 1401-1414.
FF. Almutairi, N., Coenen, F., & Dures, K. (2017, August). K-means clustering using homomorphic encryption and an updatable distance matrix: secure third party data clustering with limited data owner interaction. In International Conference on Big Data Analytics and Knowledge Discovery (pp. 274-285). Springer, Cham.