ISSN: 2395-7639

International Journal of Multidisciplinary Research in Science, Engineering, Technology & Management (IJMRSETM)

(A Monthly, Peer Reviewed Online Journal)

Visit: www.ijmrsetm.com

Volume 6, Issue 10, October 2019

Reduction of Data Deduplication in Cloud Storage System Using Neural Network Based Machine Learning Method

Mrs.S.Ruba, Dr. A.M.Kalpana

Assistant Professor, Department Computer Science and Engineering, Government College of Engineering, Salem, India

Professor, Department of Computer Science and Engineering, Government College of Engineering, Salem, India

*Corresponding author

ABSTRACT: Deduplication is often used in storage systems in order to save storage space, communication bandwidth, write energy, and recovery and error-protection infrastructure. However, deduplication overhead increases latency and computation energy. Determining whether a data chunk is already stored by comparing signatures constitutes a significant fraction of this deduplication overhead. In this paper, proposed a streaming chunk incremental learning (SCIL-NN) based on a neural network. Our technique based on learning method for distributing incoming data with their multi-classes. A cloud storage data was analyzed with the hash index of cache value from the data chunk process. The hash functions are stored by hash value for further process of duplication analysis. The SCIL-NN adaptively process the chunking method for the neural network with incremental learning. The neural network partitioning the chunk according to the streaming process of data. And finally, proposed an AppDataDedup, is an effectiveapplication-aware distributed deduplication framework in cloud platform, to meet this challenge by exploiting data similarity to optimize deduplication process. AppDataDedup constructs application-aware similarity indices with super-chunk process to speed up the deduplication with high efficiency. In the experiments, we evaluate the classification performance for several large-scale cloud storage data sets to discuss the scalability of SCIL under one-pass incremental learning environments and evaluation of AppDataDedup against driven by real-world datasets demonstrates to achieves the highest global deduplication

KEYWORDS: Chunking, Deduplication, SCIL-NN algorithm, AppDataDedup, Machine Learning, Hash Pattern.

I. INTRODUCTION

With the advent of cloud computing in recent years, the data volumes in cloud are increasing significantly due to continued growth of internet, adoption of smartphones and social networking platforms. In 2011, International Data Corporation (IDC) reported that data volume created and copied in the world will be 35ZB by 2020 [1]. The enterprises are facing problems in storing and processing a large amount of data volumes. In order to enhance the reliability and availability and provide disaster recovery, data are generally duplicated on multiple storage locations. Most of these duplicated data exert an extra load on the storage system in terms of additional space and bandwidth to transfer the duplicated data on the network. An efficient data storage management is critical, data deduplication considered as supporting technology for the efficient big data storage [2]. Deduplication technique is a distinctive data compression technique to eliminate redundant data and reduce network transmission rate and storage space in the cloud storage systems [3,4]. The techniques find out the duplicate data, save only one copy of the data and strategically use logical pointers for duplicated data [5].

Deduplication addresses the growing demand for storage capacity [6]. Many cloud storage providers like Amazon S3, Bitcasa and Microsoft Azure [7] and backup services such as Dropbox and Memopal are employing data deduplication techniques [8] to improve storage efficiency. The deduplication techniques are data type specific, and different techniques are employed on different types of data such as text, image and video data. All three types of data have different storage formats and implicit characteristics. Based on type of data, deduplication techniques have different processes to find and remove duplicate information. So, type of data is important for the development of deduplication

ISSN: 2395-7639

International Journal of Multidisciplinary Research in Science, Engineering, Technology & Management (IJMRSETM)

(A Monthly, Peer Reviewed Online Journal)

Visit: www.ijmrsetm.com

Volume 6, Issue 10, October 2019

techniques. The format of information is critical for reading, finding and matching the information. Bit- level matching is required to find duplication in executable files. The techniques to check duplicates in text, image and video have different processes due to varied format of data. A least number of data replica called replication factor are maintained in a large distributed storage system to achieve high data availability. Any duplicate data above replication factor is removed to reduce storage requirement, storage cost, computation and energy. The predictive model like Classification, Regression, Time series analysis, Prediction etc., using the known values determine the unknown data [9]. Classification technique which comprises many decision-theoretic methods for recognizing data is a wide-ranging research field capable of processing a broader category of data than regression [10,11]. The classification algorithm constructs a model by learning from the training set. New objects are classified by using this model.

Due to these significant benefits to industry, deduplication techniques for a large distributed storage systems gained momentum in academia and industry. Still, these techniques are facing challenges due to efficiency and efficacy of data matching techniques. Numerous classification algorithms available are k-Nearest Neighbor (k-NN) algorithm, neural network, Naive Bayes, decision tree, Bayesian network, and Support Vector Machine (SVM). The exceedingly popular classification algorithm called k-NN exhibits good performance characteristics and it is used in several diverse applications [12-14] like 3- dimensional object rendering, content-based image retrieval, statistics (estimation of entropies and divergences), biology (gene classification).Data cleaning [15], also called data cleansing or scrubbing, enhance the quality of data by identifying and eradicating errors and inconsistencies from the large amount of data. It aims at enhancing the overall data compatibility by concentrating on eradication of changes in data contents and minimizing data repetition. Record duplicates, omitted values, record and field resemblances and duplicate eradications are detected by current data cleaning techniques. In detecting duplicate documents within a corpus, and in the filtering of search engine results, the approach was applied and the results obtained were hopeful. Detection of other or several records that signify one distinct real-world entity or object is performed by the duplicate record detection process.

II. RELATED WORK

Mikhail Bilenko et al., [16] The difficulty of duplicate detection is really to find whether the same real-world object is represented by two or more distinct database entries. Record linkage, object identification, record matching etc., are remarkable names for Duplicate detection. It is a greatly researched topic and has high importance in fields such as master data warehousing, data management, minimum and ETL (Extraction, Transformation and Loading), customer relationship management, and data integration. The two innate problems that must be addressed by duplicate detection are quick detection of all duplicates in large data sets (efficiency) and proper determination of duplicates and non-duplicates (effectiveness). The researchers in academia and industry are working to develop efficient distributed deduplication techniques by using the EL techniques. K.Deepa et al., [17] have proposed an approach to Duplicate Record Detection Using Similarity Metrics and ANFIS. They developed a domain independent approach to detect duplicate records presented in large databases. The approach adopts ANFIS and similarity functions to improve duplicate detection. The main aim of using ANFIS was to reduce the time taken for making decisions in detecting the duplicates. To minimize the number of record comparisons, an appropriate clustering method, known as K-means clustering was used in the duplicate detection phase. Their method was tested on the real-life datasets and the performance was evaluated with the evaluation metrics. They showed that their proposed approach detects duplicates efficiently and accurately.

Elhadi M et al., [18] have planned method that bring information on experiments performed to investigate the use of a combined part of speech (POS) and an improved longest common subsequence (LCS) in the analysis and calculation of similarity between texts. For the representation of documents, the text's syntactical structures were used. To compare and rank the documents according to the similarity of their representative string, an improved LCS algorithm was applied to such a representation. P. Christen et al., [19] In order to detect duplicate records more effectively, semantic similarity should be considered other than string similarity based on experimental results. They presented a survey of twelve variations of six indexing techniques. Their complexity was analyzed, and their performance and scalability were evaluated within an experimental framework using both synthetic and real data sets. They aimed at reducing the number of record pairs to be compared in the matching process by removing obvious non-matching pairs, while at the same time maintaining high matching quality. Gianni Costa et al., [20-22] have proposed an incremental technique for discovering duplicates in large databases of textual sequences, i.e., syntactically different tuples, that refer to the same real-world entity. Each newly arrived tuple was assigned to an appropriate cluster via nearest-neighbor classification.

ISSN: 2395-7639

International Journal of Multidisciplinary Research in Science, Engineering, Technology & Management (IJMRSETM)

(A Monthly, Peer Reviewed Online Journal)

Visit: www.ijmrsetm.com

Volume 6, Issue 10, October 2019

This was achieved by means of a suitable hash-based index, which maps any tuple to a set of indexing keys and assigns tuples with high syntactic similarity to the same buckets. An extensive experimental evaluation on both synthetic and real data has shown the efficacy of their proposed approach.

III. MATERIALS AND PROPOSED METHODS

Here, we propose a neural network-basedclassification, trained by locality-sensitive hash for duplicate chunk recognition. Our method is based on low complexity hash functions and neural network architecture. Chunk size is expected to reach several Kilo-Bytes, so we reduce the computation complexity by examining the hash value of each chunk. The proposed classification method consists of following phases:

Data Collection Processing
Proposed SCIL-NN algorithm
Hash Function
AppDataDedup Searching index method

A) DATA COLLECTION PROCESSING

Sample dataset:

According to the collected history data, the sample set is created as {(X(i), y(i)}.X(i) is the influencing factor set and y(i) is the actual load value on the i the point. The total integer of the data points is N.

The influencing factor set is shown as the following:

X(i) = {ir(i), im(i), li(i), lu(i), he(i), ff(i), mf(i), bs(i), dm(I)}

Whereas, ir(i) is iris dataset, im(i) is image classification dataset, li(i) is liver dataset, he(I) is the heart dataset, mf(i) is mangoleaf dataset, ff(i) is forest fire dataset, bs(i) is bamboo stick dataset, sm(i) is social media dataset. Treat all data of the sample by normalization and smoothing processing so that computing overflow will be avoided.

Load type classification

According to the actual load value, the data points are classified into three types as the following: the high load type, the medium load type and the low load type. Respectively, the given set of data points {(X(i), y(i)} is divided into three subsets, which are high load sub-set H, medium load sub-set M and low load sub-set S.

H= {(X(i), y(i)} i=1,2…m y(i) € [y(max)-y*, y(max)]
M={X(i), y(i)} i=1,2…n y(i) € [y(min)-y*, y(max)-y]
S= {X(i), y(i)} i=1,2…l y(i) € [y(min)-y(min)+y]
Sd={m+n+l=N and y*=1/3(y(max-ymin}

Where mn is the quantity of the data points for each sub-set, y(max) is the largest load value in the given data point set and the y(min) is lowest load. Each load type is given a value as the output of the artificial neural network. The value for high load type is 0, the medium load type is 1/3 and the low load type is 1.0.

Sample preprocessing:

The ANN model of load type classification is a three-layer BP network, which contains the input layer including 6 nodes, the intermediate layer including 10 nodes and the output layer including one node. According to the classification mentioned above, train the BP network. The influencing factor set is used as the input of the Back Propagation neural network while loading type values as an output. Both the input data and output data will be normalized before the training. Sigmoid Function is selected as the activated Function F1(x) in the intermediate layer is,

F1(x)= 1/(1+e-x)

ISSN: 2395-7639

International Journal of Multidisciplinary Research in Science, Engineering, Technology & Management (IJMRSETM)

(A Monthly, Peer Reviewed Online Journal)

Visit: www.ijmrsetm.com

Volume 6, Issue 10, October 2019

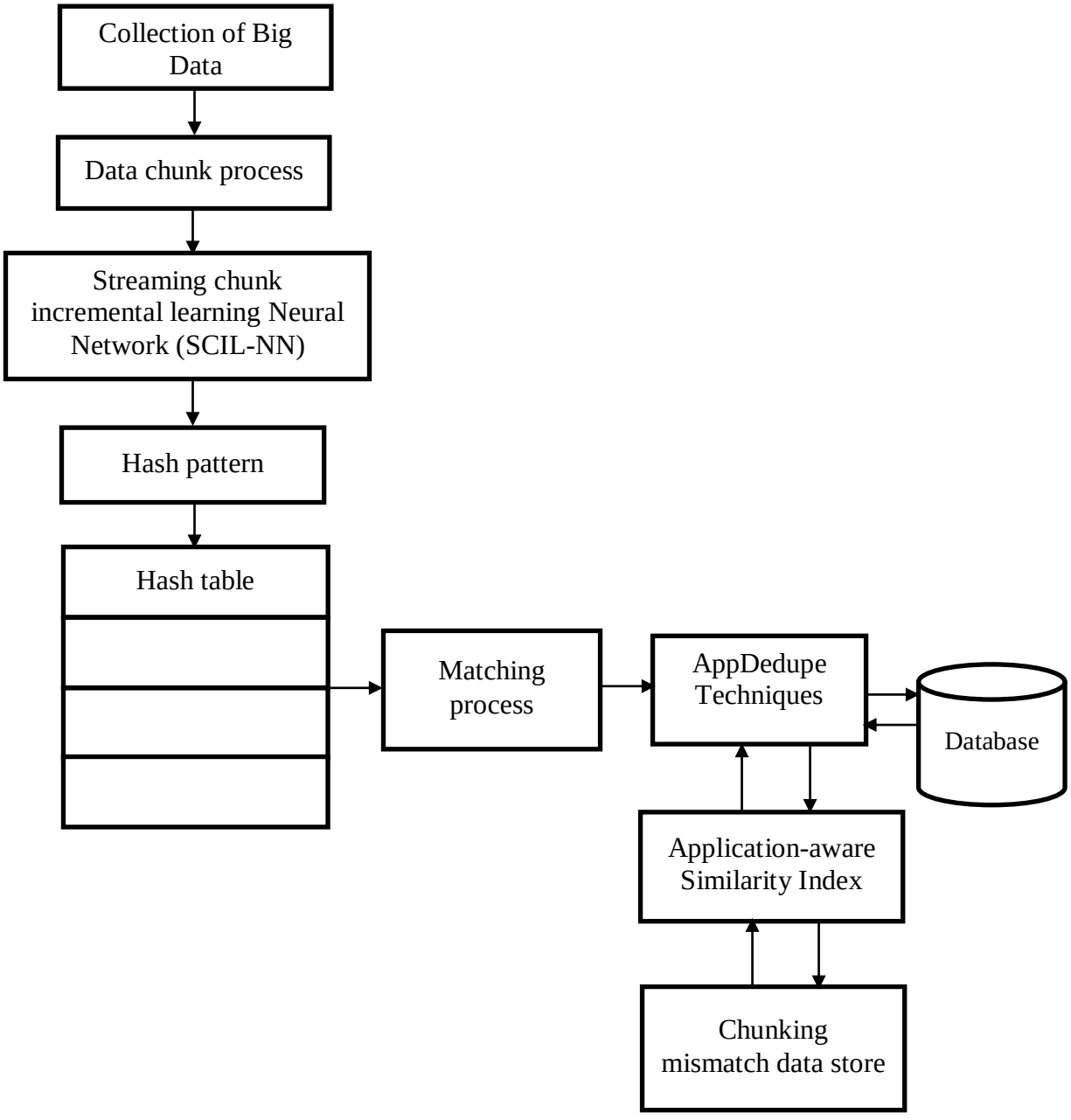

Figure-1 Architecture of the proposed system

The activated Functionin output layer is PureLinear Function, which is
F2(X)= ax
Where 'a'is a linear coefficient.
As all the sample data has been normalized, the variables of Sigmoid Function are within [-1, 1] and a real nonlinear transference is made successfully. Using the ANN tool box, selecting the gradient decreasing method to train the Trained function, letting the convergence precision be 2e-7, setting the dynamic parameter as 0.05 and the largest training times as 2000， we can acquire the demanded precision within 1000 times of interactions. After being trained, the BP network is used to forecast the load type of the predict point. The sub-set (The high load sub-set, the medium load sub-set H or the low load sub-set M) of the same load type will be selected as the training sample set for the SVM

ISSN: 2395-7639

International Journal of Multidisciplinary Research in Science, Engineering, Technology & Management (IJMRSETM)

(A Monthly, Peer Reviewed Online Journal)

Visit: www.ijmrsetm.com

Volume 6, Issue 10, October 2019

in next step.

B) PROPOSED (SCIL-NN) ALGORITHM

Neural network training

In order to set adaptive statistical classifier, we have two-phase training procedure. In the 1st phase, referred to as pre-training, we train the NN with given pass data. This data consists of the hashes of the stored data. Second phase learning is ongoing during deduplication system run: if a chunk is marked as stored by NN but actual storage search did not result in finding a duplicate, the NN is trained with the chunk's hash.

The pre-training is done by the following steps: random initialize weights Θ, implemented forward propagation algorithm to get hΘ(x(i)) for any x(i). Compute cost function J(Θ). Perform backpropagation and forward propagation with gradient decent to try to minimize J(Θ) as a function of Θ. All those are known NN algorithms. In the second learning phase, we perform only the last step, namely just backpropagation and forward propagation with gradient decent according to new added hash.

Chunking in neural network

The NN outputs a probability that the chunk is already stored. A threshold value is required in order to decide whether to store the chunk as there is no identical stored chunk or to speculate that a duplicate does exist and go on to find it and verify, then store a pointer to it. The choice of the threshold value represents a trade-off between false positive (resulting in extra computation) and false negative (resulting in wasted storage space) decision probabilities. The "pass" distribution is the collection of all chunk's hashes and their corresponding probabilities that are already stored. Similarly, "fail" distribution is the collection of chunk hashes that are not stored and their NN assign probabilities.

Streaming Chunk Incremental Learning- Neural Network (SCIL-NN) technique

For each Δk, is defined number of neurons in any classofkcan be enlarged due to the large number of distances between the uncovered datum and the existing VEBFs in Δk. Overfitting problems occurred due to the present of too many neurons. Therefore, one method to avoid the increase of neurons is to merge two nearby hidden neurons in the same Δk. Two hidden neurons αk= (a1(k), b1(k), c1(k), d1(k)) and βk= (a2(k), b2(k), c2(k), d2(k)) are merged if the following circumstance is satisfied:

$\in \alpha k (b1(k)) \leqslant 0$ or $\in \beta k (b2(k)) \leqslant 0$

This means that either αk or βk cover the center of another. Anew neuron Ωk = (a3(k), b3(k), c3(k), d3(k)) is convinced to interchange αk and βk after integration them, and the parameters are calculated and defined as follows

a3(k) = a1(k)+ a2(k),

b3(k)=1/ a3(k) (a1(k)x1(k)+ a2(k) b2(k))

c3(k)=(a1(k))/a3(k) +(a2(k))/a3(k) +(a1(k)a2(k))/(a3(k))(b1(k)-b2(k))(b1(k)-(b2(k))

w(k) = z(a/2). √ (μ(k)/m(k))

Load forecasting by SCIL-NN

The selected training sample set above is used to train the support vector machine and the SCIL-NN algorithm is adopted. The influencing factor set is used as the input of the SCIL-NN, while the weight value as the output. Both the input data and the harvest data will be normalized before the training.

ISSN: 2395-7639

International Journal of Multidisciplinary Research in Science, Engineering, Technology & Management (IJMRSETM)

(A Monthly, Peer Reviewed Online Journal)

Visit: www.ijmrsetm.com

Volume 6, Issue 10, October 2019

Algorithm:

Input: (1) Data set **X***k* of class *k* in *n*-dimensional space.

(2) Initial width value of each created neuron.

Output: A set of trained neurons for data set **X***k*.

1. **If** class *k* is a new class **then**
2. Let set **Δ***k* =μ;
3. Create a set of hidden neurons by using **Algorithm 1** and put themin **Δ***k*.
4. Compute all parameters of neurons in **Δ***k* by using the recursive functions.
5. **Else**
6. **Do** lines 7-10 **Until X***k* is empty *or* no neuron used for updatingparameter.
7. Compute the mean vector **x** of the current data **X***k*.
8. Select the neuron **Ω***k* such that

   a= arg min {¥(k)(x)}
9. Update all relevant parameters of **Ω***k by* using the recursivefunctions
10. merge **Ω***k* with other neurons of thesame class.
11. end do
12. If x(k) is not empty
13. Do lines 14-15 until x(k) is empty
14. create a new hidden neuron **Ω***k and set* **Δ***k*= **Δ***k*. **Ω***k*
15. Update all relevant parameters of **Ω***k by* using the recursivefunctions
16. end do
17. end if
18. end if
19. Discard x(k)

The input layer has 6 nodes and the output layer has only one node. The Gauss function is selected as the kernel function.

ISSN: 2395-7639

International Journal of Multidisciplinary Research in Science, Engineering, Technology & Management (IJMRSETM)

(A Monthly, Peer Reviewed Online Journal)

Visit: www.ijmrsetm.com

Volume 6, Issue 10, October 2019

$K(x(i), x(j) = \exp(ix(I)-x(j)/\beta)$

Where the β is the width parameter of the Gauss kernel. According to experience, the value of parameters is selected as the following

C=10, €=0.01, β=1, α=10-5

Input the influencing factors of the predict point into the trained SCIL-NN, the output is the deduplicated value.
The experimental results exposed that the accuracy of SCIL-NN ishigher than the others data sets of result. The below algorithm of SCIL-NN is used to chunk the data into various type of gathered big data. In demand to determine the threshold, collected two range of probability distributions.

C) HASH FUNCTION

To increase the efficiency of neural network, duplicate chunks have to produce higher values than unique chunks. Therefore, duplicate chunks have to produce similar hash values. In order to accomplish this, Locality-Sensitive Hashing (LSH) functions are utilized. In our design, LSH function is implemented by bit sampling at the required hash length. The bit indices are constant, selected randomly at system initialization time.

Error estimation

Error probability is calculated as the overlap area between duplicate and non-duplicate chunks distributions. Detailed analysis assigns different error weights for storage and computation according to cost or another metric. For example, the energy spent on unnecessary chunk search is modeled to be half of spare chunk storage. The error probability is calculated apostriori and accumulated as more data is stored. Threshold is dynamically adjusted for error minimization.

D) APPDEDUPE BASED REDUNDANCY ANALYSIS

In cloud data centers there are large number of massive data comes from the application clients for processing. The paper compared with chunk dataset with aninter and intra deduplication using the chunk level outcome result. Empirical observation reveals the total amount of data overlapped among the various types of uploaded files. The goal of the research consists of content aware deduplication techniques by search index of hash pattern and achieve high level of redundancy analysis by deduplication techniques that effectively neglect the overlap process.

Principle Design of AppDataDedup

In this section, used the following three principles design to manage our AppDataDedup system design:

Scalability. The data distributed deduplication system would simply scale out to handle huge data volumes with stable workload among the nodes.

Capacity. Similar data content should be progressed to the matching deduplication node to succeed high duplication to eliminate ratio.

Throughput. Across the storage nodes deduplication throughput should be scaled with the number of nodes by matching deduplication

In this section the framework of data deduplication distributed to achieve negligible capacity loss with good scalability and high level of throughput. Then research presents our data routing scheme to achieve scalable performance with high deduplication efficiency. This is followed by the explanation of the application-aware data schema for high deduplication throughput in data deduplication nodes.

IV. EXPERIMENTAL ANALYSIS

In the proposed deduplication technique two criteria are considered for the evaluation purpose, one is accuracy and the other is time for execution. The datasets, which we are used in our proposed approach, is detailed as follows: Iris, image segmentation, liver, lungs, heart, forest fire, mango leaf, bamboo stick, and social media.

ISSN: 2395-7639

International Journal of Multidisciplinary Research in Science, Engineering, Technology & Management (IJMRSETM)

(A Monthly, Peer Reviewed Online Journal)

Visit: www.ijmrsetm.com

Volume 6, Issue 10, October 2019

Table-1 Average classification accuracy with standard deviation of each dataset

Dataset	SCIL-NN	ArtificialNeural Network (ANN)	Robust Incremental Learning method (RIL)
Iris	96.34	85.6	76.27
image segmentation	65.43	45.64	38.6
Liver	87.4	74.2	74.7
Lungs	79.5	61.3	76.8
Heart	69.2	61.6	67.3
Forest fire	97.2	56.3	87.5
Mango leaf	81.6	69.1	70.4
Bamboo stick	93.6	81.6	78.3
Social media	75.3	69.1	70.3
Average rank	**1.67**	**3.8**	**2.78**

Table-2 Average Number of utilized hidden layer of neurons with the Standard Deviation (SD) of each data set.

Dataset	SCIL-NN	ArtificialNeural Network (ANN)	Robust Incremental Learning method (RIL)
Iris	35.71	4.567	8.45
image segmentation	14.75	5.78	67.34
Liver	56.87	6.78	74.7
Lungs	3.75	3.56	20.85
Heart	57.1	8.45	12.67
Forest fire	6.67	7.234	67.45
Mango leaf	8.34	7.67	56.7
Bamboo stick	6.78	9.753	6.45
Social media	3.56	8.456	5.34
Average rank	**2.67**	**1.45**	**4.79**

The experimental results suggest that SCIL-NN and AppDataDedup can reduce the training time effectively as compared with existing methods unless the number of input attributes is too large.

ISSN: 2395-7639

International Journal of Multidisciplinary Research in Science, Engineering, Technology & Management (IJMRSETM)

(A Monthly, Peer Reviewed Online Journal)

Visit: www.ijmrsetm.com

Volume 6, Issue 10, October 2019

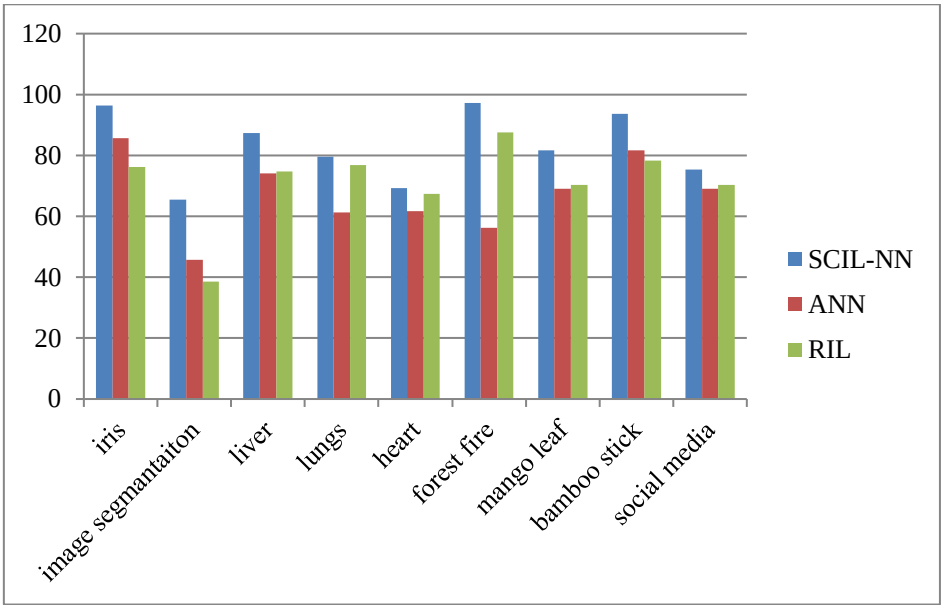

Figure-2 comparison of Average classification accuracy

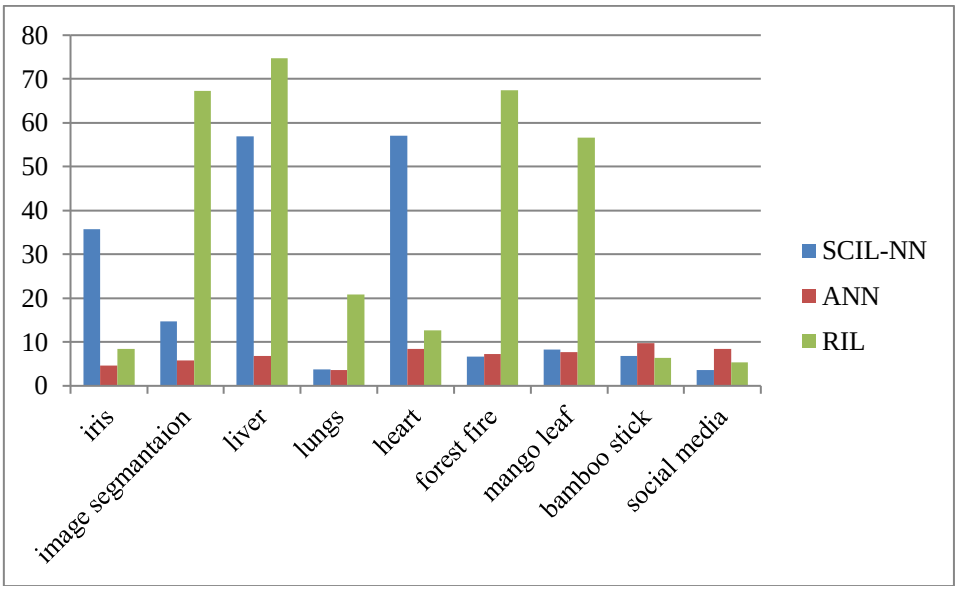

Figure-3 Used average number of hidden neurons

Accuracy:

The accuracy is the percentage of exact results such as true positives and true negatives in the overall data. This is considered has parameter of the test data and the accuracy value is considered from the following equation:

Accuracy=Number oftruepositives/numbers oftruenegatives

Here the total number of duplicates is measured as the number of true negatives and the numbers of non-duplicates are considered as the true positive. The variance in their value is considered as the accurateness of the proposed deduplication technique.

ISSN: 2395-7639

International Journal of Multidisciplinary Research in Science, Engineering, Technology & Management (IJMRSETM)

(A Monthly, Peer Reviewed Online Journal)

Visit: www.ijmrsetm.com

Volume 6, Issue 10, October 2019

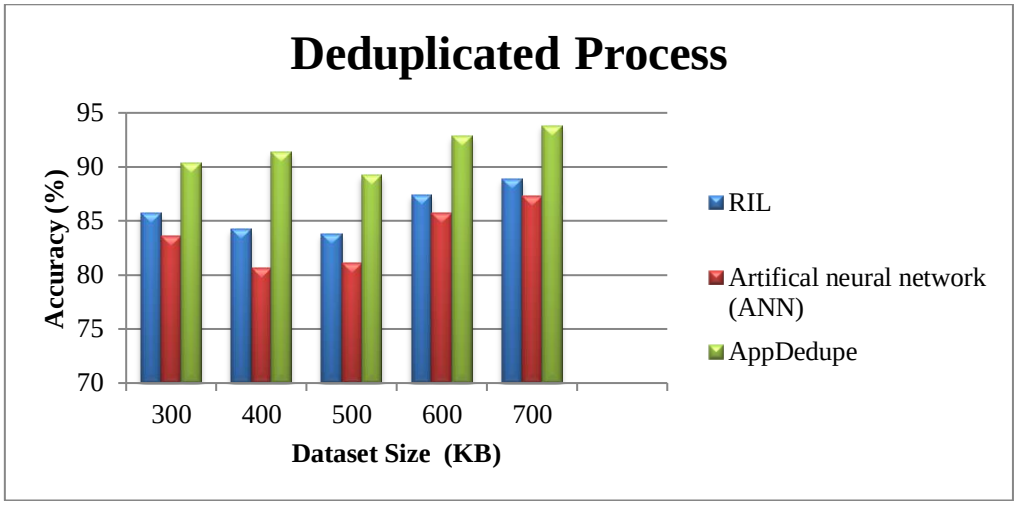

Figure-4 Accurate Deduplication Analyzing

2. Time calculation

Time is the factor that defines the required time for executing the proposed deduplication technique. The time for execution is calculated from the starting of the proposed technique to till the termination of the proposed technique.

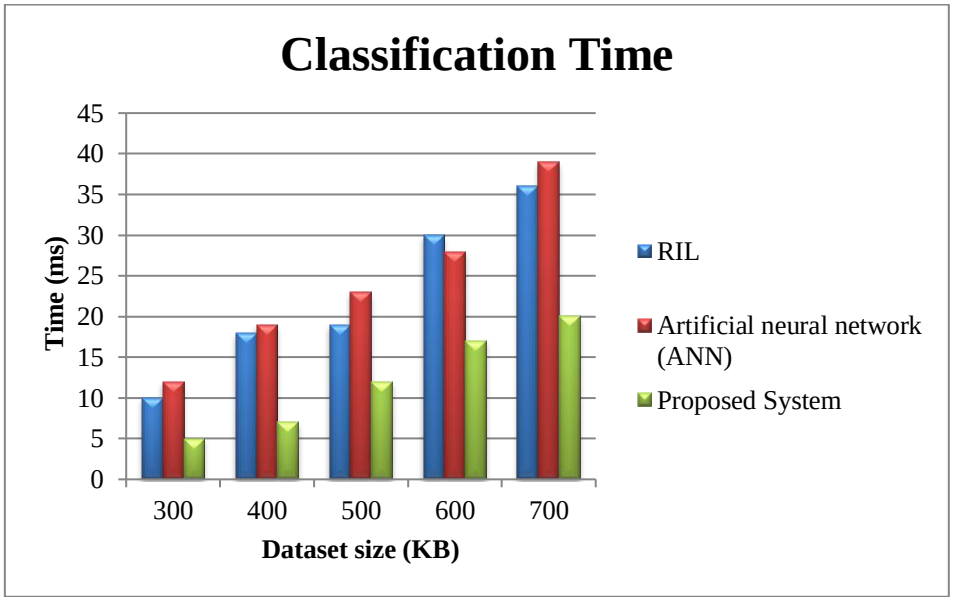

Figure-5 Time classification method

The proposed system of SCIL-NN and AppDataDedup algorithm are taken less time for classification and detection process. It is probable to study the effect of the size of primary training data set and the size of given chunk and deduplication data on classification accuracy and learning time.

V. CONCLUSION

The research concluded with SCIL-NN algorithm to handle a data chunk through streaming data. In this study, the chunk data are partitioning by multiple classes. Significant aspects of the learning algorithm based on chunking method from gathered massive amount of data. The SCIL-NN algorithm is used to chunk and hash index are stored in the cache memory of data for hashing pattern process. The proposed algorithm has completely analyzed the whole data and

partitioned with chunk data and the index term are stored as hash function. Describe AppDataDedup, an application-aware distributed deduplication frame-work for managing big data, which reaches scalable performance between effective tradeoff distributed deduplication by manipulating data similarity. Our real-world clearly demonstrates AppDataDedup's important advantageous over large clusters of distributed deduplications.

REFERENCES

[1] Gu M, Li X, Cao Y (2014) "Optical storage arrays: a perspective for future big data storage". Light Science Application 3(5): e177.
[2] Tian Y, Khan SM, Jiménez DA, Loh GH (2014) "Last-level cache deduplication". In: Proceedings of the 28th ACM International Conference on Supercomputing, pp 53–62.
[3] Hovhannisyan H, Qi W, Lu K, Yang R, Wang J (2016) "Whispers in the cloud storage: a novel crossuser deduplication-based covert channel design. Peer-to-Peer Networking and Applications", pp 1–10.
[4] Mandagere N, Zhou P, Smith MA, Uttamchandani S (2008) "Demystifying data deduplication". In: Proceedings of the ACM/IFIP/USENIX Middleware'08 Conference Companion, pp 12–17. https:// doi.org/10.1145/1462735.1462739
[5] Parthasarathy, P., &Vivekanandan, S. (2020). Biocompatible TiO2-CeO2 Nano-composite synthesis, characterization and analysis on electrochemical performance for uric acid determination. Ain Shams Engineering Journal, 11(3), 777-785.
[6] Mao B, Jiang H, Wu S, Fu Y, Tian L (2014) "Read-performance optimization for deduplication-based storage systems in the cloud. In: ACM Transactions on Storage (TOS)", volume 10(2).
[7] Parthasarathy, P., &Vivekanandan, S. (2020). A typical IoT architecture-based regular monitoring of arthritis disease using time wrapping algorithm. International Journal of Computers and Applications, 42(3), 222-232.
[8] Wang J, Chen X (2016) "Efficient and secure storage for outsourced data: a survey". Data Sci Eng 1(3):178–188.
[9] S. P. Deshpande and V. M. Thakare," Data Mining System and Applications: A Review," International Journal of Distributed and Parallel systems, Vol. 1, No. 1, pp. 32-44, 2010
[10] Panchatcharam, P., &Vivekanandan, S. (2019). Internet of things (IOT) in healthcare–smart health and surveillance, architectures, security analysis and data transfer: a review. International Journal of Software Innovation (IJSI), 7(2), 21-40.
[11] Xindong Wu, Vipin Kumar, Qiang Yang, J. Ross Quinlan, Joydeep Ghosh, Michael Steinbach, David J. Hand and Dan Steinberg," Top 10 Algorithms in Data Mining," Knowledge and Information Systems, Vol. 14, No. 1, pp. 1-37,2007.
[12] H. Zhang, A. C. Berg, M. Maire, and J. Malik, "SVM-KNN: Discriminative nearest neighbor classification for visual category recognition", In International Conference on Computer Vision and Pattern Recognition, New York (NY), USA, 2006.
[13] Ektefa, M, Sidi. F,Ibrahim. H, Jabar. M.A., Memar. S, Ramli. A, "A threshold-based similarity measure for duplicatedetection ", IEEE conference on Open systems, pp: 37-41, 2011
[14] Hong-Jie Dai, Chi-Yang Wu, Richard Tzong-Han Tsai and Wen-Lian Hsu, "From Entity Recognition to Entity Linking: A Survey of Advanced Entity Linking Techniques", The 26th Annual Conference of the Japanese Society for Artificial Intelligence, 2012.
[15] Parthasarathy, P., &Vivekanandan, S. (2018). Investigation on uric acid biosensor model for enzyme layer thickness for the application of arthritis disease diagnosis. Health information science and systems, 6(1), 5.
[16] Mikhail Bilenko and Raymond J. Mooney, "Adaptive duplicate detection using learnable string similarity measures", ACM SIGKDD international, New York, 2003.
[17] K.Deepa and Dr.R.Rangarajan, "An Approach to Duplicate Record Detection Using Similarity Metrics and Anais", Journal of Computational Information Systems, vol. 8, no. 6, pp. 2231-2243, 2012.
[18] Elhadi. M, Al-Tobi. A, "Duplicate Detection in Documents and Webpages Using Improved Longest Common Subsequence andDocuments Syntactical Structures", "International Conference on Computer Sciences and Convergence Information Technology", pp: 679-684, 2009
[19] P. Christen, "A Survey of Indexing Techniques for Scalable Record Linkage and Deduplication", IEEE, no. 99, pp. 1, 2011.
[20] Gianni Costa, Giuseppe Manco and Riccardo Ortale, "An incremental clustering scheme for data de-duplication", DATA MINING AND KNOWLEDGE DISCOVERY, vol. 20, no. 1, pp. 152-187, 2010
[21] Varadharajan, R., Priyan, M. K., Panchatcharam, P., Vivekanandan, S., & Gunasekaran, M. (2018). A new approach for prediction of lung carcinoma using back propagation neural network with decision tree classifiers. Journal

of Ambient Intelligence and Humanized Computing, 1-12.
[22] Mathan, K., Kumar, P. M., Panchatcharam, P., Manogaran, G., &Varadharajan, R. (2018). A novel Gini index decision tree data mining method with neural network classifiers for prediction of heart disease. Design Automation for Embedded Systems, 22(3), 225-242.