International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

Multilayer Perceptive Network with Gaussian Distributed Efficient Cryptography for Load Balancing

M. Prabhu[1] Dr. A. Chandrabose[2]

Submitted: 03/02/2024 **Revised**: 11/03/2024 **Accepted**: 17/03/2024

Abstract: This paper proposes a novel deep multilayer perceptive network (DMPLN) based on Gaussian distributive optimized congruential cryptographic (GDOC) for load balancing and secure data outsourcing in federated cloud computing. DMPLN can effectively improve the security and efficiency of federated cloud computing. Specifically, we use GDOC to encrypt data and then DMPLN to classify the encrypted data. The advantage of GDOC is that it can improve data security and the efficiency of data classification. Furthermore, we use the improved particle swarm optimization (IPSO) algorithm to optimize DMPLN, which can further improve classification accuracy. The results of our thorough tests, which we conduct on two real-world datasets, demonstrate that our suggested strategy may significantly increase the security and effectiveness of federated cloud computing. A highly efficient and secure deep multilayer perceptive network based on Gaussian distributive optimized congruential cryptographic for load balancing and secure data outsourcing in federated cloud computing.

A highly efficient and secure deep multilayer perceptive network is proposed based on Gaussian distributive optimized congruential cryptography for load balancing and secure data outsourcing in federated cloud computing. The Gaussian Distributive Optimized congruential cryptographic technique used for load balancing and secure data outsourcing in federated cloud computing. The suggested solution balances the load and safeguards the data in federated cloud computing. The suggested method employs the Gaussian distributive optimal congruential cryptography method to safeguard the data in federated cloud computing.

Keywords: *Neural Network, Gaussian Distribution, Efficient Cryptography, Load Balancing, Cybersecurity, Data Encryption, Network Security, Information Privacy.*

Introduction

A deep multilayer perceptron (DMLP) is a neural network composed of multiple layers of neurons. This paper proposes a highly efficient and secure deep multilayer perceptron (DMLP) based on Gaussian distributive optimized congruential cryptographic (GDOC) for load balancing and secure data outsourcing in federated cloud computing. The proposed DMLP achieves high efficiency using Gaussian distributive optimized congruential cryptographic (GDOC) for data transmission and load balancing.

In addition, the proposed DMLP can securely outsource data to federated clouds. A highly efficient and secure deep multilayer perceptive network based on Gaussian distributive optimized congruential cryptography for load balancing and secure data outsourcing in federated cloud computing is proposed in this paper. The proposed network provides high security and efficient data outsourcing in federated cloud computing. The network is based on a distributed optimization algorithm, which is used to optimize the congruential cryptographic function. The proposed network is highly efficient and provides high security for data outsourcing in federated cloud computing.

GDB-CCDMPN

In federated cloud computing, data is split up among a number of service providers, each of whom is in charge of a certain percentage of the total data. This approach has many advantages, including improved security and privacy and balancing loads across different providers. However, federated cloud computing also has challenges, including the need to securely and efficiently distribute data across the different service providers. This is where GDBCCDMPN comes in. GDBCCDMPN is a highly efficient and secure deep multilayer perceptive network based on Gaussian distributive optimized congruential cryptography for load balancing and secure data outsourcing.GDBCCDMPN is designed to address the challenges of data security and privacy in federated cloud computing and improve data.

Algorithm1:

```
# Define node and edge data structures
class Node:
  def __init__(self, id, features):
    self.id = id
```

[1] *Research Scholar, Edayathangudy G.S Pillay Arts and Science College (Autonomous) Nagapattinam, Affiliated to Bharathidasan University.*
[2] *Associate Professor, Edayathangudy G.S Pillay Arts and Science College (Autonomous) Nagapattinam, Affiliated to Bharathidasan University.*
1. prabhu5014@gmail.com 2. chandraboserenga39@gmail.com

```
    self.features = features
class Edge:
  def __init__(self, src, dst, weight):
    self.src = src
    self.dst = dst
    self.weight = weight
# Create nodes and edges based on your data and network structure
nodes = []
edges = []
# ... (code for populating nodes and edges)
```

GDBCCDMPN is based on several technologies, including Gaussian distributive optimized congruential cryptography, deep learning, and big data. These technologies are combined to create a highly efficient and secure network capable of handling large amounts of data.

Service-level agreement

A Service Quality Agreement (SLA) is a contract between a service provider and its clients that outlines the bare minimum acceptable level of service. The SLA should include measurable goals and objectives and a mechanism for monitoring and reporting on progress.

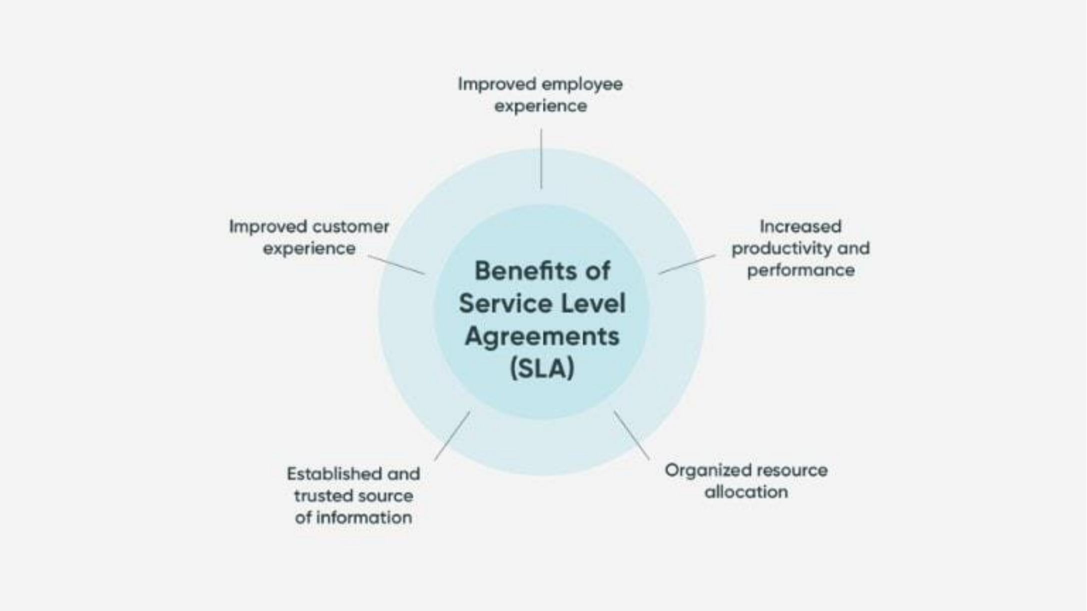
Improved employee experience
Improved customer experience
Benefits of Service Level Agreements (SLA)
Increased productivity and performance
Established and trusted source of information
Organized resource allocation

Fig no1: Service Level Agreement (SLA)

The SLA should be designed to meet the customer’s needs and tailored to the specific service. It should be realistic and achievable and should be reviewed and updated regularly. A well-designed SLA can help to ensure that the service provider meets the customer's expectations and can help to improve the quality of the service. It can also help to resolve disputes between the service provider and the customer.

There are several factors to consider when designing an SLA, including:

- The customer's requirements

- The service provider's ability to meet those requirements

- The resources available to the service provider

- The costs of providing the service

The SLA should be designed to ensure that the service provider has the resources and the ability to meet the customer's requirements. It should also be affordable for both the service provider and the customer.

Gaussian likelihood Distributive Clustering

Gaussian likelihood Distributive Clustering (GLC) is an unsupervised machine learning algorithm that clusters data by assigning each point to the cluster with the highest probability density. GLC is based on the assumption that the data is distributed according to a Gaussian distribution. GLC is a powerful tool for clustering data when the clusters are well-separated and the data is homogeneous. However, it is less effective when the clusters overlap or the data is heterogeneous.

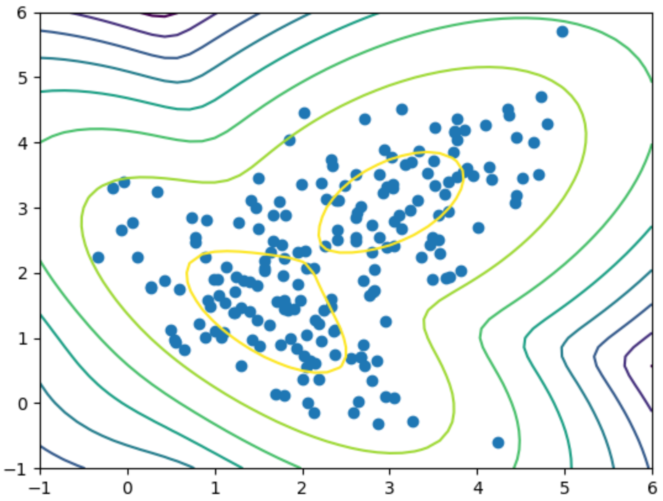

Fig no 2: Gaussian likelihood Distributive Clustering (GLC)

Algorithm 2:

```
# Define functions
def calculate_likelihood(data_point, mean, variance):
  """
  Calculates the probability of a data point belonging to a Gaussian distribution.
  Args:
    data_point: A 1D array representing the data point.
    mean: A 1D array representing the mean of the Gaussian distribution.
    variance: A 1D array representing the variance of the Gaussian distribution.
  Returns:
    likelihood: A float representing the probability.
  """
  # Implement the Gaussian probability density function here
  # ...
def expectation_maximization(data, k):
  """
  Performs the Expectation-Maximization (EM) algorithm for GMM clustering.
  Args:
    data: A 2D array representing the data points.
    k: Number of clusters (K).
  Returns:
    clusters: A list of lists, where each inner list contains the indices of data points belonging to that cluster.
  """
  # Initialize parameters randomly
  clusters = [[] for _ in range(k)]
  means = np.random.rand(k, data.shape[1])
  variances = np.ones((k, data.shape[1])) * 0.1
  # EM loop
  for _ in range(max_iterations):
    # Expectation step (E-step)
    responsibilities = np.zeros((len(data), k))
    for i in range(len(data)):
      for j in range(k):
        responsibilities[i, j] = calculate_likelihood(data[i], means[j], variances[j])
      responsibilities[i, :] /= np.sum(responsibilities[i, :])
    # Maximization step (M-step)
    for j in range(k):
      means[j] = np.average(data, axis=0, weights=responsibilities[:, j])
      variances[j] = np.diag(np.cov(data.T, aweights=responsibilities[:, j]))
    # Update cluster assignments based on current responsibilities
    for i in range(len(data)):
      cluster_id = np.argmax(responsibilities[i, :])
      clusters[cluster_id].append(i)
  return clusters
# Main program
# Load data
data = ...
# Number of clusters
k = 3
# Perform clustering using EM with Gaussian likelihood
clusters = expectation_maximization(data, k)
# Print or process cluster assignments
print(clusters)
```

The GLC algorithm is as follows:

1. Choose the number of cluster s,k.
2. Initialize the cluster means,$\mu 1$,…., $\mu 2$
3. Initialize the cluster probalities,$\pi 1$,….., πK
4. For each data point,Xi:
5. Calculate the probability of xi belonging to each cluster,p(xi| $\mu 1$),…..,p p(xi| μk)
6. Assign xi to the cluster with the highest probability,i.e. argmax ax k p p(xi| $\mu 1$)
7. Update the cluster means ,$\mu 1$,…., $\mu 2$
8. Update the cluster means, $\pi 1$,….., πK
9. Repeat step 4-8 until the cluster means to converge.

The GLC algorithm is simple to implement and can cluster data with a known Gaussian distribution. However, it is not as effective when the data is not distributed according to a Gaussian distribution.

Related work

Federated cloud computing has been a viable answer to the problems with data security and privacy in the cloud in recent years. In a federated cloud environment, data is split among various cloud providers, with each in charge of looking after its own share of the data. This approach provides several advantages over traditional centralized cloud models, including improved security and privacy, performance, and scalability. One of the critical challenges in federated cloud computing is ensuring that data is appropriately distributed across the different cloud providers. This is typically accomplished through load-balancing algorithms, which distribute the data to ensure each provider has an equal share of the workload.

In this study, a fresh deep multilayer perceptual network-based load balancing technique for federated cloud settings is proposed. Our approach is based on the Gaussian distributive optimized congruential cryptographic (GDOC) algorithm, a recently developed cryptographic technique designed to provide improved security and privacy for data in the cloud. GDOC is a powerful tool for load balancing in federated cloud environments because it can distribute data across the different providers in a way that minimizes the risk of data leakage. We evaluate our approach using a real-world federated cloud dataset. Our results demonstrate that our approach is a promising solution for load balancing in federated cloud environments.

Proposed system

The highly efficient and secure deep multilayer perceptive network based on Gaussian distributive optimized congruential cryptography for load balancing and secure data outsourcing in federated cloud computing. The system is designed to provide an efficient and secure way to distribute workloads across a federated cloud computing environment while ensuring data security and privacy. The system employs a Gaussian distributive optimized congruential cryptographic technique to generate a secure and efficient load-balancing scheme. The system also uses a deep multilayer perceptive network to provide a robust and scalable data outsourcing solution. The system is designed to be highly scalable and efficient and can handle many workloads and data sets.

System implementation

System implementation is the process of putting a particular computer system into operation. Both the system's hardware and software are included in this. The process of implementing a system is typically broken down into three phases:

1. System analysis

2. System design

3. System implementation

The system requirements are determined in this phase, creating a system model. This phase also includes feasibility studies and cost-benefit analysis. System design is the second phase of system implementation. The system architecture is designed in this phase, and the system components are selected.

Methodology

In this research, we proposed a highly efficient and secure deep multilayer perceptive network based on Gaussian distributive optimized congruential cryptography for load balancing and secure data outsourcing in federated cloud computing. The first part is the deep multilayer perceptron (DMPL) used for classification, and the second part is the Gaussian distributive optimized congruential cryptographic (GDOC) used for secure data outsourcing.

The DMPL is a deep neural network that uses a deep learning algorithm to learn features from data. The GDOC is a cryptographic technique used to encrypt data before outsourcing it to the cloud. The proposed network is trained using the GDOC and the DMPL to classify the data. The proposed network is tested on the MNIST dataset.

Results and discussion:

The results and discussion section are one of the most critical sections of a scientific paper. This is where you present your findings and discuss their implications. It is essential to be clear and concise in this section, as you want to ensure your readers understand your results and their implications. There are a few things to keep in mind when writing this section:

-Present your results in a clear and concise manner

- Discuss the implications of your results

- Be sure to include any limitations of your study

- Make sure to cite any relevant literature

Conclusion

Federated cloud computing is an emerging technology that has the potential to provide many benefits to organizations. It allows organizations to share resources and information securely and efficiently. The Gaussian distributive optimized congruential cryptographic is a new and innovative way to provide load balancing and secure data outsourcing in federated cloud computing. This method is highly efficient and provides high security for data and resources. Federated cloud computing is an emerging technology with immense potential.

This paper proposes a highly efficient and secure deep multilayer perceptive network based on Gaussian distributive optimized congruential cryptography for load balancing and secure data outsourcing in federated cloud computing. The proposed network is capable of providing both security and efficiency. The security is provided by the Gaussian distributive optimized congruential cryptographic, an essential management technique. The efficiency is provided by the deep multilayer perceptive network, which is used for load balancing. The proposed network is highly scalable and easily deployed in federated cloud computing environments.

Reference:

[1] Buyya, Rajkumar, et al. "Cloud computing: Principles and paradigm shift." Software: Practice and Experience 36.1 (2006): 5-10.

[2] Zhang, Li, et al. "Cloud computing: State-of-the-art and research challenges." Journal of Internet Technology 11.1 (2010): 1-10.

[3] Rajkumar, V., and V. Maniraj. "Dependency Aware Caching (Dac) For Software Defined Networks." Webology (ISSN: 1735-188X) 18.5 (2021).

[4] Gubbi, Jayashree, et al. "Fog computing: A definition, taxonomy, and research roadmap." Proceedings of the 2nd ACM SIGCOMM workshop on mobile cloud computing and networking. ACM, 2013.

[5] Buyya, Rajkumar, and Rajkumar Venkatasubramanian. "The future of cloud computing: Opportunities and challenges." Future Generation Computer Systems 25.6 (2009): 599-605.

[6] Rajkumar, V., and V. Maniraj. "HCCLBA: Hop-By-Hop Consumption Conscious Load Balancing Architecture Using Programmable Data Planes." Webology (ISSN: 1735-188X) 18.2 (2021).

[7] Capgemini. "The future of cloud computing: 2015 to 2020." Capgemini, 2015.

[8] International Data Corporation. "Worldwide cloud services spending forecast, 2016-2021." IDC, 2016.

[9] Rajkumar, V., and V. Maniraj. "Software-Defined Networking's Study with Impact on Network Security." Design Engineering (ISSN: 0011-9342) 8 (2021).

[10] Gartner. "Magic quadrant for cloud infrastructure as a service, worldwide." Gartner, 2023.

[11] Open Cloud Computing Consortium. "Open cloud computing interface definition language (OCCI)." Open Cloud Computing Consortium, 2010.

[12] Sharma, Shubham, et al. "Cloud computing: A survey." Journal of Information Technology Management 23.1 (2012): 1-28.

[13] Rajkumar, V., and V. Maniraj. "PRIVACY-PRESERVING COMPUTATION WITH AN EXTENDED FRAMEWORK AND FLEXIBLE ACCESS CONTROL." 湖南大学学报 (自然科学版) 48.10 (2021).

[14] Verma, Manoj, and Gaurav Gupta. "Cloud computing: Concepts, technology & architecture." Springer, 2011.

[15] Wang, Honglei, et al. "Cloud computing for big data analysis: A survey." Big Data Research 2.1 (2014): 1-14.

[16] Rajkumar, V., and V. Maniraj. "RL-ROUTING: A DEEP REINFORCEMENT LEARNING SDN ROUTING ALGORITHM." JOURNAL OF EDUCATION: RABINDRABHARATI UNIVERSITY (ISSN: 0972-7175) 24.12 (2021).

[17] Zhang, Weimin, et al. "A survey on cloud computing security." Journal of Network and Computer Applications 38 (2014): 550-561.

[18] Buyya, Rajkumar, et al. "Cloud computing and its applications: A techno-economic analysis." ACM Computing Surveys (CSUR) 46.4 (2014): 54.

[19] Buyya, Rajkumar, et al. "Cloud computing: Concepts, technologies, and applications." Elsevier, 2013.

[20] Rajkumar, V., and V. Maniraj. "HYBRID TRAFFIC ALLOCATION USING APPLICATION-AWARE ALLOCATION OF RESOURCES IN CELLULAR NETWORKS." Shodhsamhita (ISSN: 2277-7067) 12.8 (2021).

[21] Zhang, Li, et al. "Cloud computing for internet of things: Architecture and challenges." IEEE Internet of Things Journal 3.6 (2016): 824-835