Scholastic Seed Inc.
www.gjeis.com
global Journal of ENTERPRISE INFORMATION SYSTEM

Fog Computing: An Innovative Technique for the Quality Improvement in IOT Communication

– Rajbala Simon*
Addl. Superintendent examination, AIIT, Amity University, Noida, India
rajbalasimon@gmail.com *https://orcid.org/0000-0002-7204-3486*

– Laxmi Ahuja
Dy. Director, AIIT, Amity University, Noida, India
lahuja@amity.edu *https://orcid.org/0009-0002-1446-2735*

– Akansha Kumari
Student, Amity Institute of Information Technology, Noida, India
akanshakumarichand@gmail.com *https://orcid.org/0009-0001-4631-5185*

– Saurav Vinod Nair
Student, Amity Institute of Information Technology, Noida, India
sauravvinodnair@outlook.com *https://orcid.org/0009-0005-4048-5975*

ARTICLE HISTORY

Paper Nomenclature: View Point
Paper Code: GJEISV15I4OD2023VP1
Submission at Portal (www.gjeis.com): 22-Oct-2023
Manuscript Acknowledged: 25-Oct-2023
Originality Check: 29-Oct-2023
Originality Test (Plag) Ratio (DrillBit): 05%
Author Revert with Rectified Copy: 10-Nov-2023
Peer Reviewer Comment (Open): 21-Nov-2023
Single Blind Reviewers Explanation: 24-Nov- 2023
Double Blind Reviewers Interpretation: 27-Nov-2023
Triple Blind Reviewers Annotations: 29-Nov-2023
Author Update (w.r.t.correction, suggestion & observation): 11-Dec-2023
Camera-Ready-Copy: 20-Dec-2023
Editorial Board Excerpt & Citation: 26-Dec-2023
Published Online First: 31-Dec-2023

ABSTRACT

Purpose: Fog computing is introduced as a relatively new trend in analysis. The purpose is to extend cloud computing services to the edge of the network to reduce latency and congestion. It emphasises low latency, broader distribution, and geographically dispersed nodes for improved real-time communication.

Design/Methodology/Approach: Cloud computing services are extended to the edge of the network. Focus on reducing latency and congestion through fog computing architecture. Mention of comparable resources and services between cloud and fog computing, with an emphasis on the latter's low latency and geographic distribution.

Findings: Highlighting the limitations of cloud computing, including minimum mobility, lack of location-awareness, and unreliable latency. Recognition of fog computing as a solution offering flexible resources and services that extend to devices at the network's outer edges. Cloud computing is acknowledged for handling resource demands of devices at the core of the network.

Originality/Value: Fog computing is presented as a solution addressing the impracticalities of cloud computing, particularly for certain applications like IoT. Fog computing offers the computational efficiency of cloud computing with the bandwidth enhancements of a local network. Noting the interest of computer scientists and researchers due to fog computing's unique capabilities.

Paper Type: View Point

KEYWORDS: Fog | Computing | Latency | Bandwidth | Cloud

**Corresponding Author* (Rajbala)

- Present Volume & Issue (Cycle): Volume 15 | Issue-4 | Oct-Dec 2023
- International Standard Serial Number:
  Online ISSN: 0975-1432 | Print ISSN: 0975-153X
- DOI (Crossref, USA) https://doi.org/10.18311/gjeis/2023
- Bibliographic database: OCLC Number (WorldCat): 988732114
- Impact Factor: 3.57 (2019-2020) & 1.0 (2020-2021) [CiteFactor]
- Editor-in-Chief: Dr. Subodh Kesharwani
- Frequency: Quarterly
- Published Since: 2009
- Research database: EBSCO https://www.ebsco.com
- Review Pedagogy: Single Blind Review/ Double Blind Review/ Triple Blind Review/ Open Review

- Publishers: Scholastic Seed Inc. and KARAM Society
- Place: New Delhi, India.
- Repository (figshare): 704442/13

Introduction

The "pay-as-you-go" computing also known as cloud computing is a good option to manage confidential data centers for clients using web applications. It has gotten progressively significant due to many related qualities, for example, minimal storage cost, accessibility, availability of information whenever and wherever required, and easy maintenance. Mobile devices can gather personal information from different sensors within a shorter timeframe and sensor-based information comprises significant data from users. However, mobile access presents numerous issues, for example, duplication to make information effectively accessible, access to required information, security of information, and Artificial Intelligence strategies for speedy and effective access to information.

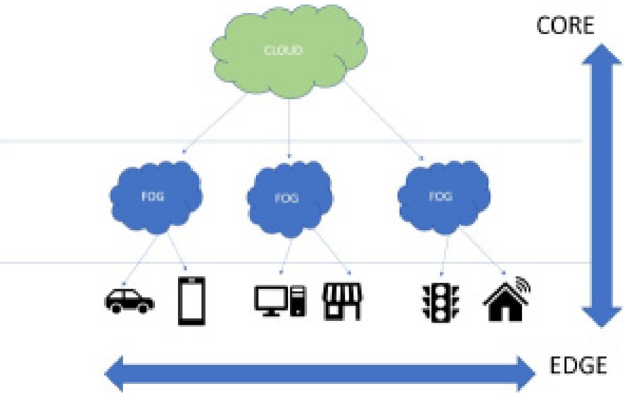


Figure 1. Fog Computing and its Overview

Another choice for Cloud Computing is Fog Computing since fog is closer to the ground when contrasted with cloud additionally assert that as opposed to utilizing the cloud for registering, Fog processing serves another type of uses an administration. It revolves around bringing Cloud-based administrations closer to Internet of Things (IoT) customers in an advantageous way. An enormous bit of the physical devices in Fog figuring conditions, typically named Fog hubs, geographically dispersed and are heterogeneous. To utilize the limits of the Fog hubs, tremendous scope applications that are disintegrated into between subordinate Application Modules can be disseminated or passed on effectively over hubs subject to their idleness affectability. In this paper, we have discussed the technique behind fog computing, its architecture, its combination with IOT, and its overview.

In the above figure, it is depicted that just as in nature fog is nearer to the ground than clouds in a similar way, by utilizing fog as an intermediary, it has become technically feasible to bring clouds within closer reach of the ground where end-users and devices are located. We can see that fog is the distributed medium of cloud computing and is thus more suitable for users as it is not centralized and hence latency and workload are low comparatively.

The Fog Platform

Fog Computing otherwise called edge figuring was presented in 2012 by Cisco. Fog processing was essentially acquainted to meet the constant prerequisites of a huge number of uses and gadgets. It must be noted that Fog is not a replacement for traditional cloud but just an alternative to it. Any end devices that have networking, computing, and storage capabilities in it are termed Fog Nodes. Examples can be video surveillance cameras, switches, routers, machines, mobile devices, and servers. Fog computing aims to decrease the workload of data being transported to the cloud for processing and storage. It leads to reacting more quickly to events.

Fog Computing Features:

Fog computing is an innovative platform that serves networking, and processing services to end devices and traditional cloud data centers. It is designed to operate on a similar foundation as the cloud, using computing, storage, and networking components. Different features make the Fog a critical extension of the cloud. A portion of the highlights of fog registering are initially, edge, region care, and low inaction. Fog enlisting gives rich organizations at the edge of the framework including applications that require low inactivity. Land conveyance is another component of Fog registering correlation with the cloud, the organizations, and applications given by the Fog demand flowed categorization. Expansive scale sensor systems to screen the natural. For instance: - a savvy framework which require conveyed figuring and capacity assets. A substantial number of hubs which is a consequence of geo-distribution that we can see by and large in sensor systems and especially in savvy lattice. Support for portability is again another vital feature of Fog. It is particularly imperative to collaborate Fog applications to cell phones and in this way bolster versatility methods e.g., LISP convention, that different host personality from area character. Real-time interactions are another feature. Fog applications work on real-time interaction other than batch processing. Control of remote access and heterogeneity where Fog hubs come in various structure components and this manner will be circulated in an assortment of conditions. Interoperability by which Fog applications or parts must have the ability to interoperate across various spaces.

Fog Computing and IOT

In this segment, we substantiate the significance of FOG in Smart Agriculture:

A. ***Smart Agriculture***

The era of agriculture 4.0, also known as smart agriculture, has arrived as a result of a profound integration of current information technology and conventional agriculture.

Agriculture intelligence and automation are taken care of by smart agriculture.

The results of a Fog Computing survey in agriculture are currently available. The following was the process used to find research for this survey. A combination of search phrases was used to search the Scopus and Web of Science archives. Edge computing and fog computing were the most popular search phrases. These were supplemented by wildcard agricultural and farm variants. Horticulture, aquaculture, and forest were introduced as a second level of domain-specific search phrases. Finally, to complete the search query, an extra level of specialized phrases was included, including crop, animal, fruit, vegetable, and fish. Wildcards were used to express all search phrases in the final script.

B. Crop production

A proposed framework that follows the Internet of Things (IoT) concept focuses primarily on comprehensive monitoring capabilities suggested, and it has been deployed and verified in a vineyard scenario that monitors a variety of weather and soil quality parameters. The objective is to be able to predict some diseases that are linked to vineyards, but where meteorological conditions are important determinants of outbreaks. Black rot and downy mildew are two instances of such disorders. Data sourced through the distributed sensor network is gathered by a Fog computing node, which makes the whole architecture Cloud-centric. Data from the distributed sensor network is gathered by a Fog computing node, which makes the whole architecture Cloud-centric. mySense is a generic platform designed for quickly developing and deploying monitoring apps in precision viticulture settings (Morais et al., 2019). Sensors/actuators, WSN/gateway, Web/Cloud, and applications are the four levels. For local operations and real-time alert creation, fog computing is used at the WSN/Gateway layer.

The platform has been used in a vineyard to study disease dynamics about current microclimates. Park et al. have given an illustrative case study that demonstrates how fog computing might offer scalable data analytics (2017). In this case, a Raspberry Pi serves as both a sensor configuration base station and a network Fog node. The Fognode generates a growth state forecast for cherry tomatoes, which is then sent to a cloud-based central server for conflation, model integration, and analysis, resulting in yield projections. This method reduces data traffic but also allows farmers to save their data and only share what they want.

Reliable Fog Computing - Based Architecture

Cloud, Fog, and Things are the three layers of a Fog hierarchical architecture. The Cloud Tier is made up of one or more Data Centers that provide services that need a lot of processing power (e.g., big data analysis). Fog Nodes are found in the Fog Tier (FNs). The physical or virtual components that enhance the connection between the Things Tier and the Cloud Tier, such as routers, switches, servers, wireless access points, video surveillance cameras, and controllers, improve the performance of IoT-based services. These services include wireless sensor-based data collection, tracking systems, and applications, which is referred to as FN. IoT-enabled devices, such as sensor nodes, make up the Things Tier. This Tier is in charge of communicating with the outside world and providing data to the Fog Tier. Our Fog-based and reliability-oriented design is shown in Figure 1. In contrast to typical fog-based systems, ours divides the Fog Tier into two separate layers:

By appropriately situating management services and classifying resources in layers based on domain needs, IoT dependability may be improved. The Layer 2 FN is the closest point to IoT devices, which is why we place FDM and FRM mechanisms in this layer, since making choices with trustworthy data is critical for IoT-based applications. As a result, we prevent apps operating on the top tiers use erroneous data.

The accuracy of harvest estimating and forecasting models can be negatively impacted by outliers in collected data, resulting in poor decisions and financial losses. To address this issue, Layer 2 FN consists of several edge computing nodes that collect data from IoT devices and implement reliability-focused processes. FDM is utilized to detect outliers (as explained in Section 3.3), while FRM is employed to determine the appropriate values to substitute the identified outliers (as discussed in Section 3.4).

Layer 1 includes Fog Controllers (FCs) that coordinate network layers and conduct preliminary data analysis, utilizing their larger processing, storage, and networking capabilities compared to Layer 2 FN. 1st Figure Fog architecture with a hierarchical structure.

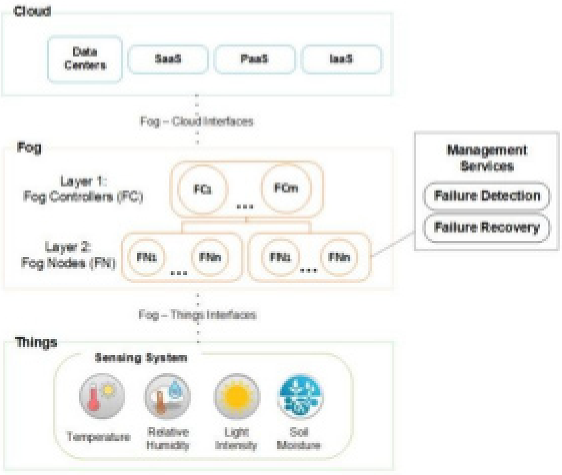
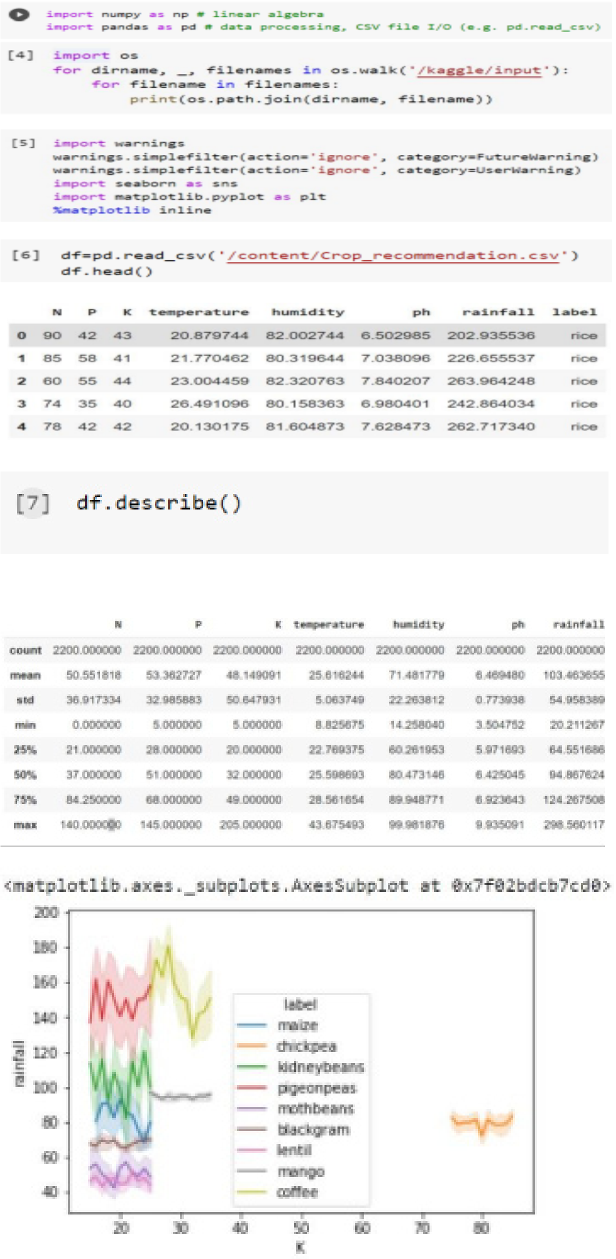


Figure 2. Fog Hierarchical Architecture

A. *Analysis and Prediction of Crop Quality*

The analysis of soil conditions data, such as temperature, moisture content, and chemical composition, which all affect crop development and livestock health, is an area where machine learning shines. This can dramatically increase the efficacy of farmers' decisions in modern agriculture by enabling crops to be produced with a great deal more accuracy and enabling farmers to treat plants and animals almost individually. This might be utilized to develop techniques for estimating harvest yields, assessing crop quality for every kind of plant, and identifying weed infestations and agricultural diseases that were previously challenging to find!



```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
```

```
[4] import os
    for dirname, _, filenames in os.walk('/kaggle/input'):
        for filename in filenames:
            print(os.path.join(dirname, filename))
```

```
[5] import warnings
    warnings.simplefilter(action='ignore', category=FutureWarning)
    warnings.simplefilter(action='ignore', category=UserWarning)
    import seaborn as sns
    import matplotlib.pyplot as plt
    %matplotlib inline
```

```
[6] df=pd.read_csv('/content/Crop_recommendation.csv')
    df.head()
```

	N	P	K	temperature	humidity	ph	rainfall	label
0	90	42	43	20.879744	82.002744	6.502985	202.935536	rice
1	85	58	41	21.770462	80.319644	7.038096	226.655537	rice
2	60	55	44	23.004459	82.320763	7.840207	263.964248	rice
3	74	35	40	26.491096	80.158363	6.980401	242.864034	rice
4	78	42	42	20.130175	81.604873	7.628473	262.717340	rice

```
[7] df.describe()
```

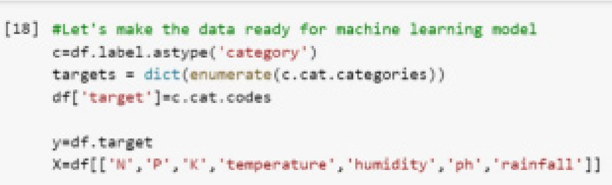
	N	P	K	temperature	humidity	ph	rainfall
count	2200.000000	2200.000000	2200.000000	2200.000000	2200.000000	2200.000000	2200.000000
mean	50.551818	53.362727	48.149091	25.616244	71.481779	6.469480	103.463655
std	36.917334	32.985883	50.647931	5.063749	22.263812	0.773938	54.958389
min	0.000000	5.000000	5.000000	8.825675	14.258040	3.504752	20.211267
25%	21.000000	28.000000	20.000000	22.769375	60.261953	5.971693	64.551686
50%	37.000000	51.000000	32.000000	25.598693	80.473146	6.425045	94.867624
75%	84.250000	68.000000	49.000000	28.561654	89.948771	6.923643	124.267508
max	140.000000	145.000000	205.000000	43.675493	99.981876	9.935091	298.560117

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f02bdcb7cd0>
```

Figure 3. Screenshots Showing Analysis and Prediction on Crop Quality

B. *Pre-Processing of Data*

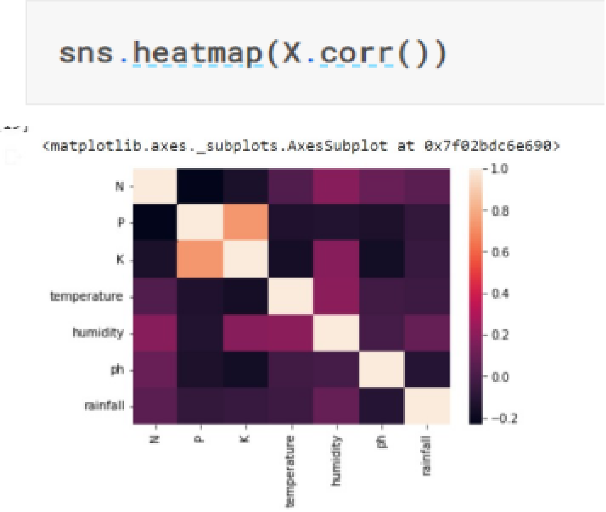
- Making the data ready for machine learning model



```
[18] #Let's make the data ready for machine learning model
     c=df.label.astype('category')
     targets = dict(enumerate(c.cat.categories))
     df['target']=c.cat.codes

     y=df.target
     X=df[['N','P','K','temperature','humidity','ph','rainfall']]
```

Figure 4. Screenshot Stating Pre-processing of Data

- Visualization of feature correlations. We can see how closely phosphorus and Potassium levels are connected.



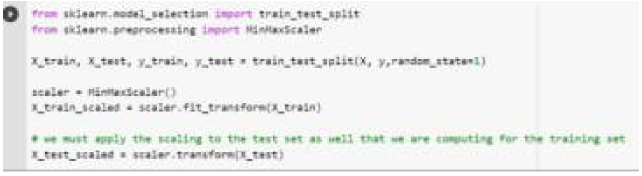
```
sns.heatmap(X.corr())
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f02bdc6e690>
```

Figure 5. Visualization of Feature Correlations

C. *Feature Scaling Analysis*

Before producing training data and feeding it to the model, feature scaling is necessary. Because two of our features (temperature and ph) are gaussian distributed, we may use MinMaxScaler to scale them between 0 and 1.

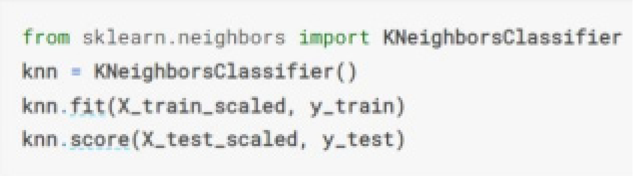


```
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import MinMaxScaler

X_train, X_test, y_train, y_test = train_test_split(X, y,random_state=1)

scaler = MinMaxScaler()
X_train_scaled = scaler.fit_transform(X_train)

# we must apply the scaling to the test set as well that we are computing for the training set
X_test_scaled = scaler.transform(X_test)
```

Figure 6. Using MinMaxScaler

D. *Algorithm for Model Selection*

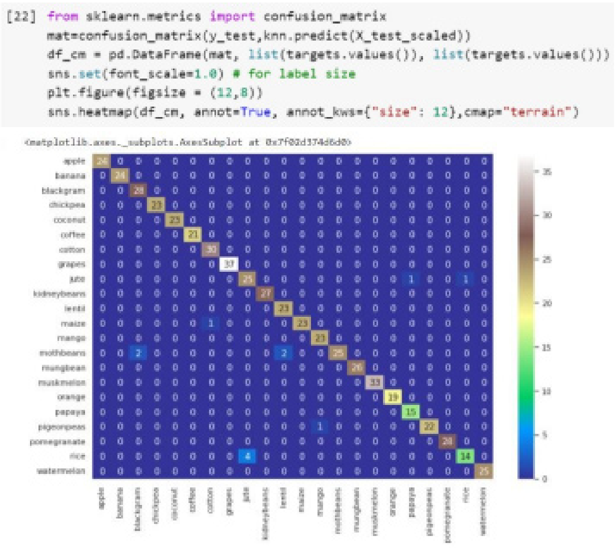
- Classifier for Crop prediction



```
from sklearn.neighbors import KNeighborsClassifier
knn = KNeighborsClassifier()
knn.fit(X_train_scaled, y_train)
knn.score(X_test_scaled, y_test)
```

Figure 7. Screenshot showing Crop Prediction

- Confusion Matrix



```
[22] from sklearn.metrics import confusion_matrix
     mat=confusion_matrix(y_test,knn.predict(X_test_scaled))
     df_cm = pd.DataFrame(mat, list(targets.values()), list(targets.values()))
     sns.set(font_scale=1.0) # for label size
     plt.figure(figsize = (12,8))
     sns.heatmap(df_cm, annot=True, annot_kws={"size": 12},cmap="terrain")
```

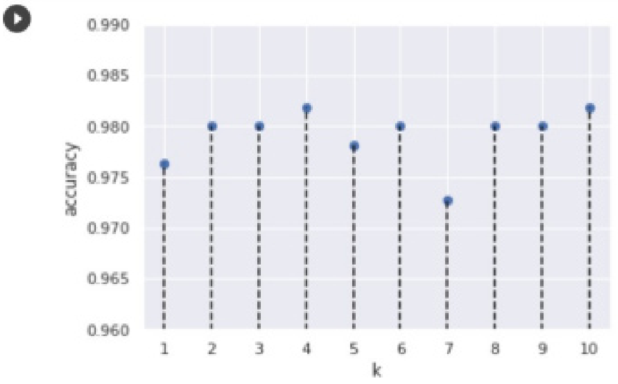
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f02d374d6d0>
```

	apple	banana	blackgram	chickpea	coconut	coffee	cotton	grapes	jute	kidneybeans	lentil	maize	mango	mothbeans	mungbean	muskmelon	orange	papaya	pigeonpeas	pomegranate	rice	watermelon
apple	24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
banana	0	24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
blackgram	0	0	28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
chickpea	0	0	0	23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
coconut	0	0	0	0	23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
coffee	0	0	0	0	0	21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
cotton	0	0	0	0	0	0	30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
grapes	0	0	0	0	0	0	0	37	0	0	0	0	0	0	0	0	0	0	0	0	0	0
jute	0	0	0	0	0	0	0	0	25	0	0	0	0	0	0	0	0	1	0	0	1	0
kidneybeans	0	0	0	0	0	0	0	0	0	27	0	0	0	0	0	0	0	0	0	0	0	0
lentil	0	0	0	0	0	0	0	0	0	0	23	0	0	0	0	0	0	0	0	0	0	0
maize	0	0	0	0	0	0	1	0	0	0	0	23	0	0	0	0	0	0	0	0	0	0
mango	0	0	0	0	0	0	0	0	0	0	0	0	23	0	0	0	0	0	0	0	0	0
mothbeans	0	0	2	0	0	0	0	0	0	0	2	0	0	25	0	0	0	0	0	0	0	0
mungbean	0	0	0	0	0	0	0	0	0	0	0	0	0	0	26	0	0	0	0	0	0	0
muskmelon	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	33	0	0	0	0	0	0
orange	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	19	0	0	0	0	0
papaya	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	15	0	0	0	0
pigeonpeas	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	22	0	0	0
pomegranate	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	0	0
rice	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	14	0
watermelon	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	25

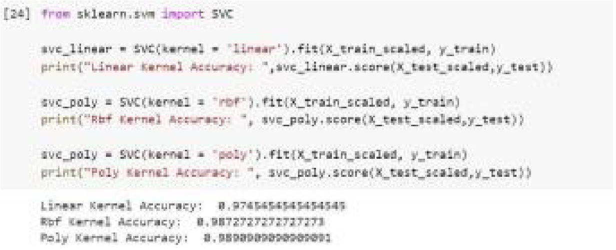
Figure 8. Screenshots showing Confusion Matrix

- Let's try different values of n_neighbors to fine tune and get better results

```
[23] k_range = range(1,11)
     scores = []

     for k in k_range:
         knn = KNeighborsClassifier(n_neighbors = k)
         knn.fit(X_train_scaled, y_train)
         scores.append(knn.score(X_test_scaled, y_test))

     plt.xlabel('k')
     plt.ylabel('accuracy')
     plt.scatter(k_range, scores)
     plt.vlines(k_range,0, scores, linestyle="dashed")
     plt.ylim(0.96,0.99)
     plt.xticks([i for i in range(1,11)]);
```



- Classification using Support Vector Classifer (SVC)



```
[24] from sklearn.svm import SVC

     svc_linear = SVC(kernel = 'linear').fit(X_train_scaled, y_train)
     print("Linear Kernel Accuracy: ",svc_linear.score(X_test_scaled,y_test))

     svc_poly = SVC(kernel = 'rbf').fit(X_train_scaled, y_train)
     print("Rbf Kernel Accuracy: ", svc_poly.score(X_test_scaled,y_test))

     svc_poly = SVC(kernel = 'poly').fit(X_train_scaled, y_train)
     print("Poly Kernel Accuracy: ", svc_poly.score(X_test_scaled,y_test))
```

```
Linear Kernel Accuracy:  0.9745454545454545
Rbf Kernel Accuracy:  0.9872727272727273
Poly Kernel Accuracy:  0.9890909090909091
```

Figure 9. Screenshots showing Support Vector Classifer (SVC)

Let's try to increase SVC Linear model accuracy by parameter tuning.

- GridSearchCV can help us find the best parameters.

```
[25] from sklearn.metrics import accuracy_score
     from sklearn.model_selection import GridSearchCV

     parameters = {'C': np.logspace(-3, 2, 6).tolist(), 'gamma': np.logspace(-3, 2, 6).tolist()}
     # 'degree': np.arange(0,5,1).tolist(), 'kernel':['linear','rbf','poly']

     model = GridSearchCV(estimator = SVC(kernel="linear"), param_grid=parameters, n_jobs=-1, cv=4)
     model.fit(X_train, y_train)
```

```
GridSearchCV(cv=4, estimator=SVC(kernel='linear'), n_jobs=-1,
             param_grid={'C': [0.001, 0.01, 0.1, 1.0, 10.0, 100.0],
                         'gamma': [0.001, 0.01, 0.1, 1.0, 10.0, 100.0]})
```

```
[26] print(model.best_score_ )
     print(model.best_params_ )
```

```
0.9866710547967747
{'C': 1.0, 'gamma': 0.001}
```

Figure 10. Screenshots Showing Algorithms for Model Selection

The Interplay Between the Fog and the Cloud

Fog center points give close by access to data and engage low inaction and setting care furthermore while the Cloud is seen united all around. Fog limitation and Cloud centralization are required by various people of the applications, especially B DATA. We have quite recently inspected this point in setting to smart traffic light. Let us take an instance of sha structure to grasp the trade.

Fog gatherers at the edge exhaust the information made by the lattice sensors and contraptions. Different information requests progressing taking care of from milliseconds to sub seconds. This is a bit of the Fog expected for machine-to-machine associates. Fog gatherers at the edge exhaust the information made by the lattice sensors and contraptions. Different information requests progressing taking care of from milliseconds to sub seconds. This is a bit of the Fog expected for machine-to-machine associations will do the social event and methodology of information and send the officials headings to the actuators. It in addition refines the information to be exhausted at that point and send the remaining to the upper fragments. The second and third a bit of the F works with virtualization and consideration and with systems and methodology. Beyond what many would con possible for these exchanges between all bits of the Fog shift from several minutes to minutes and on occasion may take even two or three days. Concludingly Fog must join different sorts of limits from brief at the most decreased level to semi-invariant at the most raised the higher the level, the broader the geographical incorporation and besides time scale increase.

Comparison of Fog and Cloud in terms of Security

The cloud is centrally managed and contains huge amounts of data that can be stored all over the world, far from client devices. The fog is scattered and contains several small nodes near the client device. F is the layer between the cloud and devices such as computers, laptops, and mobile phones. Fog acts as an intermediary, reducing the time it takes to transfer data. Without a layer, the cloud would have to communicate directly with the end device, which would take more time than using fog computing. Cloud computing has low latency, but it doesn't compare to fog. Fog computing provides low latency when it comes to networking. Cloud computing does not reduce data when it is sent, whereas fog computing reduces data when it is sent to the cloud. Cloud computing provides less bandwidth savings compared to Fog.

Table. 1. Comparison of Fog and Cloud in terms of Security

S. No	BASIS	FOG COMPUTING	CLOUD COMPUTING
1.	Response Time	Low	High
2.	Transmission	Device to Device	Device to Cloud
3.	Computational focus	Fog functions on the network edge	Data requests are handled in the Cloud
4.	Goal	Enhance the efficiency and effectiveness of the process by transferring it to the Cloud for analysis or storage.	Immensity and powerful provision of IT administrations.
5.	Operating System Support	Hypervisor Virtualization	A hypervisor (VM) on which various OS can run
6.	Ownership	Multiple	Single
7.	Service price	Utility pricing and payment are lying on the uses	Utility pricing discounted for larger usage
8.	Multitask Support	Yes	Yes

Conclusion

We have discussed about what is Fog computing and its key features. Fog computing helps the cloud to mana the two exabytes of information which is produced regularly from IOT. It animates mindfulness and reaction to activity dismissing expansion of costly transfer speed by emptying gigabytes of system movement from the center network. additionally, examinations within organization dividers and secure touchy IOT information. Concludingly, organization utilizations FOG registering increase further and speedier insights, which prompting increment in business agility, higher a more secure level. In the above algorithm, the liner kernel produces excellent results as well, but precise tweaking increases calculation time and may be inefficient in some circumstances. Tweaking settings in the poly kernel can improve accuracy, but this may result in excessive overfitting that outperforms the linear kernel in terms of performance. Poly Kernel has won by a slim margin thus far. Thus, we conclude that we observe Fog to be a joining stage that is sufficiently proficient to deliver news of administrations and help to update the new applications.

References:

- Bhatt, R., Singh, P., Majumder, D., & Kaur, G. (2023). Climate-smart technologies for reducing water footprints in different cropland ecosystems: a meso analysis. *Proceedings of the Indian National Science Academy*, 1-22.
- Bonomi, F., Milito, R., Natarajan, P., & Zhu, J. (2014). Fog Computing: A Platform for Internet of Things and Analytics. Dalam N. Bessis & C. Dobre (Ed.), Big Data and Internet of Things: A Roadmap for Smart Environments (Vol. 546, hlm. 169–186).
- Cedric, L. S., Adoni, W. Y. H., Aworka, R., Zoueu, J. T., Mutombo, F. K., Krichen, M., & Kimpolo, C. L. M. (2022). Crops yield prediction based on machine learning models: case of west african countries. *Smart Agricultural Technology*, 100049.
- Das, R., & Inuwa, M. M. (2023). A review on Fog Computing: Issues, Characteristics, Challenges, and Potential Applications. *Telematics and Informatics Reports,* 100049.
- Fan, C. T., Wu, Z. Y., Chang, C. P., & Yuan, S. M. (2016, July). Web resource cacheable edge device in fog computing. In *2016 15th International Symposium on Parallel and Distributed Computing (ISPDC)* (pp. 432-439). IEEE.
- Geetha, K. (2022). An Integrated Approach for Crop Production Analysis from Geographic Information System Data using SqueezeNet. *Journal of Soft Computing Paradigm,* 3(4), 308-321.
- Junior, F. M. R., Bianchi, R. A., Prati, R. C., Kolehmainen, K., Soininen, J. P., & Kamienski, C. A. (2022). Data reduction based on machine learning algorithms for fog computing in IoT smart agriculture. *Biosystems Engineering,* 223, 142-158.
- Kong, J. L., Fan, X. M., Jin, X. B., Su, T. L., Bai, Y. T., Ma, H. J., & Zuo, M. (2023). BMAE-Net: A Data-Driven Weather Prediction Network for Smart Agriculture. *Agronomy*, 13 (3), 625.
- Li, D., Nanseki, T., Chomei, Y., & Kuang, J. (2023). A review of smart agriculture and production practices in Japanese large-scale rice farming. *Journal of the Science of Food and Agriculture*, 103(4), 1609-1620.

- Liu, L., Wang, X., & Chang, C. (2022). Toward a smart skin: Harnessing cuticle biosynthesis for crop adaptation to drought, salinity, temperature, and ultraviolet stress. *Frontiers in Plant Science*, 13.
- M. Chiang and T. Zhang, Fog and IoT: An Overview of Research Opportunities, *IEEE Internet of Things Journal*, 3(6), 854-864, Dec. 2016. doi: 10.1109/JIOT.2016.2584538
- Mubvuma, M. T., Nyambiya, I., Makaza, K., Chidoko, P., Mudzengi, C. P., Dahwa, E., ... & Mupambwa, H. A. (2023). The Potential of Vermicomposts in Sustainable Crop Production Systems. In Vermicomposting for Sustainable Food Systems in Africa (pp. 261-277). *Singapore: Springer Nature Singapore.*
- Ngo, V. M., Duong, T. V. T., Nguyen, T. B. T., Dang, C. N., & Conlan, O. (2023). A big data smart agricultural system: recommending optimum fertilisers for crops. *International Journal of Information Technology,* 1-17.
- Oikonomidis, A., Catal, C., & Kassahun, A. (2022). Hybrid deep learning-based models for crop yield prediction. *Applied artificial intelligence*, 36(1), 2031822.
- Padhy, S., Alowaidi, M., Dash, S., Alshehri, M., Malla, P. P., Routray, S., & Alhumyani, H. (2023). AgriSecure: A Fog Computing-Based Security Framework for Agriculture 4.0 via Blockchain. *Processes,* 11(3), 757.
- Qamar, T., & Bawany, N. Z. (2023). Agri-PAD: a scalable framework for smart agriculture. *Indonesian Journal of Electrical Engineering and Computer Science*, 29(3), 1597-1605.
- Sharief M.A.Otefy and Hossam S.Hassanel, "IoT in the Fog: A roadmap for Data-centric IoT development", *157 IEEE Communication Magazine, March 2018.*
- Vignesh, B., & Suganya, G. (2023). Deep learning methodologies for crop models and decision support systems: A comprehensive review. *Soft Computing*, 1-9.
- Yu Cao, Songqing Chen, Peng Hou and D. Brown, "FAST: A fog computing assisted distributed analytics system to monitor fall for stroke mitigation," 2015 IEEE International Conference on Networking, Architecture and Storage (NAS), Boston,MA, 2015, pp. 2-11. *Fog Computing in IoT Proceedings of ARSSS International Conference, 04th February, 2018, Pune, India 21*
- Zahed, A. and Sakhi, M., 2011. A Novel Technique for Enhancing Security in Biometric Based Authentication Systems. *International Journal of Computer and Electrical Engineering*, pp.520-523.
- Ziliani, M. G., Altaf, M. U., Aragon, B., Houborg, R., Franz, T. E., Lu, Y., ... & McCabe, M. F. (2022). Early season prediction of within-field crop yield variability by assimilating CubeSat data into a crop model. *Agricultural and Forest Meteorology*, 313, 108736.

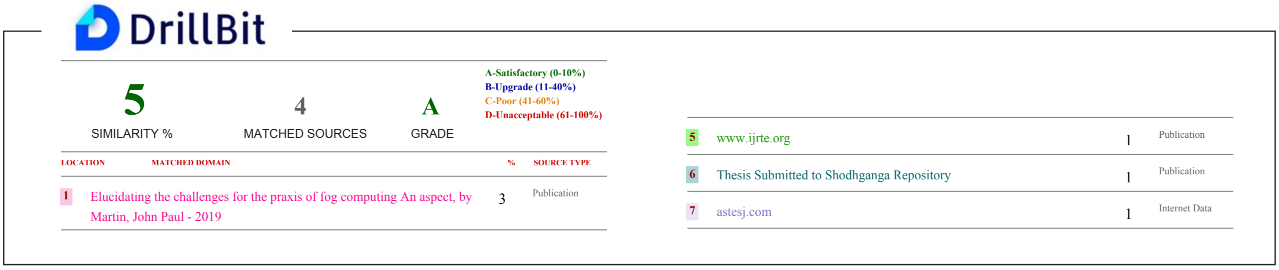
GJEIS Prevent Plagiarism in Publication

DELNET-Developing Library Network, New Delhi in collaboration with BIPL has launched "DrillBit : Plagiarism Detection Software for Academic Integrity" for the member institutions of DELNET. It is a sophisticated plagiarism detection software which is currently used by 700+ Institutions in India and outside. DrillBit is a global checker that uses the most advanced technology to catch the most sophisticated forms of plagiarism, plays a critical function for students and instructors and tag on a fully-automatic machine learning text- recognition system made for detecting, preventing and handling plagiarism and trusted by thousands of institutions across worldwide. DrillBit - Plagiarism Detection Software has been preferred for empanelment with AICTE and NEAT 3.0 (National Education Alliance for Technology) and contributing towards enhanced learning outcomes in India. On the other hand software uses a number of methods to detect AI-generated content, including, checking for repetitive phrases or sentences and AI-generated writing. As part of a larger global organization GJEIS (www.gjeis.com) and DrillBit better equipped to anticipate the foster an environment of academic integrity for educators and students around the globe. DrillBit is GDPR compliant with privacy by design and an uptime of 99.9% and have trust to be the partner in academic integrity (https://www.drillbitplagiarism.com) tool to check the originality and further affixed the similarity index which is {05%} in this case (See below Annexure-I). Thus, the reviewers and editors are of view to find it suitable to publish in this Volume-15, Issue-4, Oct-Dec 2023.

Annexure 15.4.6

Submission Date	Submission Id	Word Count	Character Count
29-Oct-2023	1387732 (DrillBit)	3108	20272

Analyzed Document	Submitter email	Submitted by	Similarity
4.1 VP1_Rajbala_GJEIS Oct to Dec 2023.docx	rajbalasimon@gmail.com	Rajbala Simon	05%



DrillBit

5
SIMILARITY %

4
MATCHED SOURCES

A
GRADE

A-Satisfactory (0-10%)
B-Upgrade (11-40%)
C-Poor (41-60%)
D-Unacceptable (61-100%)

LOCATION	MATCHED DOMAIN	%	SOURCE TYPE
1	Elucidating the challenges for the praxis of fog computing An aspect, by Martin, John Paul - 2019	3	Publication
5	www.ijrte.org	1	Publication
6	Thesis Submitted to Shodhganga Repository	1	Publication
7	astesj.com	1	Internet Data

Reviewers Memorandum

Reviewer's Comment 1: The authors discuss the concept of Fog computing and its key features, which help manage the large volumes of data produced by IoT devices. Fog computing improves response times and reduces the need for expensive bandwidth, making it a more efficient option for organizations. Furthermore, it helps secure sensitive IoT data and enables faster insights, leading to increased business agility and security. The authors also briefly touch upon the performance of different kernel types in the algorithm, but the main focus remains on the benefits of Fog computing.

Reviewer's Comment 2: Fog computing has emerged as a promising trend in the realm of data analysis. This technology offers a decentralized approach to data processing, where computation is executed at the edge of the network, in close proximity to the data source. The potential of fog computing to handle data more efficiently, reduce latency, and enhance overall system performance has led to its growing adoption in both industry and academia. Indeed, this topic is contemporary and adds valuable perspectives to the existing literature.

Reviewer's Comment 3: The study presents the limitations of cloud computing, including minimum mobility, lack of location-awareness, and unreliable latency. The author suggests fog computing as a solution that offers flexible resources and services that extend to devices at the network's outer edges, addressing the impracticalities of cloud computing, particularly for certain applications like IoT. Fog computing offers the computational efficiency of cloud computing with the bandwidth enhancements of a local network. Finally, the study notes the interest of computer scientists and researchers due to fog computing's unique capabilities.

Citation

Rajbala Simon, Laxmi Ahuja,
Akansha Kumari and Saurav Vinod Nair
"Fog Computing: An Innovative Technique for the Quality Improvement in IOT Communication"
Volume-15, Issue-4, Oct-Dec 2023. (www.gjeis.com)

https://doi.org/10.18311/gjeis/2023
Volume-15, Issue-4, Oct-Dec 2023
Online iSSN : 0975-1432, **Print iSSN** : 0975-153X
Frequency : Quarterly, Published Since : 2009

Google Citations: Since 2009
H-Index = 96
i10-Index: 964

Source: https://scholar.google.co.in/citations?user=S47TtNkAAAAJ&hl=en

Conflict of Interest: Author of a Paper had no conflict neither financially nor academically.

Editorial Excerpt

The article has 05% of plagiarism which is the accepted percentage as per the norms and standards of the journal for publication. As per the editorial board's observations and blind reviewers' remarks the paper had some minor revisions which were communicated on a timely basis to the authors (Rajbala et al.), and accordingly, all the corrections were incorporated as and when directed and required to do so. The comments related to this manuscript are noticeably related to the theme "**Fog Computing: An Innovative Technique for the Quality Improvement in IOT Communication**" both subject-wise and research-wise. Fog computing has emerged as a promising solution to tackle the limitations of cloud computing, especially for certain domains such as the Internet of Things (IoT). Fog computing has the potential to revolutionize the way we design and deploy large-scale distributed systems by leveraging the power of edge computing. After comprehensive reviews and the editorial board's remarks, the manuscript has been categorized and decided to publish under the "**View Point**" category.

Acknowledgement

The acknowledgement section is an essential component of academic research papers, as it provides due recognition to all those who contributed their hard work and effort towards the writing of the paper. The primary data analysed and presented in this paper, collected firsthand by Rajbala et al., has been acknowledged and endorsed wherever necessary. The authors express their sincere gratitude to all those who assisted in the research process and made this paper a possibility. Lastly, the reviewers and editors of GJEIS deserve recognition for their pivotal role in publishing this issue, without whom the dissemination of this valuable research would not have been possible.

Disclaimer

All views expressed in this paper are my/our own. Some of the content is taken from open-source websites & some are copyright-free to disseminate knowledge. Those some we/I had mentioned above in the references section and acknowledged/cited as when and where required. The author/s have cited their joint own work mostly, and tables/data from other referenced sources in this particular paper with the narrative & endorsement have been presented within quotes and reference at the bottom of the article accordingly & appropriately. Finally, some of the contents are taken or overlapped from open-source websites for knowledge purposes. Those are some of the i/we mentioned above in the references section. On the other hand, opinions expressed in this paper are those of the authors and do not reflect the views of the GJEIS. The authors have made every effort to ensure that the information in this paper is correct, any remaining errors and deficiencies are solely their responsibility.

Scholastic Seed Inc.
e-Publishing Aggregator & Periodical Mentor