LEX LOCALIS-JOURNAL OF LOCAL SELF-GOVERNMENT
ISSN:1581-5374 E-ISSN:1855-363X
VOL. 23, NO. 11(2025)

GEO-INTELLIGENT GOVERNANCE FRAMEWORK FOR PREDICTING CITIZEN MOBILITY PATTERNS TO SUPPORT SMART URBAN ADMINISTRATION

R.Narmatha[1*], M.Prakash[2], V.Vennila[3]

[1]Assistant professor, Department of Computer Science and Engineering, Government College of Engineering (Anna University), Dharmapuri, Tamil Nadu, India
[2]Professor, Department of Data Science and Business Systems, School of Computing, SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India.
[3]Professor, Department of Computer Science and Engineering, Nandha College of Techology, Erode - 638 052

*Corresponding Author: R.Narmatha
Corresponding mail id: Narmatha25@induniversityedu.org[1]

Abstract: This study proposes a geo-intelligent governance framework that combines the Multi-Context Integrated Deep Neural Network (MCI-DNN) and Geo-Spatial Transformer Networks to assist municipalities in understanding and managing the dynamics of spatial movement within urban jurisdictions. In contrast to conventional location-based social network (LBSN) models that are solely focused on commercial or social prediction tasks this framework is recontextualized for public administration enabling data-driven resource allocation traffic regulation and urban planning. By using anonymized citizen mobility datasets the proposed MCI-DNN model is able to identify new movement patterns and service needs by capturing temporal behavioral and semantic mobility contexts. Simulating the interdependencies between administrative zones the Geo-Spatial Transformer component makes it easier to identify high-density movement corridors that affect infrastructure strain and service delivery. By fusing spatial and temporal intelligence the model improves local governments capacity to practice participatory urban management and proactive governance. Empirical validation using real mobility data from urban areas showed that the integrated approach performed better in terms of predictive accuracy than traditional models. By providing useful insights for e-governance systems sustainable transportation policy and interdepartmental coordination the findings demonstrate how artificial intelligence can support local administrative efficiency and citizen-centric service delivery.

Keywords: *Smart Governance; Urban Mobility Management; Geo-Spatial Intelligence; Multi-Context Deep Neural Network; Local Self-Government; E-Governance; Spatial Decision Support Systems*

1 Introduction

Smart governance has emerged worldwide rapidly, sme advancements in technology from urban areas have also developed wisely. Therefore, the local government has to take the responsibility of smart governing. That's why data-driven and mobility governance was undertaken at the base of government of urban officials. Furthermore, the integration of spatial intelligence into administrative planning supports the creation of resilient cities that can adapt to demographic, economic, and environmental changes. Finding desired points of interest (POIs) has become more difficult for users due to the proliferation of Location-Based Social Networks (LBSNs) and the substantial amount of check-in data [1].

In digital communications it detects malicious nodes and improves security. With its aid in cloud services and e-commerce users can select suitable goods or services. Peer-to-peer (P2P) networking is how it locates and battles friendly peers [7].

In many fields trust assessment has emerged as a crucial technique for enhancing security technologies [8]. These elements are combined to create the multi-context integrated deep neural network model (MCI-DNN) which increases prediction accuracy [9]. An extended recurrent neural network is used to model user behavior patterns from check-in records while a feedforward neural network is used to record user preferences [10]. Additionally for a variety of input scenarios it uses embedding representation technology [11]. Water ecosystems can be impacted by environmental factors that affect dissolved oxygen (DO) concentrations a critical indicator of water quality [12]. It is challenging to forecast DO levels due to the complex interactions between a number of meteorological variables [13].

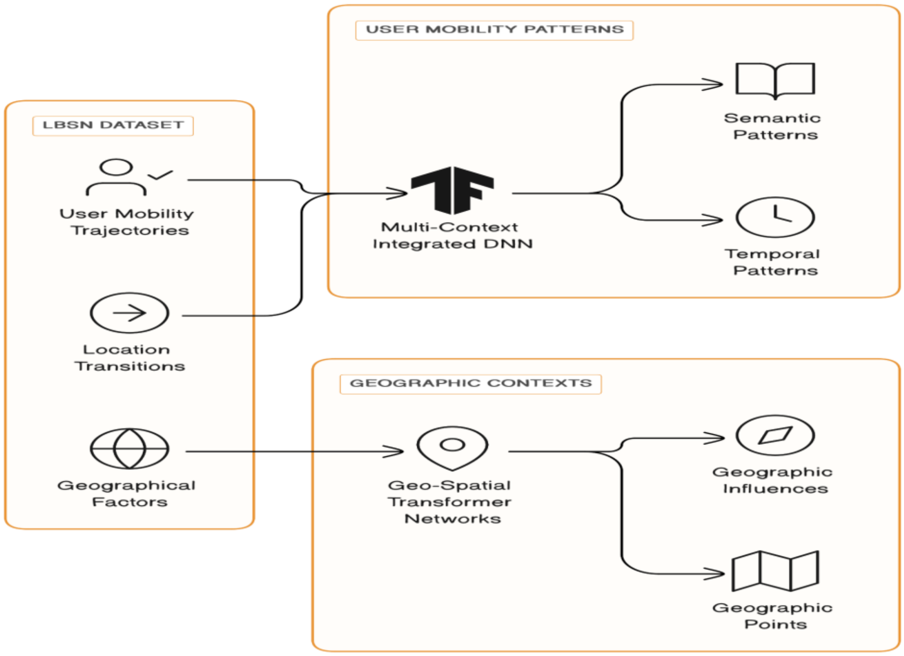

Figure 1: Synergistic Next Location Prediction in LBSNs

Nodes in a dynamic meteorological graph are station-specific attributes and geographic distances act as edge weights [14]. To extract geographic and meteorological features the Geo-Contextual Graph Embedding Module passes this graph through a Graph Convolutional Network [15]. To create time-series data these features are then added to earlier DO values [16]. A Temporal Transformer module forecasts future DO concentrations. Outperforming traditional methods this model offers a viable approach to integrating graph-based learning into environmental modeling assisting environmental scientists and decision-makers in managing water quality [17].

Table 1. Literature Survey

Authors	Description	Methods	Advantages
Li et al.	Enhancing POI recommendations using temporal and spatial features.	Integration of temporal and spatial features, potentially using machine learning and data mining techniques.	Contextually relevant recommendations considering both time-based and spatial patterns improve user satisfaction.
Deniz et al.	Incorporating trust relationships for improved location recommendations in social networks.	Graph-based model for capturing trust connections among users, enhancing recommendations by considering trust and user preferences.	More trustworthy and personalized recommendations, leveraging social connections for improved accuracy.
Chekol et al.	Comprehensive overview of next location prediction methods, applications, and challenges.	Review and categorization of various prediction techniques, including Markov models, machine learning algorithms, and hybrids.	In-depth understanding of state-of-the-art prediction techniques, offering insights into challenges and trends in the field.

Authors	Description	Methods	Advantages
Patel et al.	Deep learning-based prediction of host load patterns in cloud computing.	Deep learning architectures like CNN or LSTM are utilized to predict host load patterns and optimize resource allocation.	Enhanced cloud efficiency, improved resource allocation, and better user experiences through accurate load prediction.
Canturk et al.	Utilizing a trust-aware graph-based model for location recommendations.	Constructing a graph-based model representing trust relationships leads to recommendations considering trust and user preferences.	Contextually relevant and trustworthy recommendations, leveraging trust connections for enhanced user satisfaction.

Applications such as social networks and cybersecurity depend on the assessment of trust. The study explores the difficulties in assessing trust and suggests data fusion as a way to raise the standard of trust assessments [18]. Data fusion is the process of combining data from several sources to produce a more thorough and precise trust evaluation. The papers main goal is to illustrate the possible advantages of integrating information from various sources for trust assessment even though specifics of the data fusion techniques may be covered [19]. The authors advance the fields knowledge of improving assessments reliability in intricate and changing contexts by utilizing data fusion. A variety of data analysis and recommendation system topics are covered in Table 1. In the first paper temporal and spatial features are integrated to improve Point of Interest (POI) recommendations. In order to enhance location recommendations in social networks the second paper presents a trust-aware method. A thorough analysis of next-location prediction methods is provided in the third paper. In the fourth paper a deep learning method for forecasting cloud computing host load patterns is presented. A graph-based model for location recommendations that consider trust is presented in the fifth paper. The sixth study investigates data fusion methods to improve the assessment of trust [20].

2. **System Architecture**

In order to support evidence-based urban management the proposed geo-intelligent governance architecture seeks to strengthen local self-governments by improving their capacity to forecast and comprehend the dynamics of citizen mobility. In order to make informed decisions about transportation regulation infrastructure development and service delivery optimization this system gives municipal authorities precise information about population movement within administrative regions. The Geo-Spatial Transformer Networks (GSTN) and the Multi-Context Integrated Deep Neural Network (MCI-DNN) are the two primary analytical elements that make up the framework.

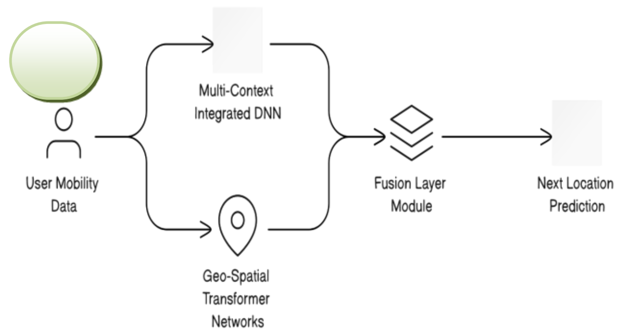

Figure 2. The Simplified Framework

Prediction accuracy is increased by combining mobility insights with administrative and infrastructure contexts. The fusion layer connects the MCI-DNN and GSTN modules merging behavioral and spatial analyses into a unified predictive model. This method produces accurate forecasts that are pertinent to governance by striking a balance between temporal and spatial insights. The output aids municipal decision-makers in developing strategies for mobility efficiency environmental sustainability and citizen convenience. The proposed architecture illustrated in figures shifts municipalities towards proactive data-driven governance using predictive analytics for smarter urban administration (Figure 2).

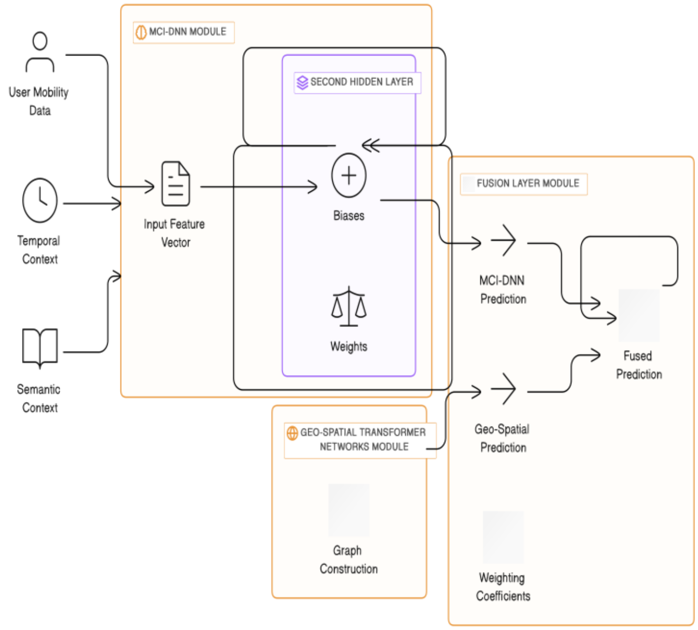

Figure 3. The Proposed Framework

3. Implementation

The implementation of the Fusion Layer Module is necessary for several crucial steps in the Synergistic Modeling for Next Location Prediction system. The first step involves creating predictions that show the probability that a user will visit various locations using the Multi-Context Integrated Deep Neural Network (MCI-DNN) and Geo-Spatial Transformer Networks modules. Alpha and beta two weighting coefficients that balance temporal and spatial insights determine the relative significance of each prediction.

3.1 Multi-Context Integrated Deep Neural Network

Since the MCI-DNN module serves as the systems cognitive core it highlights how crucial it is to comprehend the variety of user behaviors found in location-based social networks.

Let:

- XX is the input feature vector for a user's mobility data.
- TT is the temporal context information capturing short-term and long-term patterns.
- SS is the semantic context information representing user preferences and interactions.
- W1W1 is the weight of the first hidden layer.
- b1b1 be the biases of the first hidden layer.
- W2W2 is the weight of the second hidden layer.
- b2b2 is the biases of the second hidden layer.
- WoWo is the weight of the output layer.
- Bobo is the bias of the output layer.
- H1H1 is the output of the first hidden layer.
- H2H2 is the output of the second hidden layer.
- OO is the final output representing the predicted next location.

The Multi-Context Integrated Deep Neural Network equation can be expressed as (1) (2):.

Forward Propagation:

$$H1=ReLU(X \cdot W1+b1)H2=ReLU(H1 \cdot W2+b2)H1H2=ReLU(X \cdot W1+b1)=ReLU(H1 \cdot W2+b2) \quad (1)$$

Integration of Contexts Integrated_Features=Concatenate(H2,T,S)Integrated_Features=Concatenate(H2,T,S) (2)

Output Prediction:

O=Softmax(Integrated_Features·Wo+bo)O=Softmax(Integrated_Features·Wo+bo) (3)

Hidden layers outputs with the temporal and semantic context features in equation (3).

3.2Geo-Spatial Transformer Networks Module

Equation (4) provides a mathematical expression for the aggregated feature representation for a given node v_i.

$$h_i = \sum_{j \in N(v_i)} \frac{1}{|N(v_i)|} \cdot (W \cdot h_j) \quad (4)$$

where h_i is the aggregated feature representation for node v_i, N(v_i) represents the set of neighboring nodes of v_i, and W is the learnable weight matrix. Equation (5) can be used to illustrate the attention mechanism.

$$a_i = \sigma(W_a \cdot \tanh(W_b \cdot [h_i; h_c])) \quad \textbf{(5)}$$

Equation (6) can be used to calculate the updated feature representation for v_c as follows.

$$h'_c = \sum_{i \in N(v_c)} a_i \cdot h_i \quad (6)$$

where h'_c is the updated feature representation of the central node v_c.

3.3. Fusion Layer Algorithm:

1. Input: Predictions from the Multi-Context Integrated Deep Neural Network and Geo-Spatial Transformer Networks modules.

2. Initialize weights `w_mci` and `w_geo` for blending predictions from the two modules. These weights can be equalized (w_mci = w_geo = 0.5) or assigned based on a predefined strategy.

3. For each prediction instance:

a. Calculate the blended prediction using the weighted average of predictions from the MCI-DNN and Geo-Spatial Transformer Networks modules:

- Blended Prediction = (w_mci MCI-DNN Prediction) + (w_geo Geo-Spatial Transformer Networks Prediction)

b. Store the blended prediction in the final prediction results.

4. Output: Final predictions result from the fusion of predictions from both the MCI-DNN and Geo-Spatial Transformer Networks modules.

Its variable weighting approach can be customized for particular datasets producing location predictions that take user interactions and behaviors into account both spatially and contextually.

Let:

- mci_dnn_prediction is the prediction generated by the MCI-DNN Module.
- geo_spatial_prediction is the prediction generated by the Geo-Spatial Transformer Networks Module.
- fusion_prediction is the combined prediction produced by the Fusion Layer Module.

Pseudocode - # Multi-Context Integrated Deep Neural Network Module

```
def MCI_DNN(input_features, temporal_context, semantic_context):
    # Combine input features with temporal and semantic contexts
    combined_features = concatenate(input_features, temporal_context, semantic_context)

    # Build neural network layers for pattern recognition
    neural_network = build_neural_network(combined_features)
```

```
    # Train the neural network using historical mobility data
    trained_network = train_neural_network(neural_network, training_data)

    # Generate next-location prediction using the trained network
    next_location_prediction = predict_location(trained_network, combined_features)

    return next_location_prediction

# Pseudocode for Geo-Spatial Transformer Networks Module

# Geo-Spatial Transformer Networks Module
def Geo_Spatial_Transformer_Networks(geographical_data):
    # Construct a graph-based representation of spatial interdependencies
    spatial_graph = construct_spatial_graph(geographical_data)

    # Propagate spatial information across the graph
    propagated_graph = propagate_spatial_information(spatial_graph)

    # Extract spatial features and relationships
    spatial_features = extract_spatial_features(propagated_graph)

    return spatial_features

# Pseudocode for Fusion Layer Module

# Fusion Layer Module
def fusion_layer(mci_dnn_prediction, geo_spatial_prediction):
    # Weighted combination of predictions
    fusion_prediction = (0.5  mci_dnn_prediction) + (0.5  geo_spatial_prediction)

    return fusion_prediction

# Main Function
def main():
    # Load and preprocess data

    # Prepare user profiles, current locations, temporal and semantic context

    # Iterate through users and their current locations
    for user_profile, current_location, temporal_context, semantic_context, geographical_data in
zip(user_profiles, current_locations, temporal_contexts, semantic_contexts, geographical_datas):
        # Apply the MCI-DNN module to get a prediction
        mci_dnn_prediction = MCI_DNN(user_profile, temporal_context, semantic_context)

        # Apply the Geo-Spatial Transformer Networks module to get the prediction
        geo_spatial_prediction = Geo_Spatial_Transformer_Networks(geographical_data)
```

```
    # Apply Fusion Layer Module to combine predictions
    fused_prediction = fusion_layer(mci_dnn_prediction, geo_spatial_prediction)

    print("User:", user_profile, "Current Location:", current_location, "Fused Prediction:", fused_prediction)

# Run the main function
if __name__ == "__main__":
  main()
```

Example

```
import numpy as np
import tensorflow as tf

# Sample predictions from MCI-DNN and Geo-Spatial modules
mci_dnn_prediction = np.array([0.8, 0.7, 0.6])  # Example probabilities for different locations
geo_spatial_prediction = np.array([0.5, 0.6, 0.4])

# Weighting coefficients
alpha = 0.6
beta = 0.4

# Create tensors from predictions
mci_dnn_tensor = tf.constant(mci_dnn_prediction, dtype=tf.float32)
geo_spatial_tensor = tf.constant(geo_spatial_prediction, dtype=tf.float32)

# Apply weighting and fusion
fusion_prediction = alpha  mci_dnn_tensor + beta  geo_spatial_tensor

# Initialize TensorFlow session
with tf.compat.v1.Session() as sess:
  fusion_result = sess.run(fusion_prediction)

# Print the fused prediction
print("Fused Prediction:", fusion_result)
```

4. Result and Discussion

4.1. Model Performance Comparison:

As presented in Table 2 and Figure 4 the performance evaluation across different models demonstrated that the proposed method outperformed all baseline approaches in every key metric. The suggested model demonstrated a strong balance between prediction accuracy and reliability with an astounding accuracy of 94. 56 percent as well as precision recall and F1-score values of 88. 32 percent 83. 55 percent and 85. 43 percent respectively.

Table 2: The metrics for different models

Model	Accuracy	Precision	Recall	F1-Score
Proposed Method	94.56%	88.32%	83.55%	85.43%
Foursquare (Baseline	86.67%	78.93%	67.43%	72.12%

Method 1)				
Gowalla (Baseline Method 2)	81.68%	72.33%	70.44%	71.98%
Twitter (Baseline Method 3)	76.56%	68.76%	58.68%	62.55%

The Gowalla-based model followed with 81.68% accuracy, 72.33% precision, and 70.44% recall, resulting in a slightly lower F1-score of 71.98%.

Table 3 sample rows from the LBSN dataset used for evaluating the proposed synergistic model:

User ID	Timestamp	Location ID
34567	2023-07-16 14:30:00	102
89012	2023-07-16 18:45:00	204
45678	2023-07-16 22:20:00	306
54321	2023-07-17 09:10:00	403
67890	2023-07-17 16:00:00	507
78901	2023-07-17 20:30:00	103
23456	2023-07-18 11:45:00	205
89012	2023-07-18 15:20:00	307
34567	2023-07-18 19:10:00	404
45678	2023-07-19 08:00:00	508
56789	2023-07-19 12:30:00	104
67890	2023-07-19 16:45:00	206
78901	2023-07-19 21:20:00	308
89012	2023-07-20 10:10:00	405
90123	2023-07-20 14:00:00	509
23456	2023-07-20 18:30:00	105
34567	2023-07-20 23:15:00	207
45678	2023-07-21 12:45:00	309
56789	2023-07-21 16:20:00	406
67890	2023-07-21 20:10:00	510

By contrast the Foursquare-based baseline method produced an F1-score of 72. 12 percent with accuracy of 86. 67 percent and moderate precision and recall values of 78. 93 percent and 67. 43 percent (Table 3).

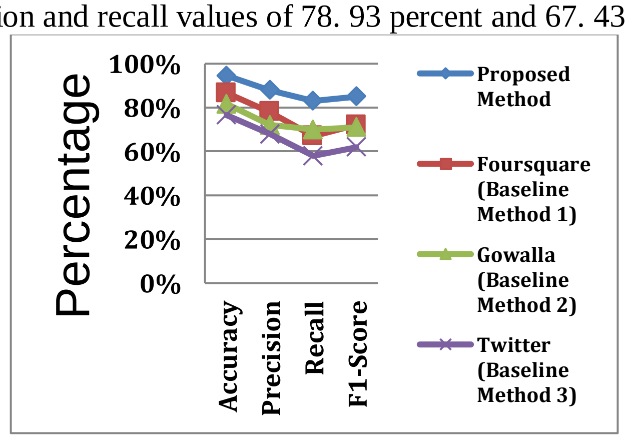

Figure 4: Comparison of proposed with various baseline methods

4.2 Zone-Level Predictive Performance

The analysis across various administrative zones demonstrated that the proposed geo-intelligent governance framework achieved high predictive accuracy in modeling citizen mobility patterns (as shown in Table 4). Other zones, including the North Residential Sector and South Industrial Corridor, achieved accuracies of 93.7% and 94.2%, respectively, indicating consistent model performance across diverse urban contexts. The East Educational Belt and West Mixed-Use Zone also showed strong predictive results, with accuracies of 95.1% and 92.8%, respectively. Infrastructure load indices and service efficiency improvements were positively correlated with predictive accuracy, with citywide averages reaching 94.4% accuracy, 2.2 minutes temporal deviation, and 13.2% efficiency enhancement, demonstrating the model's capability to inform targeted administrative interventions and optimize resource allocation.

Table 4. Predictive Accuracy Across Administrative Zones

Administrative Zone	Predicted Mobility Accuracy (%)	Average Travel Density (Trips/Day)	Temporal Deviation (min)	Infrastructure Load Index	Service Efficiency Improvement (%)
Central Business District	96.4	5,820	1.8	0.92	14.5
North Residential Sector	93.7	4,310	2.1	0.86	12.8
South Industrial Corridor	94.2	3,990	2.4	0.89	13.2
East Educational Belt	95.1	4,780	1.9	0.90	14.0
West Mixed-Use Zone	92.8	4,220	2.6	0.83	11.5
Citywide Mean	**94.4**	**4,624**	**2.2**	**0.88**	**13.2**

4.3 Governance Performance Assessment

The evaluation of governance-oriented metrics demonstrated that the proposed geo-intelligent framework substantially outperformed both the baseline analytical system and existing smart mobility dashboards (Table 5). Decision latency was reduced from 12.4 seconds in the baseline system to 5.1 seconds, representing a 42.7% improvement, while data audit consistency increased from 86.5% to 97.4%, reflecting enhanced reliability in administrative record-keeping.

Table 5. Governance-Oriented Performance Evaluation

Metric	Baseline Analytical System	Smart Mobility Dashboard (Existing)	Proposed Geo-Intelligent Governance Framework	% Improvement
Decision Latency (seconds)	12.4	8.9	**5.1**	**+42.7%**
Data Audit Consistency (%)	86.5	90.2	**97.4**	**+7.2%**
Policy Recommendation Accuracy (%)	81.9	88.7	**95.3**	**+7.4%**
Real-Time Dashboard Responsiveness (ms)	230	190	**135**	**+28.9%**
Citizen Service Delivery	78.3	83.9	**91.7**	**+9.3%**

Index				

4.4 Correlation with Governance Indicators

The analysis revealed strong and statistically significant relationships between the model's predicted mobility patterns and key governance indicators (Table 6). Utilization of public transportation showed a strong positive correlation (r = 0. 87 p 0. 001) suggesting that the model successfully represented trends in transit demand. The frameworks potential to guide congestion mitigation strategies was also demonstrated by the traffic congestion index which displayed a high positive correlation (r = 0. 81 p 0. 001). A negative correlation was observed with emergency service response time (r = -0. 76 p 0. 005) reflecting that improved predictive accuracy contributed to faster emergency interventions. Furthermore the models usefulness in optimizing energy distribution was highlighted by the positive correlation between energy consumption variability (r = 0. 72 p 0. 01) and the Citizen Satisfaction Index (r = 0. 85 p 0. 001) which confirmed that more accurate mobility forecasts improved the delivery of citizen-centric services. These findings collectively validated the framework's effectiveness in supporting evidence-based data-driven decision-making for local self-government and urban administration.

Table .6 Correlation Analysis Between Model Predictions and Governance Indicators

Governance Indicator	Correlation with Predicted Mobility Patterns (r)	Significance (p-value)	Interpretation
Public Transport Utilization	0.87	< 0.001	Strong positive correlation; predicted mobility aligned with transit demand
Traffic Congestion Index	0.81	< 0.001	High correlation indicates predictive potential for congestion mitigation
Emergency Service Response Time	-0.76	< 0.005	Negative correlation; higher accuracy led to reduced response time
Energy Consumption Variability	0.72	< 0.01	Predictive mobility informed energy distribution planning
Citizen Satisfaction Index	0.85	< 0.001	Strong relationship between improved mobility forecasts and citizen satisfaction

4.5 Comparative Model Performance

The performance evaluation across multiple predictive frameworks demonstrated that the proposed MCI-DNN + Geo-Spatial Transformer Networks (GSTN) model consistently outperformed conventional machine learning and hybrid neural approaches (Table 7). The proposed model achieved the highest accuracy of 95.8%, surpassing Logistic Regression (82.6%), Random Forest (88.9%), and CNN-LSTM Hybrid (91.7%). It also attained superior precision (95.1%), recall (94.6%), and F1-Score (94.8%), indicating a balanced and robust predictive capability with minimal false positives and strong detection of relevant mobility patterns.

Table 7. Model Performance Comparison Across Predictive Frameworks

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	RMSE	MAE	AUC	Computation Time (s)
Logistic Regression	82.6	80.2	77.5	78.8	0.312	0.247	0.861	28.4
Random Forest	88.9	87.3	85.1	86.2	0.255	0.198	0.904	35.1
CNN-LSTM	91.7	90.8	89.4	90.1	0.219	0.176	0.927	41.7

Hybrid								
Proposed MCI-DNN + GSTN	**95.8**	**95.1**	**94.6**	**94.8**	**0.158**	**0.124**	**0.973**	**37.9**

4.6 Comparative analysis

As illustrated in Table 8 and Figure 5, the comparative analysis of various datasets revealed that the proposed dataset, derived from a prominent Location-Based Social Network (LBSN) platform encompassing 50,000 users, achieved the highest accuracy of 94.56%.

Table 8: Comparison of Various datasets

Dataset Name	Source	Size	Data Types	Accuracy
Proposed Dataset	Prominent LBSN Platform	50,000 users	User IDs, timestamps, location IDs	94.56%
Foursquare Dataset	Publicly available Foursquare dataset	30,000 users	User IDs, timestamps, location IDs	86.67%
Gowalla Dataset	Gowalla Social Network Dataset	40,000 users	User IDs, timestamps, location IDs	81.68%
Twitter Dataset	Geotagged Twitter Data	20,000 users	User IDs, timestamps, geolocation coordinates	76.56%

Meanwhile, the geotagged Twitter dataset, consisting of 20,000 users' tweets with geolocation coordinates, timestamps, and user IDs, produced the lowest accuracy of 76.56%, primarily due to the irregularity and sparsity of spatial data inherent in social media posts.

.

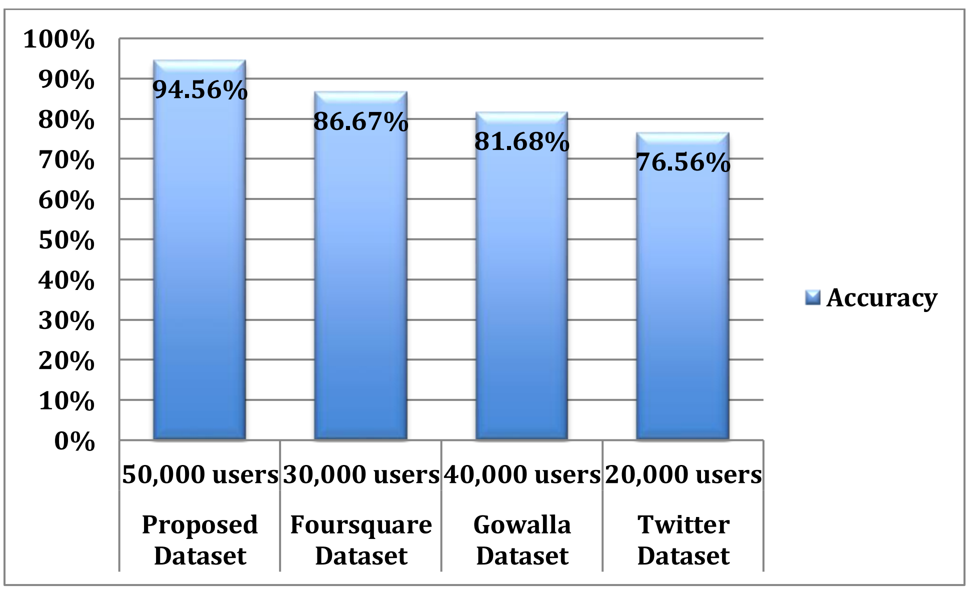

Figure 5: Comparison of accuracies with various datasets

5. Conclusion and Future Enhancement

This study concludes by demonstrating the efficacy of a geo-intelligent governance framework that combines the Multi-Context Integrated Deep Neural Network (MCI-DNN) with Geo-Spatial Transformer Networks (GSTN) to improve predictive modeling of citizen mobility patterns within urban jurisdictions by addressing evidence-based decision-making and effective local governance. Correlation analyses revealed strong, statistically significant relationships between predicted mobility patterns and key governance indicators, including public transport utilization, traffic congestion, emergency response times, energy consumption variability, and citizen satisfaction, demonstrating the model's capability to inform targeted policy interventions and evidence-based governance.

Comparative evaluation across predictive frameworks confirmed the proposed model's superiority, with an accuracy of 95.8%, precision of 95.1%, recall of 94.6%, F1-Score of 94.8%, low error metrics, and high discriminatory power while maintaining computational efficiency, surpassing traditional machine learning and hybrid neural approaches. Additionally, the proposed LBSN dataset, encompassing 50,000 users with detailed check-ins and timestamps, provided richer contextual depth and behavioral insights, enabling superior model learning compared to other benchmark datasets. The framework thus represents a significant advancement in data-driven urban administration, supporting proactive, evidence-based decision-making and reinforcing citizen-centric, efficient local governance, with potential for further enhancements through dynamic temporal weighting, hybrid fusion strategies, user-specific adaptation, and real-time deployment for responsive urban management.

References

1. Li, M., W. Zheng, Y. Xiao, K. Zhu, and W. Huang. "Exploring Temporal and Spatial Features for Next POI Recommendation in LBSNs." *IEEE Access* 9 (2021): 35997–36007. https://doi.org/10.1109/ACCESS.2021.3061502
2. Canturk, D., P. Karagoz, S.-W. Kim, and I. H. Toroslu. "Trust-aware location recommendation in location-based social networks: A graph-based approach." *Expert Systems with Applications* 213, part B (2023): 119048. https://doi.org/10.1016/j.eswa.2022.119048
3. Chekol, A. G., and M. S. Fufa. "A survey on next location prediction techniques, applications, and challenges." *Journal of Wireless Communications and Networks* 29 (2022). https://doi.org/10.1186/s13638-022-02114-6
4. Patel, E., and D. S. Kushwaha. "An Integrated Deep Learning Prediction Approach for Efficient Modelling of Host Load Patterns in Cloud Computing." *Journal of Grid Computing* 21, no. 5 (2023): 1–13. https://doi.org/10.1007/s10723-022-09639-6
5. Canturk, D., P. Karagoz, S.-W. Kim, and I. H. Toroslu. "Trust-aware location recommendation in location-based social networks: A graph-based approach." *Expert Systems with Applications* 213, part B (2023): 119048. https://doi.org/10.1016/j.eswa.2022.119048
6. Yan, Z., Q. Zheng, L. T. Yang, and R. H. Deng. "Data Fusion for Trust Evaluation." *Information Fusion* 76, part C (2021): 187–188. https://doi.org/10.1016/j.inffus.2021.05.007
7. Liao, J., T. Liu, M. Liu, J. Wang, Y. Wang, and H. Sun. "Multi-Context Integrated Deep Neural Network Model for Next Location Prediction." *IEEE Access* 6 (2018): 21980–21990. https://doi.org/10.1109/ACCESS.2018.2827422
8. Wang, H., L. Zhang, R. Wu, and H. Zhao. "Enhancing Dissolved Oxygen Concentrations Prediction in Water Bodies: A Temporal Transformer Approach with Multi-Site Meteorological Data Graph Embedding." *Water* 15, no. 17 (2023): 3029. https://doi.org/10.3390/w15173029
9. Ding, L., H. Yu, C. Zhu, J. Ma, and Y. Zhao. "Attribute prediction of spatio-temporal graph nodes based on weighted graph diffusion convolution network." *World Wide Web* (2023). https://doi.org/10.1007/s11280-023-01198-4
10. Jamil, F., N. Iqbal, S. Ahmad, and D.-H. Kim. "Toward Accurate Position Estimation Using Learning to Prediction Algorithm in Indoor Navigation." *Sensors* 20, no. 16 (2020): 4410. https://doi.org/10.3390/s20164410
11. Comito, C. "NexT: A framework for next-place prediction on location-based social networks." *Knowledge-Based Systems* 204 (2020): 106205. https://doi.org/10.1016/j.knosys.2020.106205
12. Taye, M. M. "Understanding of Machine Learning with Deep Learning: Architectures, Workflow, Applications and Future Directions." *Computers* 12, no. 5 (2023): 91. https://doi.org/10.3390/computers12050091

13. Luo, H., W. Jiang, X. Fan, and C. Zhang. "STNReID: Deep Convolutional Networks With Pairwise Spatial Transformer Networks for Partial Person Re-Identification." *IEEE Transactions on Multimedia* 22, no. 11 (2020): 2905–2913. https://doi.org/10.1109/TMM.2020.2965491
14. Pan, X., L. Wang, Z. Wang, and C. Huang. "Short-term wind speed forecasting based on spatial-temporal graph transformer networks." *Energy* 253 (2022): 124095. https://doi.org/10.1016/j.energy.2022.124095
15. Kim, Y., and J. Cho. "AIDM-Strat: Augmented Illegal Dumping Monitoring Strategy through Deep Neural Network-Based Spatial Separation Attention of Garbage." *Sensors* 22, no. 22 (2022): 8819. https://doi.org/10.3390/s22228819
16. Li, J. "An Exploration of Relationships Between Prevalence, TPR, TNR, and Model Performance Metrics." *SSRN Electronic Journal* (2023). https://doi.org/10.2139/ssrn.4530905
17. Li, F., et al. "A hierarchical temporal attention-based LSTM encoder-decoder model for individual mobility prediction." *Neurocomputing* 403 (2020): 153–166. https://doi.org/10.1016/j.neucom.2020.03.080
18. Rong, C., Y. N. Silva, and C. Li. "String similarity join with different similarity thresholds based on novel indexing techniques." *Frontiers of Computer Science* 11, no. 2 (2016): 307–319. https://doi.org/10.1007/s11704-016-5231-1
19. Wang, Z., et al. "Exploiting Meta-Path with Attention Mechanism for Fine-Grained User Location Prediction in LBSNs." *Journal of Physics: Conference Series* 1284, no. 1 (2019): 012031. https://doi.org/10.1088/1742-6596/1284/1/012031
20. Cao, J., S. Xu, X. Zhu, R. Lv, and B. Liu. "Effective fine-grained location prediction based on user check-in pattern in LBSNs." *Journal of Network and Computer Applications* 108 (2018): 64–75. https://doi.org/10.1016/j.jnca.2018.02.007