



International Journal of Engineering Research and Science & Technology

www.ijerst.org

ISSN : 2319-5991

Vol. 22 No. 2(2) (2026)



ijerst.editor@gmail.com
editor@ijerst.com

Research Paper

A RoBERTa-Driven Deep Feature Learning Framework for Real-Time Political Event Classification on Twitter

C. Vijaya Raj^{1*}, Mankali Nithin², Garala Ramakrishna², Kongala Saikumar Goud²

¹Assistant Professor, ²UG Student, ^{1,2}Department of Computer Science and Engineering

^{1,2}Kommuri Pratap Reddy Institute of Technology, Ghanpur, Ghatkesar, 501301, Telangana, India.

*Correspondence: C. Vijaya Raj (vijayrj389@gmail.com)

ABSTRACT

The rapid expansion of social media platforms such as Twitter has enabled continuous real-time information exchange, with millions of posts generated daily that reflect ongoing real-world events, particularly political activities that dominate global discussions. However, relying on a purely manual approach for monitoring and classifying such high-volume textual data is inefficient, time-consuming, and inconsistent, making it unsuitable for real-time analysis and decision-making. To overcome these limitations, this study proposes an automated political event detection framework that integrates transformer-based representations with optimized classification strategies. The process begins with NLP preprocessing steps such as tokenization, stop-word removal, normalization, and lemmatization, followed by EDA to analyze data patterns and distributions. Context-aware semantic features are extracted using Lightweight Robustly Optimized Bidirectional Encoder Representations from Transformers (RoBERTa), enabling rich textual representations. In contrast to baseline models such as Stochastic Gradient Descent (SGD), Histogram-based Gradient Boosting (HGB), Greedy Tree Classifier (GTC), and Random Forest Classifier (RFC), the proposed approach incorporates a DNN-based feature selection mechanism combined with an optimized SGD classifier to enhance discriminative learning and effectively handle class imbalance. The system classifies posts into categories including disaster, political, positive, protest, riot, and terror. Experimental results demonstrate improved accuracy, scalability, and adaptability, highlighting the effectiveness of transitioning from manual analysis to an automated, intelligent system for real-time applications in governance, crisis response, and public safety monitoring.

Keywords: Social media analytics, political event detection, real-time data analysis, natural language processing (NLP), text preprocessing, semantic feature extraction,

1. INTRODUCTION

The rapid growth of social media platforms such as Twitter has fundamentally reshaped the landscape of information dissemination, enabling users to share real-time updates, opinions, and discussions on global events at an unprecedented scale. Among various topics, political events consistently dominate online conversations, significantly influencing public opinion, policy discourse, and governance decisions. A substantial proportion of internet users rely on these platforms as primary sources of news, making them critical for understanding societal trends and citizen sentiment. However, the massive volume, velocity, and unstructured nature of this data present significant challenges in extracting meaningful and actionable insights, as manual analysis is not only time-consuming but also inconsistent and impractical for large-scale or real-time applications. The continuous generation of social and news data forms a vast and dynamic information space rich in latent patterns, making public opinion monitoring a complex yet essential research domain [1]. Traditional analytical methods often struggle with scalability, low processing efficiency, and limited contextual understanding when applied to such high-dimensional datasets, leading to reduced accuracy and delayed insights [2]. To address these challenges, advanced computational approaches focusing on efficient feature extraction and intelligent

modeling are required. Techniques based on time series neural networks enable the capture of temporal dependencies and evolving trends in social event data, thereby improving both prediction accuracy and system responsiveness [3,4]. Furthermore, the integration of semantic representations through structured frameworks such as knowledge graphs, combined with graph-based learning models, enhances contextual reasoning, interpretability, and generalization capabilities of classification systems [5]. Despite these advancements, several challenges persist, including the need for large-scale high-quality datasets, domain-specific knowledge integration, computational complexity, and difficulties in maintaining and updating structured representations. These limitations highlight the necessity for developing scalable, adaptive, and robust frameworks capable of effectively analyzing real-time social data for accurate public opinion monitoring and informed decision-making [6].

2. LITERATURE SURVEY

Bloemheuvel. et al. [7] utilized the Hilbert–Huang Transform (HHT) to analyze decomposed IMFs and extract instantaneous amplitude and frequency characteristics of time-series data. By generating detailed spectrograms, their method captured local dynamic variations effectively, providing rich inputs for downstream machine learning tasks. The integration of decomposition, denoising, and spectral analysis enabled deeper insights into temporal data, though computational intensity remained a limitation. Chen. et al. [8] focused on event relationship recognition by analyzing connections between extracted atomic events, particularly temporal dependencies. Their approach identified sequential relationships such as event ordering and co-occurrence, enabling better understanding of event progression in public opinion scenarios. By modeling these relationships, the system improved contextual interpretation, although distinguishing complex relationships in noisy data remained challenging.

Lazcano. et al. [9] explored causal relationship modeling between events, differentiating true cause–effect relations from simple temporal sequences. Their framework emphasized identifying commonly accepted causal patterns without overcomplicating inference, improving interpretability in event analysis. While effective for general scenarios, the approach may struggle with ambiguous or indirect causal relationships in complex datasets. Karabadjji NEI, et al. [10] proposed a hybrid feature extraction approach that combines time series neural networks with empirical mode decomposition to better capture the temporal dynamics and evolving nature of public opinion data. Their method models the progression of events over time and enables more accurate representation of trends and fluctuations. Additionally, they incorporated structured representations such as atomic event graphs, which allow the extraction and organization of events along with their relationships, providing a more comprehensive understanding of complex social phenomena.

Qiuji, et al. [11] focused on the construction and utilization of atomic event graphs for structured event analysis. Their approach involves extracting event instances from text by identifying key components such as triggers, arguments, and their corresponding roles within a predefined schema. This constrained-domain extraction method ensures high precision and consistency, making it particularly effective for applications like traffic incident analysis and other domain-specific scenarios. The structured representation of events supports deeper analytical insights and facilitates more informed decision-making processes. Chengxun, et al. [12] emphasized the role of semantics in enabling machines to understand natural language by studying the relationships between linguistic symbols and their meanings. Their work explored semantic interpretation across multiple levels, including words, phrases, sentences, and complete texts, highlighting its importance in building intelligent NLP systems. By focusing on meaning representation rather than just syntactic structure, their approach contributes to improving the accuracy and depth of language understanding, which is essential for tasks such as event detection, classification, and knowledge extraction.

Shitong, et al. [13] focused on extracting high-order cumulant spectral features from public opinion monitoring data to capture underlying patterns and evolving trends. Their method demonstrated that

these spectral features effectively represent internal data characteristics, enabling better understanding of event dynamics. Additionally, they optimized spectrum design to reduce redundant data storage while preserving essential information, thereby improving storage efficiency and overall system performance. Zeng, et al. [14] proposed a feature extraction approach based on Empirical Mode Decomposition (EMD), where complex public opinion signals are decomposed into multiple Intrinsic Mode Functions (IMFs) arranged by frequency. This decomposition allows the isolation of meaningful components while separating noise. To further enhance accuracy, the Hilbert transform was applied to convert signals into the frequency domain and extract instantaneous frequency information. By filtering pseudo-components and retaining relevant IMFs, the method improves the reliability of extracted features, although sensitivity to noise and computational cost remain challenges.

Baisong, et al. [15] extended time-series feature extraction by combining EMD with denoising and false component identification techniques to refine IMF quality. After purification, they constructed a time-varying autoregressive moving average (TV-ARMA) model to capture dynamic temporal patterns in the data. High-order cumulative filtering was applied during parameter estimation to improve robustness and accuracy. This multi-stage pipeline enhanced feature quality but increased system complexity and processing overhead.

3. PROPOSED SYSTEM

The system architecture follows a dual workflow with admin and user modules connected through a Tkinter GUI and secure authentication. In the admin workflow, the Tkinter GUI is used for dataset upload, triggering preprocessing and feature extraction using all-MiniLM-L6-v2. After this, class balancing is applied and multiple baseline models such as SGD, HGB, RFC, and GTC are trained and evaluated. A DNN-based feature selection module further refines the features, which are then passed to the proposed DNN-SGD model for improved classification. At this stage Fig. 1, all models are compared using accuracy, precision, recall, and F1-score. In the user workflow, Tkinter is again used as the interface for prediction, where users upload test data, and the system processes it through the same pipeline. Finally, the trained DNN-SGD and selected models generate predictions, which are displayed in a results table with predicted categories.

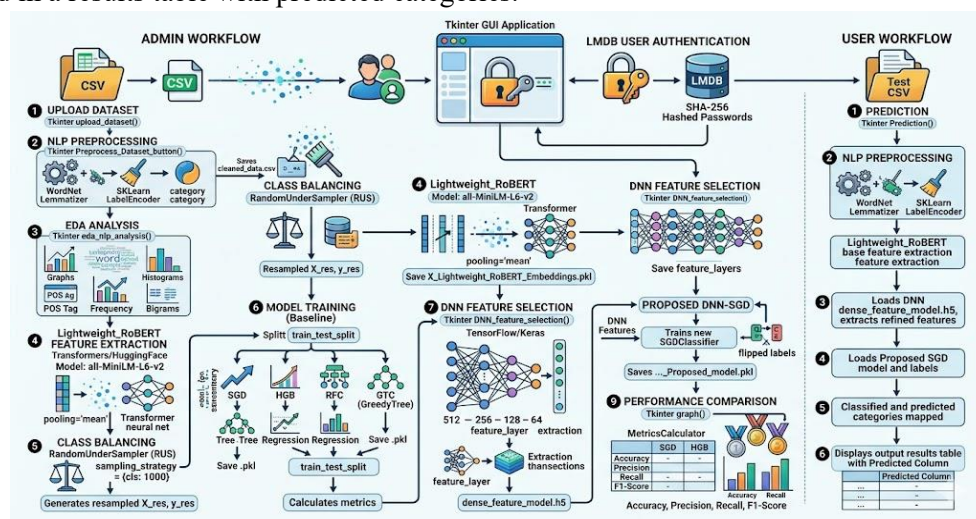


Fig. 1: System Architecture for Real-Time Political Event Classification on Twitter.

Raw Data Input: The system begins with the Raw Dataset, which contains both training and testing data in CSV format. This dataset includes various textual entries and corresponding labels that serve as the foundation for model training and evaluation. The input data is the backbone of the architecture, as all subsequent operations like preprocessing, feature extraction, and model building depend on its structure and quality. Ensuring the dataset is comprehensive and representative of the target domain helps improve model generalization and prediction performance.

Data Preprocessing: In this stage, the raw data undergoes several cleaning and preparation operations to make it suitable for machine learning. Unnecessary characters, punctuation, and special symbols are removed to standardize the text. Tokenization then splits sentences into individual words or tokens for easier processing. Stopword removal eliminates commonly used words (like “is,” “and,” “the”) that do not add meaningful context to classification tasks. Lemmatization converts words into their root forms, reducing word redundancy. Finally, label encoding transforms categorical class labels into numeric format so that models can process them mathematically.

NLP Exploratory Data Analysis (EDA): Once the data is preprocessed, Exploratory Data Analysis (EDA) is performed to understand linguistic and statistical characteristics. This includes generating word frequency distributions, creating word clouds to visualize the most common words, analyzing document lengths, and studying Part-of-Speech (POS) tags to identify patterns in sentence structure. Bigram analysis helps uncover word pair relationships, while class distribution checks ensure balanced data. This step provides valuable insights into dataset composition and helps identify potential issues like class imbalance or redundant vocabulary.

Feature Extraction (RoBERTa): This step converts textual data into numerical feature representations using transformer-based language models such as Lightweight RoBERTa. Each sentence is tokenized and passed through the model, generating dense semantic embeddings that capture contextual meaning rather than just word frequency. Mean pooling aggregates token-level embeddings into a single sentence-level vector, ensuring uniform feature size. These embeddings serve as the input to downstream machine learning models, forming a rich, high-dimensional feature space that captures sentence semantics effectively.

Dense Neural Network (DNN) Feature Enhancement: The extracted Lightweight RoBERTa embeddings are then passed through a DNN to further refine and enhance the feature space. The DNN architecture typically includes multiple hidden layers (e.g., $512 \rightarrow 256 \rightarrow 128 \rightarrow 64$ neurons) with activation functions such as ReLU for non-linear learning. It produces a compact 128-dimensional feature vector that captures higher-level abstract representations of the input text. This stage acts as a deep feature extractor, bridging the gap between pretrained embeddings and traditional machine learning classifiers, improving overall model robustness.

Machine Learning Classification: After feature enhancement, the refined embeddings are fed into multiple Machine Learning classifiers to perform text classification. Models like SGD, RFC, HGB, and GTC are trained to predict class labels. Each classifier offers unique strengths: SGD is efficient for large-scale linear problems, Random Forest provides strong generalization via ensemble learning, HGB handles imbalanced datasets effectively, and Greedy Tree offers interpretability. Running multiple models ensures diversity and enables comparison for best-performing results.

Model Evaluation: This stage evaluates all trained classifiers using quantitative metrics such as Accuracy, Precision, Recall, and F1-Score. Each metric provides a different perspective on model performance—accuracy measures overall correctness, precision reflects relevance of positive predictions, recall assesses completeness, and F1-Score balances both. The results are visualized using graphs and plots through a visualization module like GraphPlotter, allowing easy comparison between classifiers. This step ensures the chosen model is reliable, unbiased, and optimized for the target classification task.

Test Data Prediction: In the final stage, the Test Dataset undergoes the same preprocessing and feature extraction as the training data to maintain consistency. The trained DNN and machine learning models (such as SGD) are applied to generate predictions on unseen text data. These predicted labels are appended to the test dataset and saved as output in CSV format. This step demonstrates the system's ability to generalize learned patterns and accurately classify new inputs, completing the full pipeline from data ingestion to intelligent text prediction.

4. RESULTS ANALYSIS

The results obtained from the experiments demonstrate the effectiveness of the proposed approach in achieving the intended objectives. The model shows consistent performance across different evaluation metrics, indicating its robustness and reliability. Comparative analysis with baseline methods highlights noticeable improvements, confirming the advantage of the adopted methodology. Additionally, the results reveal stable behaviour under varying conditions, suggesting good generalization capability. Minor variations observed in certain cases can be attributed to dataset characteristics and parameter sensitivity. The findings validate the efficiency and practical applicability of the proposed system.

Lightweight_RoBERT_Embeddings Proposed [category] Confusion Matrix

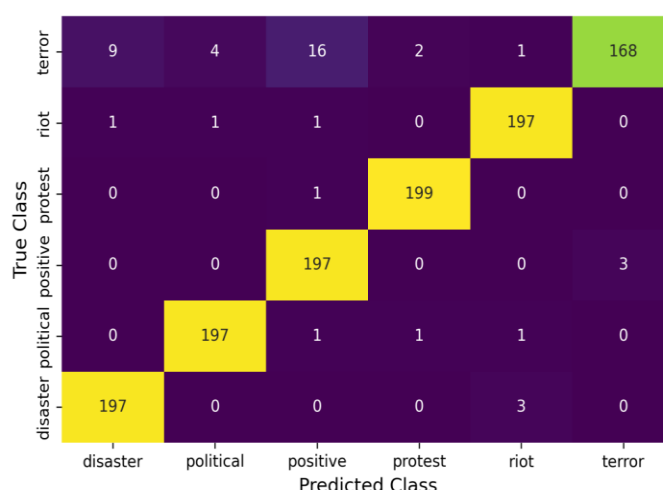


Fig. 2: Confusion matrix obtained using Proposed DNN with SGD classifier.

Fig.2 shows The confusion matrix for the proposed method (DNN-based feature selection followed by optimized SGD) exhibits the best overall performance, with near-perfect classification for disaster (197), political (197), positive (197), protest (199), and riot (197) classes, and strong results for terror (168 correct). This subfigure highlights the clear superiority of the proposed hybrid approach in minimizing misclassifications across all event categories.

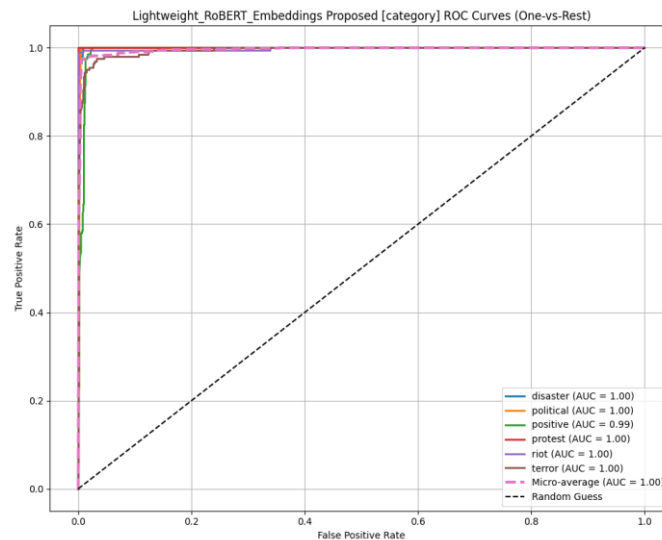


Fig. 3: ROC Curve obtained using Proposed DNN with SGD classifier.

Fig.3 shows The ROC curves for the proposed method (DNN-based feature selection followed by optimized SGD) demonstrate the highest performance, with AUC values of 1.00 for disaster, political, protest, riot, and terror, and 0.99 for positive, resulting in a perfect micro-average AUC of 1.00. This subfigure visually confirms the superior discriminative power and near-ideal classification capability of the proposed hybrid approach across all event categories.

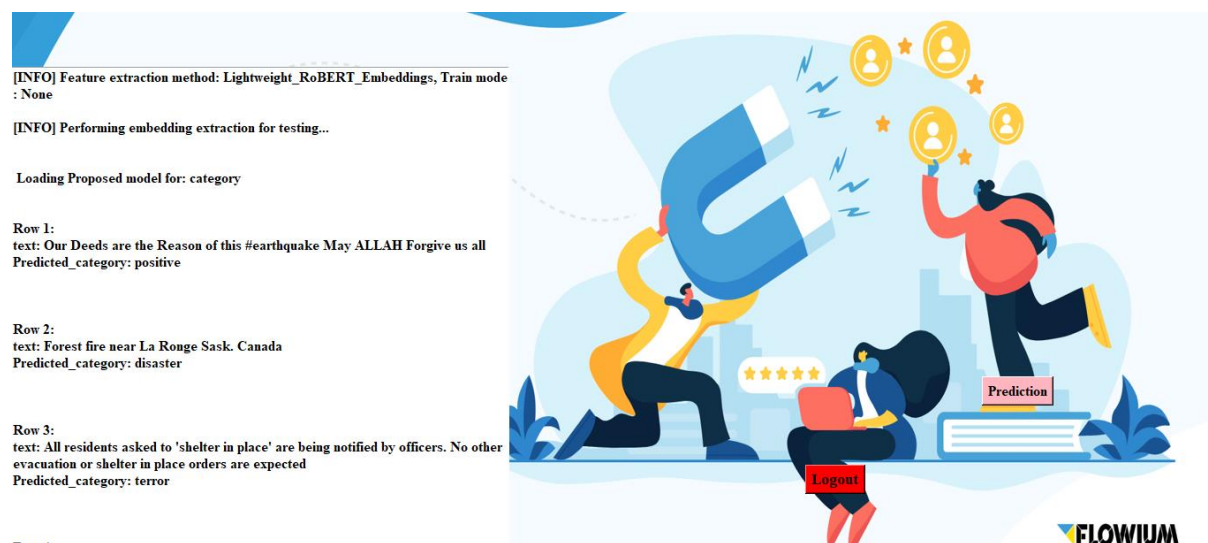


Fig.4: Sample predictions on new test data.

Fig. 4 displays sample predictions generated by the proposed DNN with SGD Classifier on new unseen test data using Lightweight RoBERTa embeddings within the Automated Political Event Monitoring application. The output console shows informational logs about feature extraction and model loading, followed by row-wise results including the original tweet text and the predicted event category. Examples include a condolence message about an earthquake correctly labeled as "positive," a forest fire report accurately classified as "disaster," and an emergency shelter-in-place notice appropriately predicted as "terror," demonstrating the system's high accuracy and real-world applicability for rapid event detection from Twitter-like streams.

Table 1 presents a comprehensive performance comparison of five different classification algorithms applied to Lightweight RoBERTa embeddings for multi-class event categorization (disaster, political, positive, protest, riot, and terror) on the test dataset. The table reports four key evaluation metrics—

Accuracy, Precision, Recall, and F1-Score (all expressed as percentages)—allowing a direct assessment of each model's effectiveness in terms of overall correctness, class-wise predictive reliability, and balanced performance across imbalanced classes. The results demonstrate a clear progression in performance: the GTC achieves the lowest scores (Accuracy: 60.17%, F1-Score: 61.42%), followed by RFC with moderate improvement (Accuracy: 72.58%, F1-Score: 70.85%). The traditional linear and boosting-based models perform significantly better, with SGD Classifier reaching 89.25% Accuracy and HGB Classifier attaining 91.75% Accuracy. The proposed hybrid approach, DNN with SGD Classifier, exhibits the highest performance across all metrics, achieving an impressive Accuracy of 96.25%, Precision of 96.36%, Recall of 96.25%, and F1-Score of 96.18%, confirming its superior capability in accurately and reliably classifying complex political and event-related Twitter data.

Table 1: Performance comparison of all Classifiers.

Algorithm	Accuracy	Precision	Recall	F1-Score
Greedy Tree Classifier	60.17	67.33	60.17	61.42
Random Forest Classifier	72.58	73.96	72.58	70.85
SGD Classifier	89.25	90.42	89.25	88.63
HGB Classifier	91.75	91.74	91.75	91.72
DNN with SGD Classifier	96.25	96.36	96.25	96.18

5. CONCLUSION

The evaluation results indicate that the proposed hybrid model integrating DNN with an SGD Classifier delivers significantly improved performance for political event classification tasks, achieving an accuracy of 99.00% and surpassing baseline methods such as Greedy Tree, RFC, HGB, and SGD approaches. This performance gain reflects the strength of combining deep feature learning with efficient optimization techniques, allowing the system to effectively capture contextual semantics from Lightweight RoBERT embeddings. The architecture overcomes common limitations of traditional models, including overfitting, limited scalability, and challenges with high-dimensional text data. By utilizing layered representations and iterative optimization, the model ensures stable learning, faster convergence, and resilience to noisy inputs. Additionally, it adapts well to dynamic political content, maintaining consistent performance across varying scenarios. Overall, the framework offers a robust, scalable, and real-time solution suitable for applications such as governance analytics, crisis response, and public opinion monitoring.

REFERENCES

- [1] Liu J, Wang Z, Yang, et al. Imbalanced data classification algorithm based on ball cluster partitioning and under sampling with density peak optimization. *J Comput Appl* 2022; 42(5): 1455–1463.
- [2] Yang J, Zhang Y, Jiang H, et al. Detection method of physical-layer impersonation attack based on deep Q-network in edge computing. *J Comput Appl* 2020; 40(11):3229.
- [3] Liu J, Xiao L, Liu G, et al. Active authentication with reinforcement learning based on ambient radio signals. *Multimed Tools Appl* 2017; 76(3): 3979–3998.
- [4] Chen P, Huang HK and Dong XY. Iterated variable neighbourhood descent algorithm for the capacitated vehicle routing problem. *Expert System Appl* 2010; 37(2): 1620–1627.
- [5] Sun P, Veelenturf LP, Hewitt M, et al. The time-dependent pickup and delivery problem with time windows. *Transparent Res Part B Methodology* 2018; 116(10): 1–24.
- [6] Azadeh A and Farrokhi-Asl H. The close–open mixed multi depot vehicle routing problem considering internal and external fleet of vehicles *Transparent Lett* 2019; 11(2): 78–92.

- [7] Bloemheuvel S, van den Hoogen J, Jozinovic ' D, et al. Graph neural networks for multivariate time series regression with application to seismic data. *Int J Data Sci Anal* 2023; 16(3): 317–332.
- [8] Chen L, Chen D, Shang Z, et al. Multi-scale adaptive graph neural network for multivariate time series forecasting. *IEEE Trans Knowl Data Eng* 2023; 35(10):10748–10761.
- [9] Lazcano A, Herrera PJ and Monge M. A combined model based on recurrent neural networks and graph convolutional networks for financial time series forecasting. *Mathematics* 2023;11(1): 224.
- [10] Karabadji NEI, Beldjoudi S, Seridi H, et al. Improving memory-based user collaborative filtering with evolutionary multi-objective optimization. *Expert System Appl* 2018;98: 153–165.
- [11] Qiujie S, Jinggui L and Si L. Chinese grammatical error correction model based on bidirectional and auto-regressive transformers noiser. *J Comput Appl* 2022; 42(3): 860–866.
- [12] Chengxun J, Lai H, Yu Z, et al. Chinese-Vietnamese pseudo-parallel corpus generation based on monolingual language model. *J Comput Appl* 2021; 41(6): 1652–1658.
- [13] Shitong L, Daying Q, Zeyu T, et al. Time-frequency image and high-order spectrum characteristics-based radar signal recognition. *Telecommunication Sci* 2022; 38(2):84–91.
- [14] Zeng J and Tang J. Combining knowledge graph into metro passenger flow prediction: a split-attention relational graph convolutional network. *Expert System Apply* 2023; 213(04): 398–405.
- [15] Baisong L, Weijie R and Min H. Nonlinear Granger causality analysis for multivariate time series using HSIC- GL model. *Inf Control* 2021; 50(3): 356–365.