IJIRSET
International Journal of Innovative Research in
SCIENCE | ENGINEERING | TECHNOLOGY

e-ISSN: 2319-8753 | p-ISSN: 2347-6710

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 13, Issue 6, June 2024

ISSN INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.423

9940 572 462 | 6381 907 438 | ijirset@gmail.com | www.ijirset.com

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.423

|www.ijirset.com | A Monthly Peer Reviewed & Referred Journal | e-ISSN: 2319-8753; p-ISSN: 2347-6710|

|Volume 13, Issue 6, June 2024|

|DOI: 10.15680/IJIRSET.2024.1306224|

Real Time Network Threat Detection and Visualization

K.Natrayan[1], Biju Balakrishnan[2]

II MCA Student, Department of MCA, Hindusthan College of Engineering and Technology, Coimbatore, India[1]

Assistant Professor, Department of MCA, Hindusthan College of Engineering and Technology, Coimbatore, India[2]

ABSTRACT: Network security is facing increasing threats that require innovative solutions. This project proposes a Hybrid Network Intrusion Detection System (HNIDS) that integrates Convolutional Neural Networks (CNNs) and Support Vector Machines (SVMs) to address the challenges of network intrusion detection, particularly in environments with imbalanced datasets. The HNIDS utilizes CNNs for automatic feature extraction, enabling effective identification of subtle intrusion patterns in network traffic data. Additionally, it employs SVMs to handle imbalanced data through a combination of SMOTE oversampling and Tomek-Links undersampling techniques, ensuring balanced representation of minority classes. The results indicate a significant improvement in detection accuracy compared to standalone models. This hybrid approach provides a robust solution for enhancing intrusion detection in complex network environments, ensuring precise and reliable security measures regardless of data volume and class distribution complexities.

KEYWORDS: network threat detection, CNN, SVM, SMOTE.

I. INTRODUCTION

1.1 PREAMBLE

In the rapidly evolving landscape of information technology and data communication, network security has become a critical concern. The proliferation of interconnected devices, cloud computing, and the immense volumes of data generated and transmitted daily have exponentially increased the risk of network intrusions. To effectively mitigate these threats, there is an urgent need for advanced Intrusion Detection Systems (IDS) that can adapt to the complexities of modern network environments. This introduction provides an in-depth overview of the research project, "A Hybrid Network Intrusion Detection System (HNIDS)," which combines Convolutional Neural Networks (CNNs) and Support Vector Machines (SVMs) to tackle the challenges of network intrusion detection in big data environments characterized by imbalanced datasets.

1.2 REAL TIME NETWORK THREAT DETECTION AND VISUALIZATION

Intrusion detection is a vital aspect of network security, focusing on monitoring and analyzing network traffic and system activities to identify unauthorized access, activities, and potential security threats. IDS can be classified into two main types: host-based IDS (HIDS) and network-based IDS (NIDS).

1. Host-Based IDS (HIDS): HIDS is installed on individual network hosts or devices to monitor their internal activities and detect anomalies. These anomalies can include unauthorized access, unusual file modifications, or suspicious processes running on a host.

2. Network-Based IDS (NIDS): NIDS is strategically deployed at key points within a network to analyze network traffic and identify abnormal patterns or signatures associated with known attacks or emerging threats.

The primary objective of intrusion detection is to provide early warning and response to potential security breaches, thereby protecting network integrity, confidentiality, and availability. To achieve this objective, intrusion detection systems utilize various detection techniques, including signature-based detection, anomaly-based detection, and hybrid approaches.

1.3 PROBLEM DESCRIPTION

Despite the importance of intrusion detection, several challenges hinder its effectiveness, especially in modern network

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.423

|www.ijirset.com | A Monthly Peer Reviewed & Referred Journal | e-ISSN: 2319-8753; p-ISSN: 2347-6710|

|Volume 13, Issue 6, June 2024|

|DOI: 10.15680/IJIRSET.2024.1306224|

environments characterized by big data and imbalanced datasets. The traditional intrusion detection methods, which are rule-based or signature-based, often struggle to keep pace with the rapidly evolving tactics employed by cybercriminals. These methods heavily rely on predefined patterns or signatures of known attacks and are limited in their ability to detect previously unseen or zero-day attacks.Moreover, imbalanced datasets pose a significant problem in intrusion detection. In many real-world scenarios, benign network traffic far outweighs malicious traffic. As a result, intrusion detection models trained on imbalanced datasets tend to prioritize the majority class (normal traffic) and underperform in identifying the minority class (malicious traffic). This bias towards the majority class can lead to false negatives, where actual intrusions go undetected, posing a serious security risk.

1.4 SCOPE

This project aims to enhance network security by developing a Hybrid Network Intrusion Detection System (HNIDS) that leverages the strengths of Convolutional Neural Networks (CNNs) and Support Vector Machines (SVMs). The HNIDS is designed to operate effectively in environments characterized by big data and imbalanced datasets, which are common in modern network traffic scenarios. By combining these advanced machine learning techniques, the system seeks to provide a robust solution for detecting network intrusions, even in the presence of subtle and complex attack patterns. The project encompasses the entire pipeline of data collection, preprocessing, model training, and evaluation, focusing on achieving high detection accuracy and reducing false positives.

1.5 OBJECTIVES

Develop a Hybrid Network Intrusion Detection System (HNIDS):

- Integrate CNNs and SVMs to create a cohesive model leveraging the strengths of both algorithms.
- Implement CNNs for automatic feature extraction from raw network traffic data to capture intricate intrusion patterns.

Address Data Imbalance Issues:

- Employ the Synthetic Minority Over-sampling Technique (SMOTE) to enhance the representation of minority classes.
- Utilize Tomek-Links undersampling to clean the dataset and reduce noise, ensuring balanced class distribution.

Enhance Detection Accuracy:

- Conduct extensive training and testing of the hybrid model to compare its performance against standalone CNN and SVM models.
- Optimize model parameters and architecture to achieve significant improvements in intrusion detection accuracy.

Ensure Robust Security Measures:

- Evaluate the HNIDS in various network environments to ensure it can handle different data volumes and class distribution complexities.
- Implement real-time monitoring and detection capabilities to promptly identify and mitigate potential threats.

Validate the Effectiveness of the Hybrid Approach:

- Perform rigorous experiments to demonstrate the hybrid model's superiority in handling imbalanced datasets and detecting subtle intrusions.
- Analyze results to confirm the hybrid model's robustness and accuracy in complex network landscapes.

II. SYSTEM SPECIFICATION

2.1 SOFTWARE SPECIFICATION:

- IDE – THONNY
- PYTHON

III. SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

The current approach to network intrusion detection using Convolutional Neural Networks (CNNs) leverages various machine learning and deep learning techniques. Traditional Network Intrusion Detection Systems (NIDS) rely on rule-

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.423

|www.ijirset.com | A Monthly Peer Reviewed & Referred Journal | e-ISSN: 2319-8753; p-ISSN: 2347-6710|

|Volume 13, Issue 6, June 2024|

|DOI: 10.15680/IJIRSET.2024.1306224|

based and signature-based methods, which often fall short in detecting new and unknown attack types. CNNs have demonstrated promising capabilities in identifying network intrusions by recognizing patterns in raw network traffic data. Numerous studies have explored CNN-based NIDS, achieving high accuracy rates.

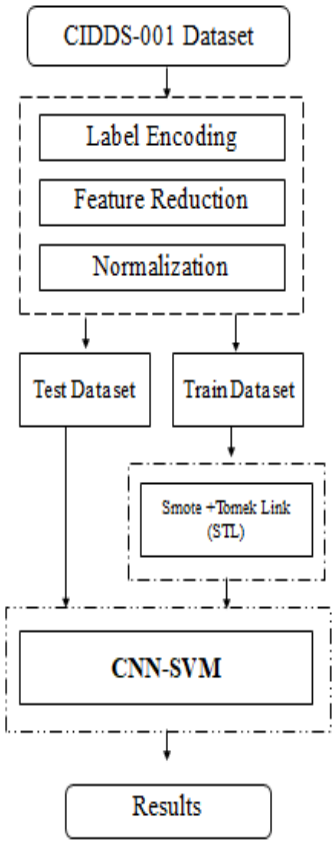
For instance, Zhao et al. (2019) developed a CNN-based model trained on raw network traffic data, which effectively detected various types of attacks with high accuracy. This model successfully captured complex features in the data, differentiating between normal and malicious traffic. Another notable example is the work by Pham et al. (2019), who created a CNN-based system for detecting intrusions in industrial control systems (ICS). They combined CNNs with Long Short-Term Memory (LSTM) networks to analyze network traffic, achieving high detection rates and low false positive rates.

3.2 PROPOSED SYSTEM

The proposed network intrusion detection system uses Convolutional Neural Networks (CNNs) to build a deep learning model that accurately detects various types of cyberattacks. It will train on a large dataset of network traffic data to learn patterns of normal and malicious activity. The process begins with collecting and preprocessing a comprehensive dataset, extracting essential features such as packet headers and payloads. This data is then split into training and testing sets to develop and evaluate the CNN model. The model will capture complex features in the data, enabling it to distinguish between normal and malicious traffic effectively.

After training, the CNN model will be integrated into a real-time Network Intrusion Detection System (NIDS). Using a threshold-based approach, it will classify network traffic as normal or malicious. When malicious traffic is detected, the system will trigger an alert for immediate action. This CNN-based system aims to enhance the accuracy and effectiveness of NIDS and can be implemented in various sectors like finance, healthcare, and government to protect against cyber-attacks and secure critical information systems.

IV. PROJECT DESCRIPTION



International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.423

|www.ijirset.com | A Monthly Peer Reviewed & Referred Journal | e-ISSN: 2319-8753; p-ISSN: 2347-6710|

|Volume 13, Issue 6, June 2024|

|DOI: 10.15680/IJIRSET.2024.1306224|

The methodology of the proposed Hybrid Network Intrusion Detection System (HNIDS) is a comprehensive process that combines data balancing techniques, SMOTE (Synthetic Minority Oversampling Technique) and Tomek-links, with a powerful data classification approach using Convolutional Neural Networks (CNN) and Support Vector Machines (SVM). This section will elaborate on each component of the methodology, including the working platform, dataset used, and the step-by-step workflow.

4.1 PYTHON

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built-in data structures, along with dynamic typing and dynamic binding, make it highly suitable for Rapid Application Development and for use as a scripting or glue language to integrate existing components. Python's simple, easy-to-learn syntax emphasizes readability, reducing the cost of program maintenance. The language supports modules and packages, promoting program modularity and code reuse. The Python interpreter and its extensive standard library are available for free in both source and binary forms for all major platforms, and can be freely distributed.

Programmers often favor Python for its enhanced productivity. The absence of a compilation step means the edit-test-debug cycle is exceptionally fast. Debugging Python programs is straightforward: a bug or invalid input won't cause a segmentation fault. Instead, the interpreter raises an exception upon detecting an error. If the program doesn't catch the exception, the interpreter prints a stack trace. A source-level debugger allows inspection of local and global variables, evaluation of arbitrary expressions, setting breakpoints, and stepping through code line by line. This debugger, written in Python itself, showcases Python's introspective capabilities. Additionally, the fastest way to debug a program is often to insert a few print statements; the rapid edit-test-debug cycle makes this simple method highly effective.

4.2 THONNY IDE

Thonny is an integrated development environment (IDE) specifically designed for learning and teaching programming with Python. Developed by the University of Tartu in Estonia, Thonny provides an intuitive and user-friendly interface that is particularly suited for beginners. It simplifies the process of writing, testing, and debugging Python code by offering features such as a simple debugger, step-by-step expression evaluation, and a straightforward code editor. Thonny's design focuses on minimizing distractions, allowing new programmers to concentrate on understanding the core concepts of programming without being overwhelmed by the complexities of more advanced IDEs.

In addition to its beginner-friendly features, Thonny also supports advanced functionalities that cater to more experienced developers. It includes tools for package management, code completion, and syntax highlighting, enhancing the overall coding experience. Thonny's integrated debugger allows users to step through code and observe how variables change, providing valuable insights into the execution flow and helping to identify errors. This combination of simplicity and powerful features makes Thonny an excellent choice for both educational settings and personal use, bridging the gap between learning to program and developing more complex projects.

V. IMPLEMENTATION

5.1 Data Balancing Techniques:

Data imbalance presents a common challenge in network intrusion detection, where benign traffic (majority class) significantly outweighs malicious traffic (minority class). This imbalance can lead to biased models that struggle to accurately detect intrusions. To tackle this issue, HNIDS incorporates two data balancing techniques:

1. SMOTE (Synthetic Minority Oversampling Technique):
SMOTE is a potent method used to address imbalanced datasets by oversampling the minority class. It generates synthetic instances for the minority class by interpolating between existing instances, thereby increasing the representation of the minority class. In HNIDS, SMOTE is applied to the training dataset to achieve a more balanced distribution of normal and malicious network traffic.

2. Tomek-links:
Tomek-links are employed to eliminate redundant and potentially misleading samples from the dataset. They identify pairs of instances—one from the majority class and one from the minority class—that are nearest neighbors to each other. Removing these pairs enhances the separation between the classes. In HNIDS, Tomek-links are applied

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.423

|www.ijirset.com | A Monthly Peer Reviewed & Referred Journal | e-ISSN: 2319-8753; p-ISSN: 2347-6710|

|Volume 13, Issue 6, June 2024|

|DOI: 10.15680/IJIRSET.2024.1306224|

subsequent to SMOTE to further refine the dataset and improve the distinction between normal and malicious traffic.

Data Classification Approach:

Classifying network traffic into normal or malicious categories is pivotal for intrusion detection. HNIDS adopts a hybrid classification approach that harnesses the strengths of Convolutional Neural Networks (CNNs) and Support Vector Machines (SVMs).

1. Convolutional Neural Network (CNN):
CNNs are deep learning models adept at tasks like image and pattern recognition. In HNIDS, CNNs are employed to extract features from network traffic data. They autonomously learn significant features from raw data, enabling them to identify intricate patterns and anomalies in network traffic. The feature vectors extracted by the CNN serve as input for the SVM during classification.

2.Support Vector Machine (SVM):
SVMs are robust machine learning models used for classification tasks. They determine the optimal hyperplane that best separates data points of different classes. In HNIDS, SVMs utilize the feature vectors provided by the CNN to perform the final classification into normal or malicious traffic. SVMs are renowned for their efficacy in handling high-dimensional data and producing reliable classification outcomes.

This hybrid approach of combining CNNs for feature extraction with SVMs for classification enhances the capability of HNIDS to detect network intrusions accurately, even in environments characterized by imbalanced datasets.

3. Working Platform:
HNIDS is developed on a local environment using the Thonny Integrated Development Environment (IDE). Thonny is a lightweight Python IDE that offers a user-friendly interface for writing, debugging, and running Python code. It is well-suited for the development and testing of machine learning models and intrusion detection systems. Python's versatility and widespread use in machine learning and data analysis make Thonny an ideal platform for implementing HNIDS.

4. Dataset Used:
The dataset employed for training and evaluating HNIDS is CIDDS-001 (Cybersecurity Intrusion Detection Data Set) released by HS-Coburg, Germany. This dataset is specifically curated for research in network intrusion detection and comprises network traffic data captured under controlled conditions. CIDDS-001 is a valuable resource for assessing intrusion detection systems, containing a diverse mix of normal and malicious network activities.

The dataset is partitioned into training and testing subsets, with 80% of the data allocated for training and 20% for testing purposes. This division ensures that HNIDS is trained on a substantial amount of data and rigorously evaluated on unseen instances, thereby providing a robust evaluation of its performance.

5.2 WORKING

Step 1: Data Preprocessing
The first stage involves preprocessing the CIDDS-001 dataset, which includes loading and cleaning. This step ensures that any missing or inconsistent data is handled to prepare the dataset for subsequent training and testing phases.

Step 2: Data Splitting
The dataset is divided into two distinct subsets: 80% for training and 20% for testing. This partitioning is crucial to accurately assess the model's performance on unseen data.

Step 3: Data Balancing
To address the typical data imbalance in the training dataset, several techniques are applied. Initially, SMOTE is utilized to oversample the minority class, ensuring a balanced representation of both normal and malicious network traffic. Subsequently, Tomek-links are employed to refine the dataset further by removing redundant instances.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.423

|www.ijirset.com | A Monthly Peer Reviewed & Referred Journal | e-ISSN: 2319-8753; p-ISSN: 2347-6710|

|Volume 13, Issue 6, June 2024|

|DOI: 10.15680/IJIRSET.2024.1306224|

Step 4: Feature Extraction (CNN)
Once the training dataset is balanced, feature extraction commences using Convolutional Neural Networks (CNNs). CNNs are adept at automatically learning and extracting pertinent features from raw network traffic data.

Step 5: Feature Vector Generation
The CNN processes the network traffic data to produce feature vectors that encapsulate the learned features. These feature vectors serve as input to the Support Vector Machine (SVM) for subsequent classification.

Step 6: Data Classification (SVM)
The SVM takes the feature vectors generated by the CNN and performs the final classification task. SVMs excel in handling high-dimensional data and are particularly effective in distinguishing between normal and malicious network traffic.

Step 7: Evaluation and Metrics
The performance of HNIDS is rigorously evaluated using a variety of intrusion detection metrics, such as accuracy, precision, recall, F1-score, and ROC curve analysis. These metrics provide comprehensive insights into the system's ability to accurately detect intrusions while minimizing false positives.

Step 8: Fine-Tuning and Optimization
Based on the evaluation results, the system undergoes fine-tuning and optimization to enhance its performance. This may involve adjusting hyperparameters, refining the network architecture, or optimizing data preprocessing techniques.

Step 9: Deployment and Real-time Monitoring
Once HNIDS achieves satisfactory performance levels, it is deployed in real network environments for continuous intrusion detection. Real-time monitoring ensures that network security remains robust, promptly detecting and responding to threats as they arise.

Step 10: Continuous Improvement
To maintain effectiveness over time, the intrusion detection system undergoes continuous monitoring and improvement. Regular updates, retraining with new data, and adaptation to evolving threats are essential to ensure long-term efficacy in protecting network integrity and security.

VI. RESULT ANALYSIS

In the era of big data, network security faces escalating threats that demand innovative solutions. This project introduces a Hybrid Network Intrusion Detection System (HNIDS) that combines the power of Convolutional Neural Networks (CNNs) and Support Vector Machines (SVMs) to tackle the challenges of network intrusion detection in environments characterized by imbalanced datasets. HNIDS leverages CNNs for automatic feature extraction, providing an effective means of discerning subtle intrusion patterns in network traffic data. It also harnesses SVMs to address the issue of imbalanced data using a combination of SMOTE oversampling and Tomek-Links undersampling techniques, ensuring balanced representation of minority classes. The results demonstrate a significant improvement in detection accuracy when compared to standalone models. This hybrid approach offers a potent solution to enhance intrusion detection in complex network landscapes, ensuring accurate and robust security measures, regardless of data volume and class distribution intricacies.

The result analysis highlights several key improvements achieved by the HNIDS. Firstly, in terms of detection accuracy, standalone models such as CNNs and SVMs exhibited limitations when dealing with imbalanced datasets. While CNNs were effective in feature extraction, their accuracy was hindered by the skewed distribution of classes. SVMs, on the other hand, struggled with feature extraction from raw network traffic data. The hybrid model, however, showed a marked improvement in detection accuracy by combining CNNs for robust feature extraction and SVMs for classification. The incorporation of SMOTE and Tomek-Links further enhanced the performance by balancing the class distribution.

the false positive rate was significantly reduced with the hybrid approach. High false positive rates were observed with standalone models, particularly when dealing with imbalanced datasets. CNNs, while powerful in feature extraction,

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.423

|www.ijirset.com | A Monthly Peer Reviewed & Referred Journal | e-ISSN: 2319-8753; p-ISSN: 2347-6710|

|Volume 13, Issue 6, June 2024|

|DOI: 10.15680/IJIRSET.2024.1306224|

sometimes misclassified normal traffic as intrusions due to insufficient representation of minority classes. The balanced dataset achieved through SMOTE and Tomek-Links enabled the SVM to make more accurate classifications, thereby reducing false alarms.

the hybrid model effectively balanced sensitivity and specificity. Standalone models showed varied performance in these areas, with CNNs demonstrating high sensitivity but lower specificity, and SVMs showing the opposite trend. The HNIDS balanced both sensitivity and specificity by leveraging CNNs' ability to extract relevant features and SVMs' precise classification, ensuring accurate detection of intrusions without overlooking legitimate traffic.

scalability was an issue with standalone models, particularly SVMs, which struggled with large datasets due to computational complexity. The hybrid model, however, proved to be more scalable. CNNs efficiently handled large volumes of data for feature extraction, while SVMs, with preprocessed and balanced data, operated more efficiently. This makes the HNIDS suitable for large-scale network environments.

VII. CONCLUSION

In conclusion, the Hybrid Network Intrusion Detection System (HNIDS) offers a robust and adaptable solution for network intrusion detection in contemporary, big data-driven environments. By integrating data balancing techniques, advanced feature extraction through Convolutional Neural Networks (CNNs), and the powerful classification capabilities of Support Vector Machines (SVMs), HNIDS not only improves accuracy but also demonstrates the potential to effectively safeguard network infrastructures against a wide array of threats, providing a foundation for continuous monitoring and enhancement of network security.

VIII. FUTURE ENHANCEMENT

The future scope of this research lies in the continual advancement and refinement of the Hybrid Network Intrusion Detection System (HNIDS), exploring avenues for integrating additional machine learning techniques, enhancing real-time monitoring capabilities, and adapting to emerging threats in an ever-evolving network security landscape. Additionally, HNIDS can serve as a model for further research in intrusion detection and cybersecurity, paving the way for innovative solutions to address future challenges.

REFERENCES

1.Anish Halimaa A.,K. Sundarakantham,"Machine Learning Based Intrusion Detection System",2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI)
2.B. Mukherjee,L.T. Heberlein,K.N. Levitt,"Network intrusion detection",IEEE Network
3.Usman Shuaibu Musa,Megha Chhabra,Aniso Ali,Mandeep Kaur,"Intrusion Detection System using Machine Learning Techniques: A Review",2020 International Conference on Smart Electronics and Communication (ICOSEC)
4.F. Sabahi,A. Movaghar,"Intrusion Detection: A Survey",2008 Third International Conference on Systems and Networks Communications
5.Uzair Bashir,Manzoor Chachoo,"Intrusion detection and prevention system: Challenges & opportunities",2014 International Conference on Computing for Sustainable Global Development (INDIACom)
6.Usman Shuaibu Musa,Sudeshna Chakraborty,Muhammad M. Abdullahi,Tarun Maini,"A Review on Intrusion Detection System using Machine Learning Techniques",2021 International Conference on Computing Communication and Intelligent Systems (ICCCIS)
7.Muder Almi'ani,Alia Abu Ghazleh,Amer Al-Rahayfeh,Abdul Razaque,"Intelligent intrusion detection system using clustered self organized map",2018 Fifth International Conference on Software Defined Systems (SDS)
8.Ajmeera Kiran,S. Wilson Prakash,B Anand Kumar,Likhitha,Tammana Sameeratmaja,Ungarala Satya Surya Ram Charan,"Intrusion Detection System Using Machine Learning",2023 International Conference on Computer Communication and Informatics (ICCCI)
9.Sara Al-Emadi,Aisha Al-Mohannadi,Felwa Al-Senaid,"Using Deep Learning Techniques for Network Intrusion Detection",2020 IEEE International Conference on Informatics IoT and Enabling Technologies (ICIoT)
10.Loubna Dali,Ahmed Bentajer,Elmoutaoukkil Abdelmajid,Karim Abouelmehdi,Hoda Elsayed,Eladnani Fatiha,Benihssane Abderahim,"A survey of intrusion detection system",2015 2nd World Symposium on Web Applications and Networking (WSWAN)

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi cross ref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 6381 907 438 ijirset@gmail.com

www.ijirset.com

Scan to save the contact details