Advanced Real-Time Anomaly Detection and Predictive Trend Modelling in Smart Systems using Deep Belief Networks Architectures

Amro ameid alkato [1] *and Yara sakhnini* [2]

[1] *King Abdullah II School of Information Technology, University of Jordan, Amman Jordan*
[2] *Faculty of Computer & Information Technology, Jordan university of science and technology, Irbid Jordan*

ABSTRACT

Intelligent systems are becoming more data-intensive and complicated, so it's essential to have reliable methods to recognize anomalies immediately and monitor trends to keep the system running well. The present research delves into using the DeepSense Framework, built up on top of Deep Belief Networks (DBNs), to create a system that can change with changing settings and recognize and anticipate anomalies accurately. DBNs are generally used to examine data collections with several dimensions because of their multilayered, unsupervised prior instruction and fine-tuning capacities. The framework is built to capture complicated, unpredictable relationships amongst the collected information to discover minor abnormalities and anticipate patterns in the future. The DeepSense Framework successfully reduces the amount of false positives while keeping the identification of anomalies responsiveness substantial, according to an experimental assessment. The structure also outperforms more traditional quantitative and neural network methods when it comes to trend predictions.

Keywords: Anomaly detection, predictive modelling, DeepSense Framework, Deep Belief Networks, intelligent systems.

1. Introduction

Artificial intelligence significantly influences transformation across various sectors, particularly in medical treatment. Deep neural networks (DNNs) have considerably improved medical image analysis, enhancing detection accuracy and efficiency[1]. Systems operated by artificial intelligence evaluate intricate medical data to assist physicians in diagnosing critical conditions such as brain lesions, bone cancer, breast cancer, and broken bones. These developments illustrate the growing popularity of data-based methods to enhance healthcare procedures. Regardless of these improvements, automated systems for instantaneous fashion anomaly recognition and predictions in rapidly changing and filled with information surroundings encounter obstacles related to robustness, adaptability, and preciseness[2].

ISSN: 3006-8894
https://doi.org/10.70023/sahd/250209

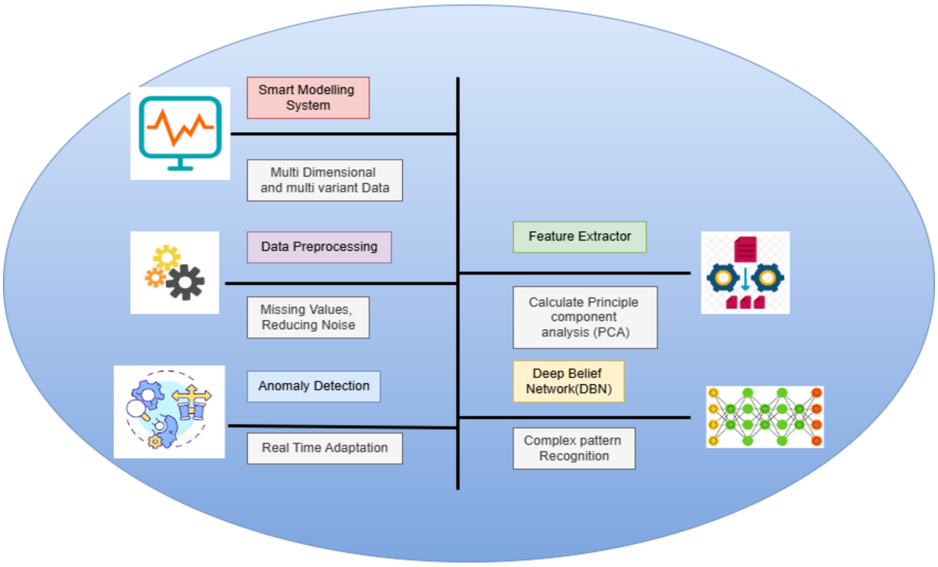
Smart Modelling System
Multi Dimensional and multi variant Data
Data Preprocessing
Missing Values, Reducing Noise
Anomaly Detection
Real Time Adaptation
Feature Extractor
Calculate Principle component analysis (PCA)
Deep Belief Network(DBN)
Complex pattern Recognition

Figure. 1. The Framework Components Of DeepSense

The present article examines the development of a dependable and versatile mixture for immediate time detection of anomalies and forecasting trends within automated systems. Current methodologies face challenges in achieving an equilibrium among sensibility, erroneous costs, and expansion throughout different applications[3]. Multidimensional and multiple-variate stream structures may encounter the boundaries above. This article presents the DeepSense Framework for Windows, which utilizes Deep Belief Networks to tackle these obstacles.

This method mitigates such obstacles by offering an adaptable framework to identify discrepancies and forecast movements. This research employs the DeepSense Framework to optimize the advantages of Deep Belief Networks (DBN)[4]. Deep Belief Networks (DBNs) are well-suited for analyzing multifaceted datasets due to their intricate structure, autonomous prior instruction, and subsequent adjustment processes. The technique identifies complex, unpredictable data connections to uncover delicate anomalies and predict future patterns. Comparative experiments of the DeepSense Framework against statistical and artificial neural network procedures emphasized adaptation in detecting anomalies, false favourable rates, and predictability[5].

A) The Main Contributions are:

1) Decrease in Incorrect Positive Results For more accurate anomaly identification, DeepSense Framework shows a substantial decrease in the rate of false positives as contrasted with conventional methods[6].
2) Improved Analytical Power By capitalizing on DBNs' intrinsic ability, this structure achieves better results than traditional methods in anticipating trends duties, offering a versatile and expandable option for many scenarios[7].
3) The architecture is an excellent instrument for applications running in real-time since it can modify different conditions despite being very sensitive and accurate.

This paper's architecture is as follows. Section 1 extensively reviews applicable research, stressing anomaly detection and modeling-related issues. Section 2 describes the DeepSense Framework's design and features. Empirical information sets, assessment measurements, and comparing methods are described in Section 3. The conclusions and discourse in Section 4 highlight the system's effectiveness and improvements versus conventional approaches. Section 5 summarizes the main findings, discusses the investigation's consequences, and suggests additional studies.

2. Literature Survey

a) Activity Recognition

Elsayed et al.[8]PredictDeep is a deep learning-powered security intelligence platform for Wireless significant data anomaly identification and forecasting. It analyzes architectural and environmental linkages using log information, network designs, and sophisticated neural networks with graphs to find both conventional and abnormal behaviours. PredictDeep surpassed existing methods using a freely available Apache log collection in preciseness, resilience, and computing economy. The integrity and reliability of log information and graph-based method adaptability in significantly changing or massive systems can impact their efficiency.

Nizam et al.[9] hybrid deep anomaly detection (DAD) design uses a CNN including a two-phase LSTM-based autoencoder (AE) to identify abnormalities and rare occurrences in speed-stamped IIoT data collected from sensors in actual or near-to-immediate time. Considering immediate and distant distinctions between actual and anticipated measurements, the model surpasses current techniques. Following its effectiveness, the architecture performed well on restricted resources edge gadgets in real-time in a single experiment and two data sets from the real world. Handling severe variability in information and improving for varied business uses are problems.

Ahmad et al.[10] The recommended IoT neural-based intrusion detection system (NIDS) uses mutually beneficial data (MI) to choose characteristics and a deep neural network (DNN) to identify anomalies. The IoT-Botnet 2020 database gave the algorithm a 0.57–2.6% inaccuracy improvement and a 0.23–7.98% FAR decrease over previous models trained with deep learning. MI selected the top 16–35 computational characteristics to retain throughput and simplify the computational framework. Combining the most significant five categories and quantitative characteristics increased detection accuracy by 0.99–3.45%. Nevertheless, changing information and forms of attack may affect the system's overall efficiency, requiring additional verification across various IoT contexts.

Wong et al.[11] The suggested abnormality detection technique integrates convolutional neural network (CNN) and long short-term memory (LSTM) algorithms to instantly identify abnormalities in traffic through networks inside large-scale data centres. Classifiers trained using particle data captured were evaluated on standard malware detection databases and a vast amount of data representing actual network activity. Deep learning methods operate better at recognizing anomalies than their basic learning equivalents, as shown by the data. The algorithms' capacity to process massive volumes of data in streams with little latency makes them ideal for use in applications that operate in real-time. Optimization methods, such as compression of models and transfer learning, were proposed to increase detection effectiveness. It may be challenging to adjust to novel ways to attack and shift communications sequences, while learning the structure may require a lot of computational power.

b) Anomaly Detection

El-Shafeiy et al.[12]In particular, the suggested MCN-LSTM approach combines many convolutional network techniques and LSTM to identify outliers in complicated time-series information for immediate form surveillance of water quality. The mathematical framework has been taught and verified using statistics from actual quality water surveillance circumstances utilizing practical problem sensors. The experimental findings demonstrated that the prediction system accurately distinguished between typical and unusual data scenarios, with an accuracy percentage of 92.3%. However, more studies are required to determine how well the procedure works in different

ISSN: 3006-8894
https://doi.org/10.70023/sahd/250209

physiologic contexts and its performance, which may depend on how accurate and trustworthy the data obtained from the monitoring devices are.

Abdallah et al.[13] provided Machine learning (ML) and deep learning (DL) techniques are used to identify computer network abnormalities like distributed denial of service (DDoS) assaults and breaches. CNN and LSTM networks, in particular, have proven successful in identifying complicated anomaly sequences. Model training and evaluation often use CICDDoS2019 and UNSW-NB15 information. DL algorithms, notably CNNs and LSTMs, outperform classical approaches in Precision for detection, usually reaching 95%. Fortunately, novel approaches to attack that are absent from training information may reduce efficiency and require significant computational energy. In conclusion, ML and DL algorithms can improve virtual network protection by detecting anomalies, but limited resources and responsiveness to new vulnerabilities must be examined.

Arjunan et al.[14] The research investigation proposes an integrated abnormality detection system that uses CNN and LSTM algorithms to recognize network activity abnormalities in massive data situations in continuous time. NSL-KDD and CICDDoS2019 provide annotated standard and abnormal network activity to train the model. Experiments show that the combined CNN-LSTM strategy outperforms classic artificial intelligence methods in detecting Precision and missed opportunity frequencies. Fortunately, the model's effectiveness might be contingent on data from training integrity and generalization to new approaches to attack. In conclusion, CNN-LSTM architecture can improve actual time network activity detection of anomalies, but data integrity and expansion of models are still important.

3. Proposed Methodology

a) The DeepSense Operational Overview

The design is a Deep Belief Network (DBN) for intelligent anomaly identification and forecasting. The input stage will be followed by an array of Restricted Boltzmann Machines (RBMs) for uncontrolled extracting features. RBMs build hierarchy depictions of information provided to capture intricate trends throughout layers. RBMs are adjusted with labelled data after training to enhance the accuracy of predictions. Predictions or finding anomalies come from the very last output level. Its design allows immediate form development and flexibility in changing settings, improving the identification of anomalies and pattern forecasting.

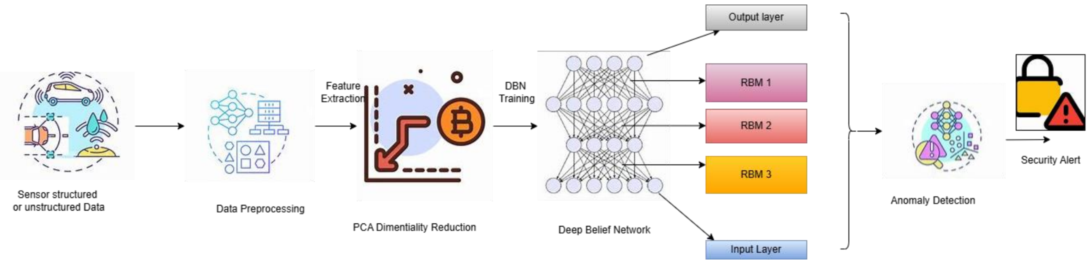

Figure. 2. The DeepSense Framework Operational Overview

The Information Collection and Preparation modules guarantee that the data inputted into the Deep Belief Networks (DBNs) is accurate and high-quality. The first step of this component is to gather unprocessed information from the sophisticated method's built-in sensors. Because these measuring devices are so varied, the information collected includes a lot of distinct parameters, some of which may have various methods of measurement and dimensions. As an example, the given information contains data values with many different characteristics in every table, includingSeveral preliminary processing operations are performed on the information to make it ready for efficient evaluation:

Dealing with Missing Values: Because of transmission mistakes or instrument problems, real-world data frequently includes missed information. To tackle this, methods including restoration are used, which involves substituting estimates from statistics (which can consist of the mean or median) for missing values.

Reducing Noise: Information from sensors may contain noise due to external influences or technical constraints. By utilizing smooth algorithms like averages that move or Stochastic filtration systems, the resulting noise can be reduced, and the data can more faithfully represent the real-world actions of the overall system.

Standardization: Since various characteristics utilize distinct dimensions, it is essential to normalize them so that they all have a single scale between 0 and 1. That way, the simulation won't be skewed by attributes with more significant dimensions. Most often, people will use the Min-Max normalized method, which is:

$$X_{normalized} = \frac{X - X_{min}}{X_{max} - X_{min}} \qquad (1)$$

where $X$is the original value,X_{min} Is the minimum value in the dataset, and X_{max}It is the maximum value.

The component guarantees healthy, constant and appropriately scalable information by carefully carrying out these processing procedures. Intelligent systems efficiently identify anomalies and forecast trend modelling when DBNs are adequately trained to acquire information and analyze the fundamental trends.

b) *Principal Component Analysis*

The main objective is to find and extract significant characteristics derived from data that has been processed to help identify anomalies and forecast trends. It is crucial to minimize the total quantity of parameters and the most relevant ones to improve computational effectiveness and the effectiveness of models, especially considering the high dimensionality of data in intelligent systems. To that end, many turn to Principal Component Analysis (PCA).In principal component analysis (PCA), the initial set of correlated variables is transformed into a new set of uncorrelated variables called the primary components.

ISSN: 3006-8894
https://doi.org/10.70023/sahd/250209

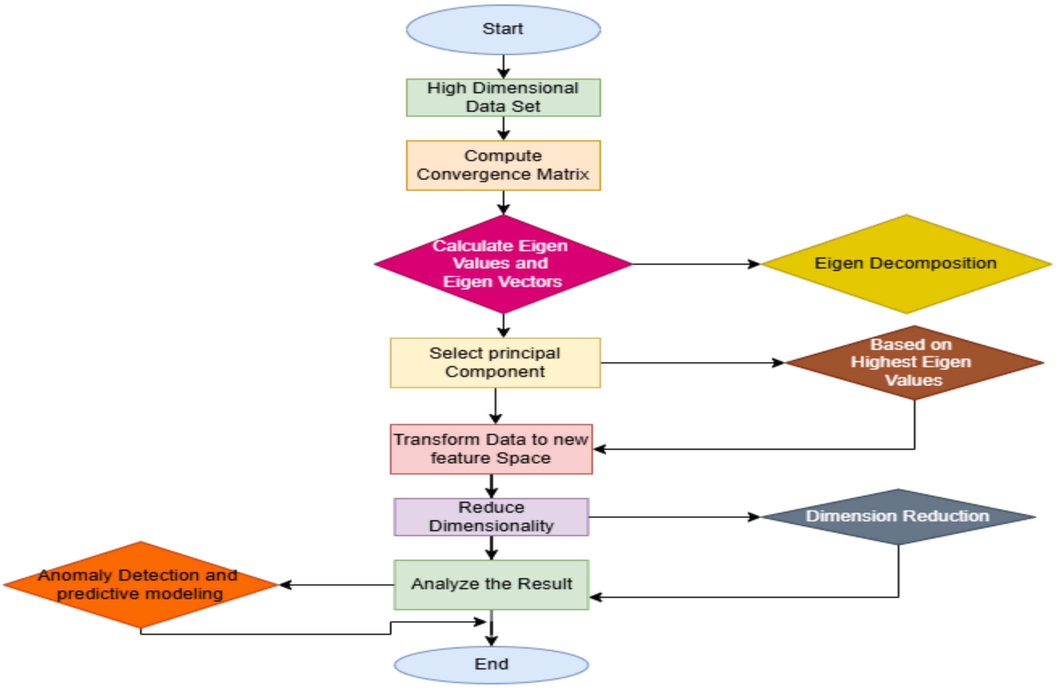
Start
High Dimensional Data Set
Compute Convergence Matrix
Calculate Eigen Values and Eigen Vectors
Eigen Decomposition
Select principal Component
Based on Highest Eigen Values
Transform Data to new feature Space
Reduce Dimensionality
Dimension Reduction
Anomaly Detection and predictive modeling
Analyze the Result
End

Figure. 3. Flowchart representation of the PCA process

These elements are then ranked according to the percentage of variability they incorporate from the information being analyzed. As part of this modification, calculate the information's coefficient matrix to see the interrelationships of the factors. This coefficient matrix then has its Eigenvalues as well as its Eigenvectors determined. The patterns of the most significant variability in information are represented by the coefficients (principal components) that coincide with the highest eigenvalue. The following equation must be solved scientifically to understand the steps involved:

$$\boldsymbol{C}v = \lambda v \qquad (2)$$

where $\boldsymbol{C}$ is the covariance matrix, v is the eigenvector, and λ is the eigenvalue. By projecting the original data onto these principal components, PCA reduces the dimensionality while preserving as much variability as possible, facilitating more efficient and effective anomaly detection and predictive modelling in smart systems.

c) Model Training with Deep Belief Networks (DBNs)

Deep Belief Networks (DBNs) represent the typical actions associated with intelligent technologies to identify anomalies and anticipate trends efficiently. DBNs are dynamic graphic representations that usually consist of numerous layered Restricted Boltzmann Machines (RBMs) containing unpredictable latent parameters. The reconstructive mistakes are typically measured as the squared variance among the source vector (x) and the regenerated vector ($\hat{x}$), each RBM learns to re-create its source vector to retain the actual information distributions.

$$\text{Reconstruction Error} = ||x - \hat{x}||^2 \qquad (3)$$

A greedy, tier-wise methodology is used to develop DBNs, whereby every RBM is taught separately to simulate the information distributions at its appropriate tier. The artificial neural network is fine-tuned for specific duties like recognizing anomalies after training every layer using supervised learning techniques. Within this framework, the DBN can represent the typical functioning characteristics of intelligent systems. An anomaly may exist if significant discrepancies exist across the information provided and the DBN's synthesis of newly added information. This capacity allows the device to identify odd behaviours instantaneously, improving the effectiveness and dependability of automated systems by allowing for quick reactions to abnormalities and precise forecasts regarding potential patterns.

Table. 1. The following table illustrates sample data points used in the DBN training process

ID	**Biomarker1**	**Biomarker1 2**	**Signal**	***Index***	**Target**	**Label**
109797	2.089	0.325	-1.610	0.577	1.99	0
109800	0.113	1.004	-0.273	-0.314	1.98	0
109801	-0.667	0.838	2.351	-0.120	11.27	0
109802	-1.127	1.277	1.970	2.305	8.31	0
109803	-0.857	1.610	-0.574	-0.450	8.91	0
109804	-0.467	0.409	0.214	-0.203	162.00	0

This data comprises multiple features (sensor readings) and corresponding target values, which the DBN utilizes to learn the standard behaviour patterns of the system. By effectively modelling these patterns, the DBN can detect deviations indicative of anomalies and predict future trends, thereby contributing to the advancement of real-time anomaly detection and predictive trend modelling in intelligent systems.

d) Anomaly Predictor

Anomaly identification and prediction pattern modelling are two essential tasks used for the trained Deep Belief Network (DBN) in the final section of the above structure. The specific steps for performing each task are as follows:

e) Anomaly Detection

The DBN may recognize discrepancies by evaluating the provided information at the current time and the connections and patterns learnt throughout instruction. The procedure entails keeping an eye out for differences between the predicted results of the model and the current information. These outliers are marked as possible anomalous when they surpass a specific benchmark. When applied to systems in the real world, this method excels at revealing anomalies, problems, or unforeseen circumstances. The Steps in Anomaly Detection: Input real-time data into the trained DBN model, Compare the output (predicted values) to the actual data, Measure deviations using a suitable error metric (e.g., mean squared error), Flag significant deviations as anomalies for further analysis or action.

f) Predictive Trend Modeling

Using historical and current information, the DBN is also used to predict patterns for the time being. The framework makes predictions about potential outcomes by sending input information throughout the communication system and generating recommendations. The analysis associated with these predictions can help with preemptive preparation and choice-making by revealing patterns, indicators, and probable potential results.

The predictive modelling is mathematically represented as:

$$y_{pred} = f(x;\theta) \tag{4}$$

Where y_{pred} The predicted output (forecasted trend) f The trained DBN model x The input vector, consisting of current and historical data θ The parameters of the DBN model, learned during training. The Steps in Predictive Trend Modeling: The DBN framework needs historical and current information.Let the communication system go through its layers, processing the incoming data.Examine the results produced by the model to spot patterns or predicted values.Use these forecasts when making plans and tweaks to the technique's effectiveness.

This structure offers one-stop shopping for automated systems by integrating the identification of anomalies with predicted trend modelling. Providing preventative projections and immediate information improves operating performance and dependability. This framework identifies and resolves possible problems before they worsen quickly. Be proactive in responding to evolving circumstances—Maximise efficiency by making well-informed choices.The framework's ability to serve multiple purposes makes it ideal for use in areas like intelligent production, managing energy, healthcare monitoring, and technologies based on the Internet of Things.

4. Experiment Evaluation

a) Mean Absolute Error (MAE)

The Mean Absolute Error (MAE) is a common statistic in predictive evaluation that considers defect course and size across expected and actual results (yPred y yPred), but not fault source. This simple indicator compares expectations to actual data by averaging their absolute differences. How to calculate MAE:

$$\text{MAE}=\frac{1}{n}\sum_{i=1}^{n}|y_{pred}^{i} - y_{actual}^{i}| \tag{5}$$

Data points are expressed as n—a realistic alternative for situations where prediction error must be recognized. MAE clearly shows how close the simulation's projections are to the real numbers. Medical surveillance can measure how well an algorithm predicts patient wellness measurements. Forecasting maintenance can measure how well a technology predicts technology expiration to improve decisions and streamline activities. MAE accurately represents strategy performance by focusing on the actual mistake rather than cancelling out both beneficial and detrimental mistakes.

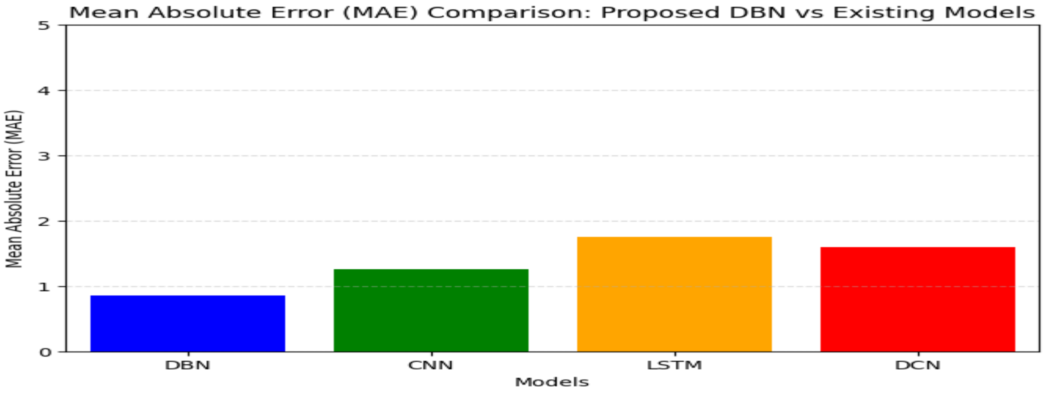

Figure. 4. Comparison graph for Mean Absolute Error (MAE)

The structures and MAE values are defined initially. Employing this bar graph shows algorithms on the X-axis and values of the MAE on the Y-axis. The Y-axis length is 0 to 5, and the X-axis range displays every design accurately. Lastly, an illuminated grid across the Y-axis improves MAE comparative visibility.

b) F1 Score

The F1 Score is crucial for assessing classification tasks, especially in imbalanced datasets like anomaly detection, where "anomalies" (positive instances) are far fewer than "normal" data points. This dataset appears to have a binary label with 0 representing standard data and 1 for anomalies.The F1 Score accounts for false positives (mislabeling normal data as anomalies) and false negatives (failing to detect anomalies) in a single performance measure. The form

$$F1\,Score = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \quad (6)$$

Where $Precision$the proportion of correctly identified anomalies out of all instances measures classified as anomalies. $Recall$ (or Sensitivity) measures the proportion of actual anomalies. If the dataset contains rare medical anomalies, the **F1 Score** ensures the model does not excessively favour identifying only the majority class (standard data). It balances the trade-off between identifying all anomalies (high Recall) and avoiding false alarms (high Precision).

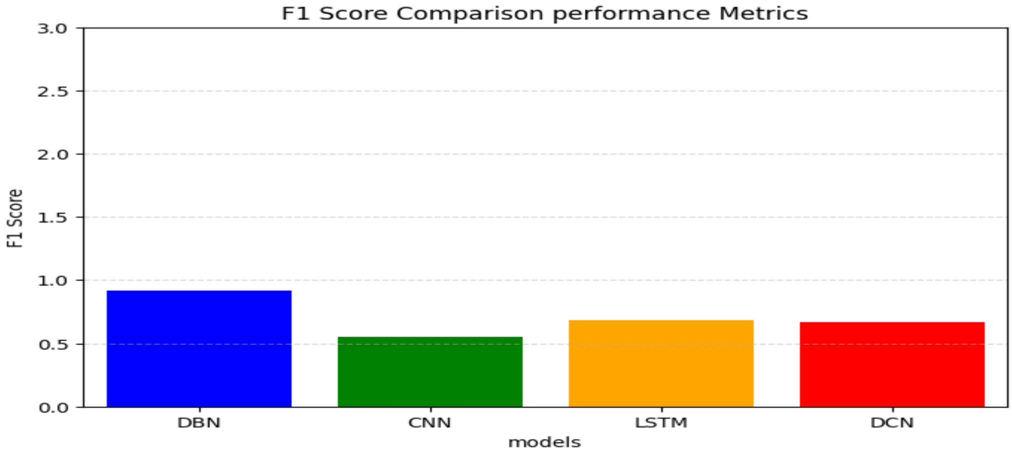

Figure. 5. Comparison graph for F1 Score

ISSN: 3006-8894
https://doi.org/10.70023/sahd/250209

It creates a bar graph evaluating the DBN model's F1 Score to CNN, LSTM, and DCN algorithms. It specifies the equations and F1 scores and then makes a bar graph with Matplotlib. Designs are on the X-axis, and F1 scores are on the Y-axis, with values ranging from 0 to 5. For simplicity, axis symbols and titles are provided. Each model fits comfortably in the X-axis range, and a Y-axis matrix enhances comprehension. This visualization allows for comparing the effectiveness of models using the F1 Score.

c) Precision Accuracy

The dataset may have a binary classification problem with a target column that indicates whether an observation is positive (e.g., an anomaly or specific condition) or hostile. Precision measures the percentage of positive instances correctly predicted out of all positive instances. Reducing misleading results helps identify fraud and medical and computer hacking. With accuracy, the algorithm's optimistic forecast is precise, minimizing false positives and intervention. Reliability is determined through the recognition of real positives and duplicate positives. This indicator indicates that modelling dependability has elevated the expenses for inaccurate results.

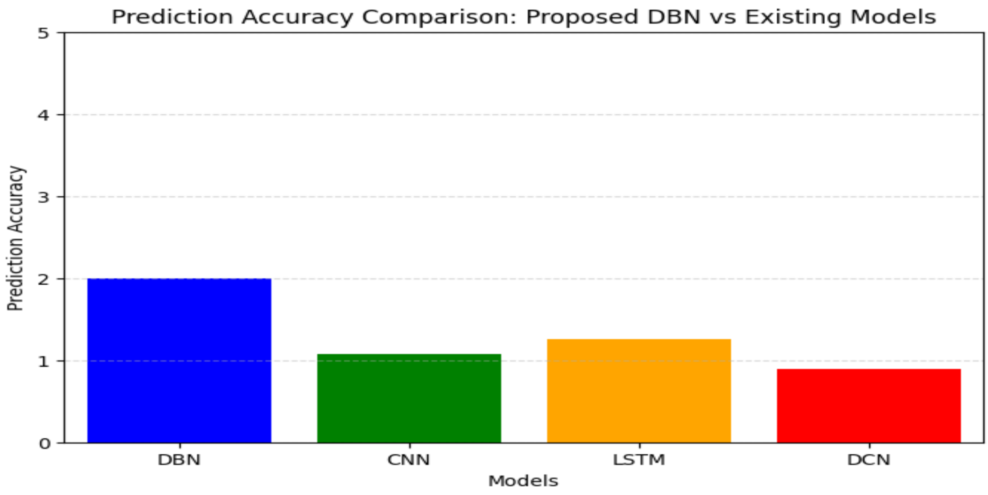

Figure. 6. Comparison graph for Precision

The graph contrasts the predictive power of the suggested DBN algorithm to that of pre-existing models (CNN, LSTM, and DCN). Lists provide algorithms and correctness metrics. A bar graph has been generated using Matplotlib, with the X-axis showing predictions and the Y-axis reliability of forecasts. For improved visualization, the Y-axis range is 0 to 5, and the X-axis separates types. For simplicity, axis labels, a title, and a grid along the Y-axis improve accurate comparative usability.

5. Conclusion and Future Work

The DeepSense Framework-based identification of anomalies and forecast pattern modelling solution uses Deep Belief Networks' resilient abilities. The structure was created to detect abnormalities and anticipate probable patterns instantaneously. The DeepSense Framework outperformed statistical models and neural network methodologies after thorough evaluation. Its low false positive rate and high anomaly detection sensitivity make it an appropriate choice for varied intelligence systems. The capacity to simulate complex, multidimensional data connections was essential for accurate trend assessments. More advanced methods like hybrid structures integrating DBNs with recurrent neural networks like LSTM or GRU might enhance timing forecasting. Online instruction might provide the structure to modify and change data patterns requiring reloading. Extending the infrastructure to accommodate audio, video, and information from sensor synthesis could render it suitable in further intelligent settings such as the Internet of Things and brilliant medical facilities. At last, investigation can optimize the structure's performance for peripheral gadgets.

References

[1]. Quraishi, A., Rusho, M. A., Prasad, A., Keshta, I., Rivera, R., & Bhatt, M. W. (2024, April). Employing Deep Neural Networks for Real-Time Anomaly Detection and Mitigation in IoT-Based Smart Grid Cybersecurity Systems. In 2024 Third International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE) (pp. 1-6). IEEE.

[2]. Maitra, S. (2024). A Data Mining-Based Dynamical Anomaly Detection Method for Integrating with an Advance Metering System. arXiv preprint arXiv:2405.02574.

[3]. Alzahrani, M. E. (2024). Elevating Smart Industry Security: An Advanced IoT-Integrated Framework for Detecting Suspicious Activities using ELM and LSTM Networks. International Journal of Advanced Computer Science & Applications, 15(2).

[4]. Rezaee, K., Rezakhani, S. M., Khosravi, M. R., & Moghimi, M. K. (2024). A survey on deep learning-based real-time crowd anomaly detection for secure distributed video surveillance. Personal and Ubiquitous Computing, 28(1), 135-151.

[5]. Demertzis, K., Iliadis, L., Tziritas, N., & Kikiras, P. (2020). Anomaly detection via blockchain deep learning smart contracts in industry 4.0. Neural Computing and Applications, 32(23), 17361-17378.

[6]. Manimurugan, S., Al-Mutairi, S., Aborokbah, M. M., Chilamkurti, N., Ganesan, S., & Patan, R. (2020). Effective attack detection in the Internet of Medical Things bright environment using a deep belief neural network. IEEE Access, 8, 77396-77404.

[7]. Abbas, Ghulam. "A Sequential Pattern Mining Method for the Individualized Detection of Online Banking Fraudulent Transactions." PatternIQ Mining.2024, (01)1, 34-44. https://doi.org/10.70023/piqm244

[8]. Elsayed, M. A., & Zulkernine, M. (2020). PredictDeep: security analytics as a service for anomaly detection and prediction. IEEE Access, 8, 45184-45197.

[9]. Nizam, H., Zafar, S., Lv, Z., Wang, F., & Hu, X. (2022). Real-time deep anomaly detection framework for multivariate time-series data in industrial IoT. IEEE Sensors Journal, 22(23), 22836-22849.

[10]. Ahmad, Z., Shahid Khan, A., Nisar, K., Haider, I., Hassan, R., Haque, M. R., ... & Rodrigues, J. J. (2021). Anomaly detection using deep neural network for IoT architecture. Applied Sciences, 11(15), 7050.

[11]. Wong, M. L., & Arjunan, T. (2024). Real-Time Detection of Network Traffic Anomalies in Big Data Environments Using Deep Learning Models. Emerging Trends in Machine Intelligence and Big Data, 16(1), 1-11.

[12]. El-Shafeiy, E., Alsabaan, M., Ibrahem, M. I., & Elwahsh, H. (2023). Real-time anomaly detection for water quality sensor monitoring based on multivariate deep learning technique. Sensors, 23(20), 8613.

[13]. Abdallah, A., Alkaabi, A., Alameri, G., Rafique, S. H., Musa, N. S., & Murugan, T. (2024). Cloud Network Anomaly Detection Using Machine and Deep Learning Techniques-Recent Research Advancements. IEEE Access.

[14]. Arjunan, T. (2024). Real-Time Detection of Network Traffic Anomalies in Big Data Environments Using Deep Learning Models. International Journal for Research in Applied Science and Engineering Technology, 12(9), 10-22214.