Veena Gadusu
(700754361)

Report

A Novel Two-Stage Deep Learning Model for Network Intrusion Detection: LSTM-AE

Introduction

In the rapidly evolving field of cybersecurity, the ability to effectively detect and classify cyber-attacks is crucial for maintaining network integrity and data security. Traditional intrusion detection systems (IDS) have relied heavily on signature-based methods and rule-based approaches, which often fall short against sophisticated and evolving threats. The integration of machine learning and deep learning techniques has significantly enhanced IDS capabilities by enabling more dynamic and adaptive threat detection.

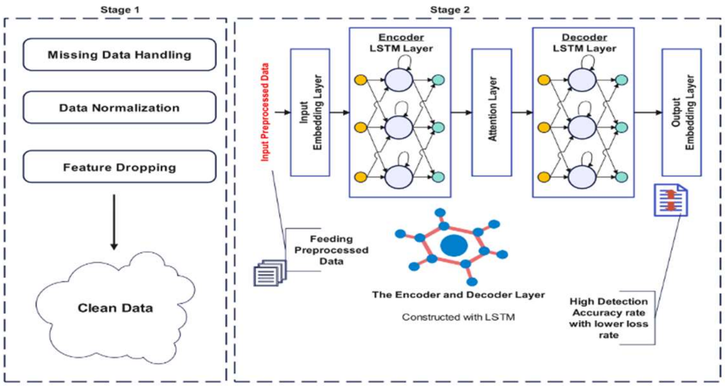
Deep learning models, such as Deep Neural Networks (DNN) and Convolutional Neural Networks (CNN), have shown promise in improving the accuracy and efficiency of IDS. However, as cyber-attacks become increasingly complex and varied, there is a growing need for more advanced models that can handle a wider range of attack scenarios and adapt to new threats

This necessity explores the development and evaluation of a novel hybrid deep learning model that combines **Long-Short Term Memory (LSTM) networks with Auto-Encoders (AE**) for intrusion detection. The proposed LSTM-AE model aims to address the limitations of existing methods by leveraging the strengths of both LSTM and AE to enhance attack detection and classification. By conducting a thorough performance analysis on the CICIDS2017 and CSE-CICDIS2018 datasets, this research evaluates the effectiveness of the LSTM-AE model compared to traditional DNN and CNN approaches.

The focus of this study is to identify a model that not only performs well on benchmark datasets but also demonstrates robustness and adaptability in detecting emerging cyber threats. Through this analysis, the research seeks to contribute valuable insights into the development of more effective IDS solutions and to advance the field of cybersecurity.

Summary

The proposed model integrates LSTM and AE to leverage their complementary strengths. LSTM is known for its ability to capture temporal dependencies and sequential patterns, while AE is effective for feature learning and dimensionality reduction.The LSTM-AE model is developed with a focus on creating a flexible and robust IDS capable of detecting and classifying various cyber-attacks.The study uses the CICIDS2017 and CSE-CICDIS2018 datasets for training and evaluating the model. These datasets are chosen due to their comprehensive representation of network traffic and attack scenarios.



Encoder Design:

The datasets are preprocessed to ensure they are suitable for training the deep learning models, including normalization and handling missing values.

Input: Sequential data from the network traffic logs or system event logs (e.g., timestamps, protocol types, source/destination IP addresses, etc.).

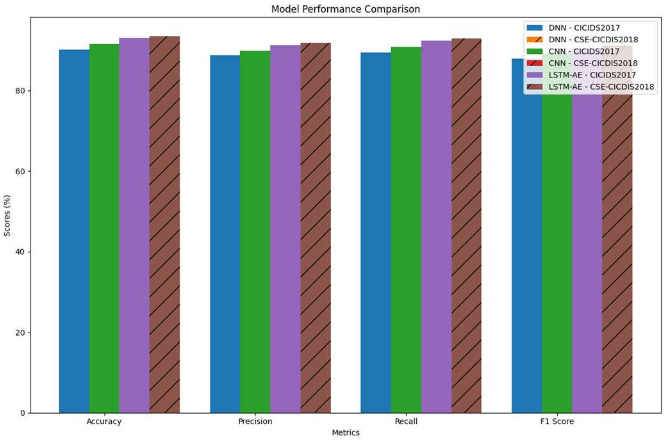
LSTM Layers: Multiple LSTM layers in the encoder part of the autoencoder. Each LSTM layer processes the input sequence and extracts features that capture temporal dependencies and patterns in the data.

Latent Representation: The final LSTM layer (decoder)outputs a compressed representation (latent space) of the input data. This latent representation is a compressed and abstracted form of the input sequence, capturing its essential features.

Score	**CNN**	**DNN**	**LSTM-AE**
Accuracy	98.7%	98.55%	**99.20%**
Loss	0.0245	0.0107	**0.0040**
Recall	98.6%	98.45%	**99.20%**
Precision	98.52%	98.31%	**99.12%**
F-Measure	98.86%	98.21%	**99.02%**

The performance of the LSTM-AE model is compared with traditional DNN and CNN models. Metrics used for evaluation include Accuracy, Precision, Recall, and F1 Score . The proposed LSTM-AE is compared with other models and is found to have a better accuracy in some models. The

LSTM-AE may be better suited for intrusion detection tasks compared to DNN and CNN because of its sequential data processing, feature extraction , anomaly detection, data efficiency.



Critical Analysis

This paper presents a compelling advancement in intrusion detection systems by proposing a hybrid model that combines Long-Short Term Memory (LSTM) networks with Auto-Encoders (AE). The model’s promising results in terms of improved detection accuracy and efficiency suggest a significant step forward in addressing the challenges of modern cyber-attacks.

One of the strengths of this paper is its clear demonstration of how the LSTM-AE model outperforms traditional deep learning approaches on the CICIDS2017 and CSE-CICDIS2018 datasets. This improvement in precision and recall indicates that the model effectively captures complex attack patterns and adapts to evolving threats. The careful optimization of network parameters also highlights a strong commitment to enhancing the model's performance and practical applicability.

Although this article focuses on specific datasets, this choice provides a solid foundation for evaluating the model's capabilities. Furthermore, researchers can consider employing ensemble methods, which involve combining multiple LSTM-AE models to enhance their overall performance. This approach can help to mitigate the impact of overfitting and improve the model’s ability to generalize to new data Future work could expand on this by testing the model across a wider range of datasets and real-world scenarios, which would further validate its robustness and generalizability. The potential for the model to handle diverse network conditions and attack types is promising and could lead to significant improvements in real-time intrusion detection.

Conclusion

The LSTM Autoencoder (AE) framework presents a powerful solution for anomaly detection in Intrusion Detection Systems (IDS), leveraging its strengths in processing sequential data and capturing complex temporal dependencies. By employing multiple LSTM layers in the encoder, the framework effectively learns to compress and reconstruct normal behaviors from network traffic or system logs. During deployment, the LSTM AE monitors incoming data streams, detecting anomalies through deviations in reconstruction errors from learned normal patterns. This unsupervised learning approach is particularly advantageous in IDS, where labeled intrusion data may be limited. However, successful implementation requires careful consideration of training data quality, hyperparameter tuning, and efficient deployment strategies to ensure accurate and timely anomaly detection in real-world scenarios. Overall, the LSTM AE encoder framework represents a sophisticated tool for enhancing cybersecurity defenses by identifying unauthorized access and malicious activities within computer networks.

Code:

```
import numpy as np
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM, Dense, TimeDistributed, RepeatVector
from tensorflow.keras.callbacks import EarlyStopping
# Dummy data (replace with actual datasets)
# X_train, y_train, X_test, y_test = load_dataset('CICIDS2017', 'CSE-CICDIS2018')
# Define LSTM-AE model
model = Sequential()
model.add(LSTM(units=64, input_shape=(sequence_length, num_features), return_sequences=False))
model.add(RepeatVector(sequence_length))
model.add(LSTM(units=32, return_sequences=True))
model.add(TimeDistributed(Dense(num_features)))
model.compile(optimizer='adam', loss='mse')
# Train the model
early_stopping = EarlyStopping(monitor='val_loss', patience=3, restore_best_weights=True)
history = model.fit(X_train, X_train, epochs=20, batch_size=64, validation_split=0.2, callbacks=[early_stopping])
# Evaluate the model
loss = model.evaluate(X_test, X_test)
print(f'Test Loss: {loss}')
# Example of predicting and detecting attacks (pseudo-code)
predictions = model.predict(X_test)
anomaly_scores = np.mean(np.abs(predictions - X_test), axis=-1)
```

```
 # Assume a threshold for anomaly detection

 threshold = 0.1

 detected_anomalies = np.where(anomaly_scores > threshold)[0]

print(f'Number of detected anomalies: {len(detected_anomalies)}')
```

This code structure outlines the general flow of implementing a deep learning model (LSTM-AE) for intrusion detection as described in the abstract .