EAAW XI

Deep Learning-Based Models for Malicious File Segregation in Naval Networks

Aarzoo Gosain, B.E. (I.T.) – Netaji Subhas Institute of Technology, University of Delhi
Aseem Gosain, B.Tech. (Gold Medallist), M.Tech – IIT BHU Varanasi
Cdr. Rakesh Kumar Gosain, Retd. – Indian Navy

Corresponding Author Email: aseemgosain@gmail.com

Introduction - Advancing Naval Cyber-Security

01 **Evolving Threat Landscape**

Naval networks face increasing threats from ransomware, spyware, and denial-of-service attacks.
These threats target operational security and confidentiality.

02 **Limitaions of Manual File Segregation Model**

Entropy-based file segregation model required manual operation and yielded high false positive rates.
Balanced accuracy of previous model was limited to 51.5%.

03 **Need for Automation**

Traditional Deep Packet Inspection is resource-intensive and unsuitable for real-time defence.
Deep learning automates classification by analyzing file bytes like natural language sequences.

04 **Scope of Research**

Focuses on CNN, LSTM, BERT-Large, and CodeBERT neural architectures for file classification.
Application in real-time Naval Ship traffic to segregate benign and malicious files effectively.

Methodology - Training and Evaluation Pipeline

01

Data Collection

Used the Dike Dataset: 10,000 malware and 1,000 benign files (PE and OLE types).

02

Balancing the Dataset

Random subsampling to mitigate class imbalance ensures 50:50 ratio of benign to malicious files.

03

Model Training

CNN, LSTM, BERT-Large, and CodeBERT models trained on byte-level patterns.
Trained with balanced mini-batches using AdamW optimizer over five epochs.

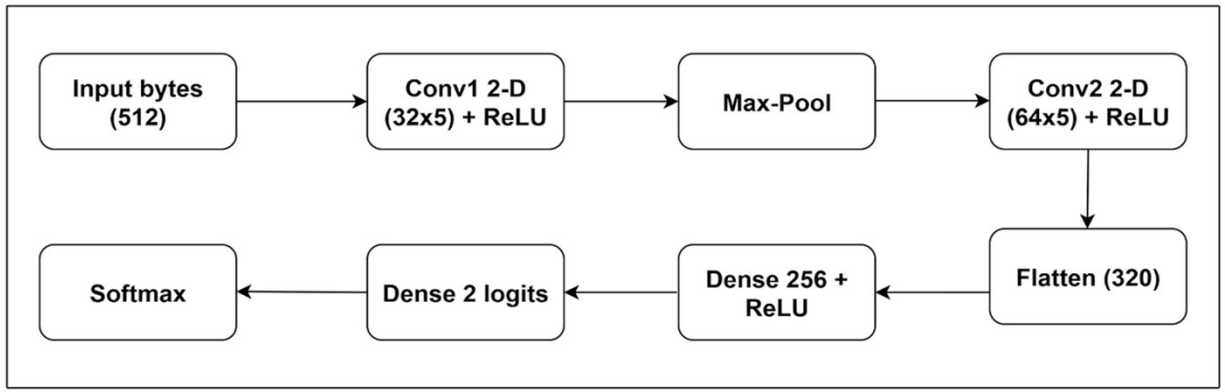
04

Evaluation Metrics

Measured accuracy, balanced accuracy, precision, recall, F1-score, and inference time for each model.

Experimental Set-Up : Convolutional Neural Network

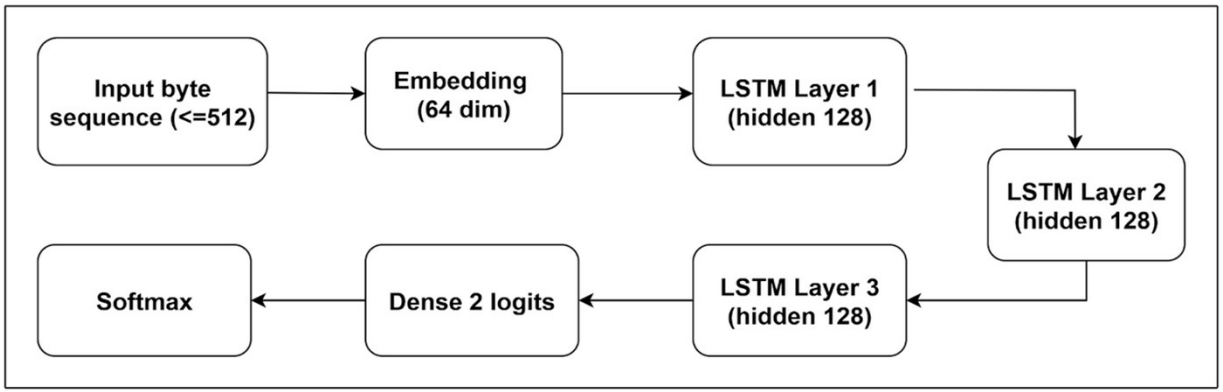
- Utilizes two convolutional layers with 1.2 million parameters.
- The first layer has 32 filters scanning 5-byte windows; the second layer has 64 filters detecting higher-order patterns.
- Produces a 320-dimensional feature vector, followed by dense layers classifying the file as benign or malicious.
- Noted for its speed and efficiency, especially suitable for real-time processing aboard Naval Ships.



Baseline CNN Architecture

Experimental Set-Up : Long Short-Term Memory Network

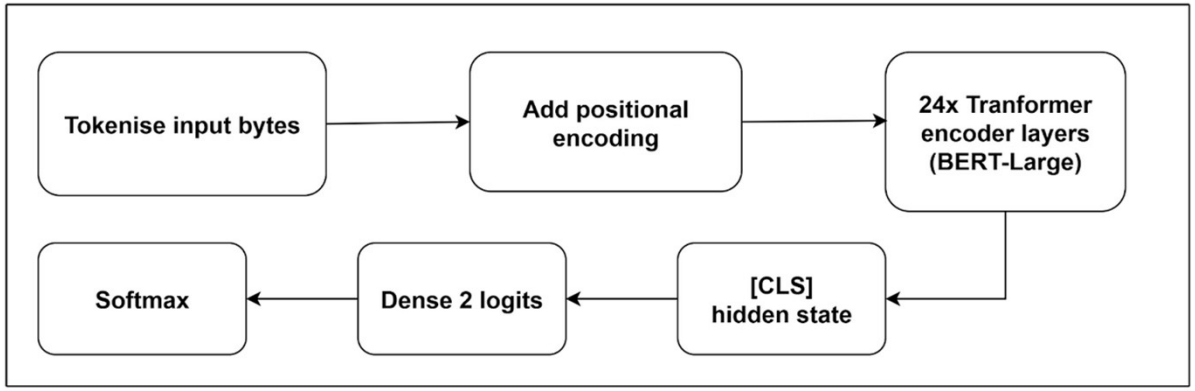
- Comprises three stacked LSTM layers with 2.3 million parameters.
- Processes byte sequences one at a time, capturing long-term dependencies.
- Each layer builds upon the memory of the previous one, allowing complex pattern recognition even from distant byte positions.
- Moderately slower than CNN but effective for catching stealthy patterns in sequential data.



Three-Layer LSTM Architecture

Experimental Set-Up : Bidirectional Encoder Representations from Transformers (BERT) - *Large*

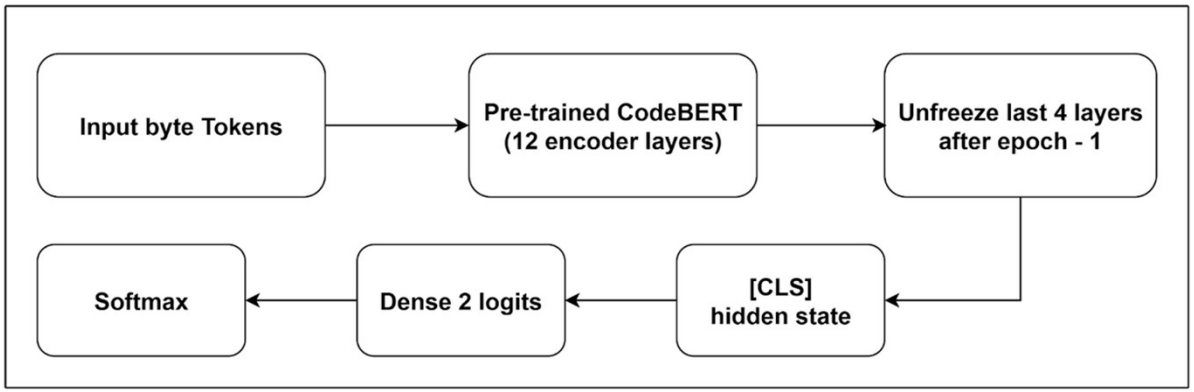
- A 340M parameter transformer model pre-trained on natural language.
- Processes 512-byte windows using self-attention to model inter-byte relationships.
- Trained for 5 epochs with a learning rate of 2×10^{-5} on AWS SageMaker with NVIDIA L4 GPU.
- Delivered low balanced accuracy (54.5%) due to domain mismatch and limited dataset fine-tuning.
- High inference time of 24.5 ms per file makes it impractical for real-time Naval deployment.



Bert-Large Architecture

Experimental Set-Up : CodeBERT Network

- A 125M parameter transformer pre-trained on natural + programming languages.
- Fine-tuned on byte-level malware data with progressive layer unfreezing from epoch one.
- Processes 512-byte input chunks, optimized using the AdamW optimizer.
- Achieved 95.4% balanced accuracy and 0.984 F1-score, the best among all models.
- Inference time of 13.9 ms is acceptable for deeper inspection in hybrid configurations.



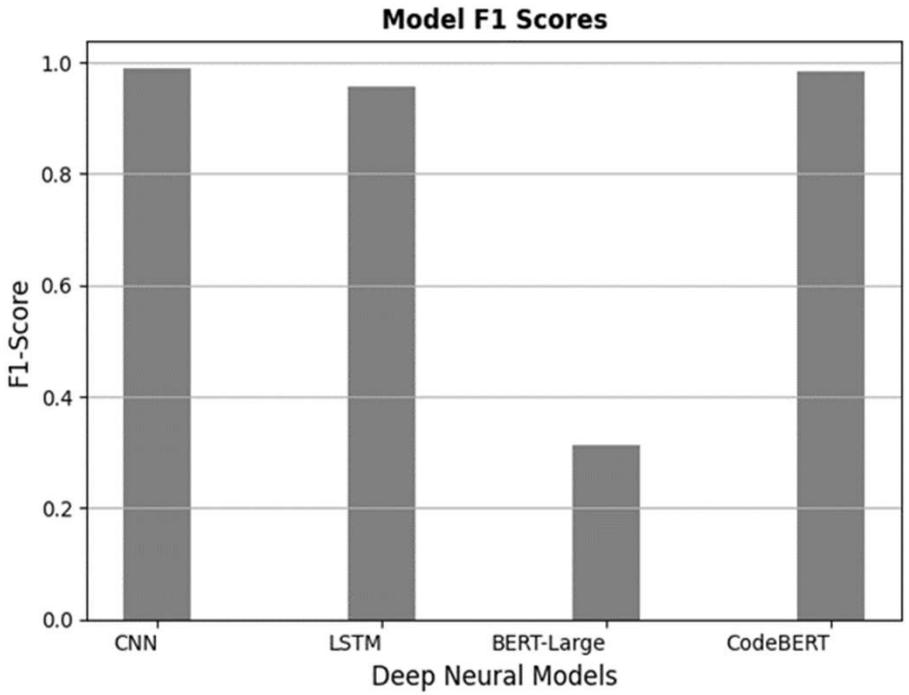
CodeBERT Architecture as used in Recommendation Model

Results and Performance Comparison

Deep Neural Model	Accuracy (%)	Balanced accuracy (%)	Precision	Recall	F1 Score	False Positive (%)	Inference Time (ms)	Parameter count (Million)
CNN	97.8	93.5	0.988	0.987	0.988	1.1	0.1	1.2
LSTM	92.5	92.3	0.991	0.926	0.957	0.7	0.3	2.3
BERT-Large	25.3	54.5	0.95	0.188	0.314	0.9	24.5	340
CodeBERT	97.2	95.4	0.995	0.975	0.984	0.4	13.9	125

Performance of Deep Models on Dike Dataset

Analysis



- CodeBERT has the highest balanced accuracy, around 95 % and CNN has the least inference time
- LSTM is accurate but takes three times as long as the CNN
- BERT-Large could not deliver good accuracy because it is naturally pre-trained on English language data, and hence, it is not suitable for this malware file traffic classification research.
- In the CodeBERT model, which is originally pre-trained on both English language and programming language data, CodeBERT's balanced accuracy is better than that of the CNN, but the model is much slower in comparison.

Proposed Framework : Hybrid CNN + CodeBERT Model

- The CodeBERT model is slower yet has the highest balanced accuracy
- CNN has the least inference time but falters on harder encoded malware cases
- A fully automated Hybrid CNN + CodeBERT model is proposed with its three-stage workflow as described below

Stage 1: Fast Scan with CNN

Quickly classifies benign files with 0.1 ms inference, reducing latency on routine traffic.

Stage 2: Deep Analysis with CodeBERT

Suspicious files undergo CodeBERT inspection with enhanced accuracy using code-level semantics.

Stage 3: Decision Merge

Final verdict is based on combined outputs; suspected malware is quarantined while safe files are released.

Efficiency Gains

Reduces false positives by 60% while maintaining average latency under 0.5 ms.

Naval Applications of the Recommendation Model

01

Smart Decoy Systems

Secures AI-based drone and decoy systems by filtering command packets in real-time under constrained bandwidth.

02

Radar Signal Integrity

Mitigates sea and ground clutter by filtering malicious radar signal patterns and dynamically adjusting thresholds.

03

Anti-Jamming Technology

Prevents adversarial interference in adaptive filtering systems by ensuring only clean commands are executed.

04

D3IL Compatibility

Designed for Denied, Degraded, Disrupted, Intermittent, or Limited communication scenarios with full offline inference.

Conclusion: Enhancing Naval Cyber Resilience

Challenges Addressed

- Manual model lacked speed and learning capabilities.
- The Manual file segregation tool had only a single-feature check, it gave a high rate of false alarms and it had no learning or adapting capabilities.
- Deep learning enables real-time classification with high accuracy and low latency.

Hybrid Model Success

- The proposed Hybrid CNN + CodeBERT framework combines the speed of CNN and accuracy of CodeBERT for a robust file segregation.

Model Evaluation

- Among all models, CodeBERT delivered the highest balanced accuracy, while CNN offered the fastest inference time.
- CodeBERT delivered the balanced accuracy of 95.4 %, an F1-score of 0.984 and the CNN had an inference time of only 0.1 ms per file.
- BERT can only deliver higher balanced accuracy on larger datasets and with more training epochs.

Future Potential

- Scalable to the Naval Command and Control Systems where secure real-time data filtering is vital.

Important References

1. Gosain, A., & Gosain, RK. (2022). Preliminary investigation method for segregating malware-encrypted files from the regular traffic. Institute of Marine Engineering, Science and Technology (IMarEST). https://doi.org/10.24868/10650

2. GitHub 2022, 'DikeDataset - labeled dataset containing benign and malicious PE and OLE files', GitHub, viewed 20 May 2025, <https://github.com/iosifache/DikeDataset>.

3. Wang, W., Zhu, M., Zeng, X., Ye, X. and Sheng, Y., 2017, January. Malware traffic classification using convolutional neural network for representation learning. In 2017 International conference on information networking (ICOIN) (pp. 712-717). IEEE.

4. Ullah, F., Ullah, S., Srivastava, G. and Lin, J.C.W., 2024. IDS-INT: Intrusion detection system using transformer-based transfer learning for imbalanced network traffic. Digital Communications and Networks, 10(1), pp.190-204.

5. Feng, Z., Guo, D., Tang, D., Duan, N., Feng, X., Gong, M., Shou, L., Qin, B., Liu, T., Jiang, D. and Zhou, M., 2020. Codebert: A pre-trained model for programming and natural languages. arXiv preprint arXiv:2002.08155.

Thank You