LSTM based IDS on ADFA-LD dataset

Amber Gupta (2024AIM1001)
Yash Narnaware (2024AIM1011)
Prince Raj (2024AIM1012)
Aksshay Mathew P (2024AIM1013)

Team 1

IIT ROPAR

14-NOVEMBER-2024

Contents

Motivation

- **Growing Threat of Cyber-attacks:** Cyber-attacks are becoming more common and sophisticated.
- **Need for Intelligent Detection:** There is a need for smarter, adaptive systems that can detect unknown attack patterns.
- **Deep Learning for Better Detection:** Deep learning, particularly models that process sequences of data, can spot complex and subtle attack patterns.
- **Our Approach:** LSTM model to analyze system call sequences and classify/detect intrusions.

Introduction to ADFA-LD Dataset

- **ADFA-LD** dataset contains System Call sequences from both normal and malicious activities, used for evaluating intrusion detection systems.

Data Type	Trace Count
Training Traces	833
Validation Traces	4373
Attack Traces	746

Attack Type	Trace Count
Normal	833
Hydra_SSH	176
Hydra_FTP	162
Java_Meterpreter	124
Web_Shell	118
Adduser	91
Meterpreter	75

Dataset Preprocessing for Binary-Classifier

- **Initial Setup:** Downloading **ADFA-LD** dataset from Github Repository and Unzipping in Google Colab.
- Building **Dataset** with all System Calls along with their Attack_Types & Labels(0 for Normal and 1 for Attack)
- **Test_Train Split:** Splitting dataset 80% for training and 20% for testing.
- **Padding** is added after considering 90^{th} percentile of the max_length
- **Validation_Data** contains System call for Validation purposes.

Dataset Preprocessing for Multi-Classifier

- Building **Dataset** with all System Calls along with their Attack_Types & Labels(0 for Normal and 1,2,3,4,5,6 for consecutive Attacks)
- **Over Sampling** to increase the number of samples of the minority class.
- **Under Sampling** to decrease the number of samples of the majority class.
- **New class distribution in training set:** (0: 350, 1: 280, 2: 280, 3: 280, 4: 280, 5: 280, 6: 280)

Introduction to Word2Vec

- **Word2Vec** is a technique that represents words as vectors, capturing semantic and syntactic similarities.
- Developed by **Google**, it uses two main architectures: **Continuous Bag of Words (CBOW)** and **Skip-gram**.
- **Application in NLP:** Word2Vec creates dense embeddings where semantically similar words have similar vector representations, making it useful for a variety of NLP tasks.
- **Application of LSTM in HIDS:** Due to its ability to capture sequential patterns, LSTM models can learn temporal dependencies in system logs, making them effective for detecting suspicious behavior.

Using Word2Vec in HIDS

- Word2Vec embeddings can be applied to encode sequences in logs or network events, transforming them into numeric vectors.
- **Enhanced Context**: N-grams capture short-term, adjacent dependencies in system calls,
- **Enhancing Detection:** By using embeddings, patterns can be learned more effectively, improving anomaly detection.
- **Combination with LSTM:** When combined with LSTM, Word2Vec enhances the model's ability to learn meaningful sequences in intrusion data.

Multi-Class Model Classifier Architecture - Diagram

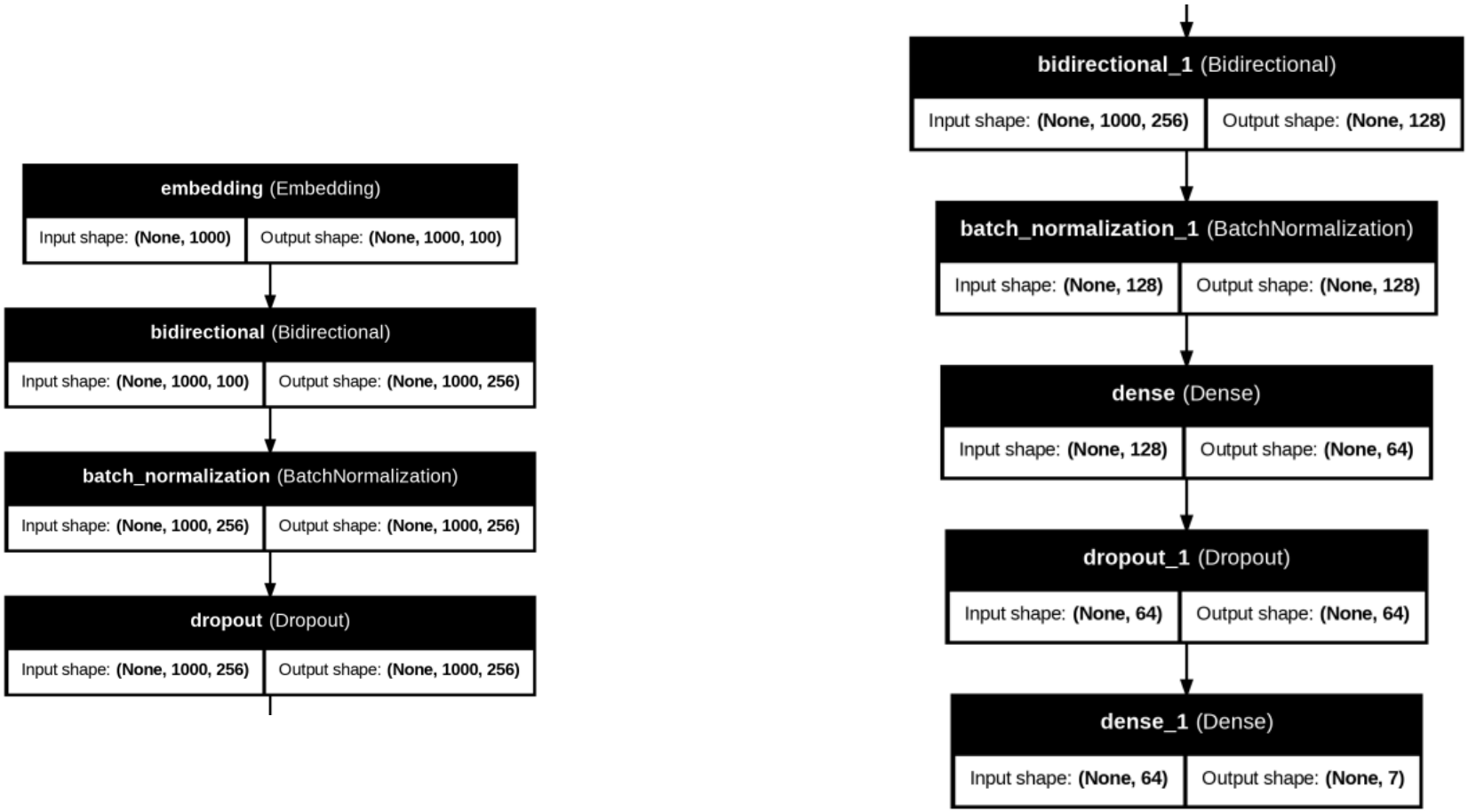

Figure 1: Model Architecture - multi-class classifier

Multi-Class Classifier Model Architecture - Details

- Trained for 30 epochs
- Used batch normalization and dropout layers to improve the model accuracy.
- Bi-directional LSTM instead of normal LSTM significantly improved accuracy.
- Tried a more complex architecture but encountered overfitting.
- Achieves around 80% accuracy on validation data.

Binary Model Classifier Architecture - Diagram

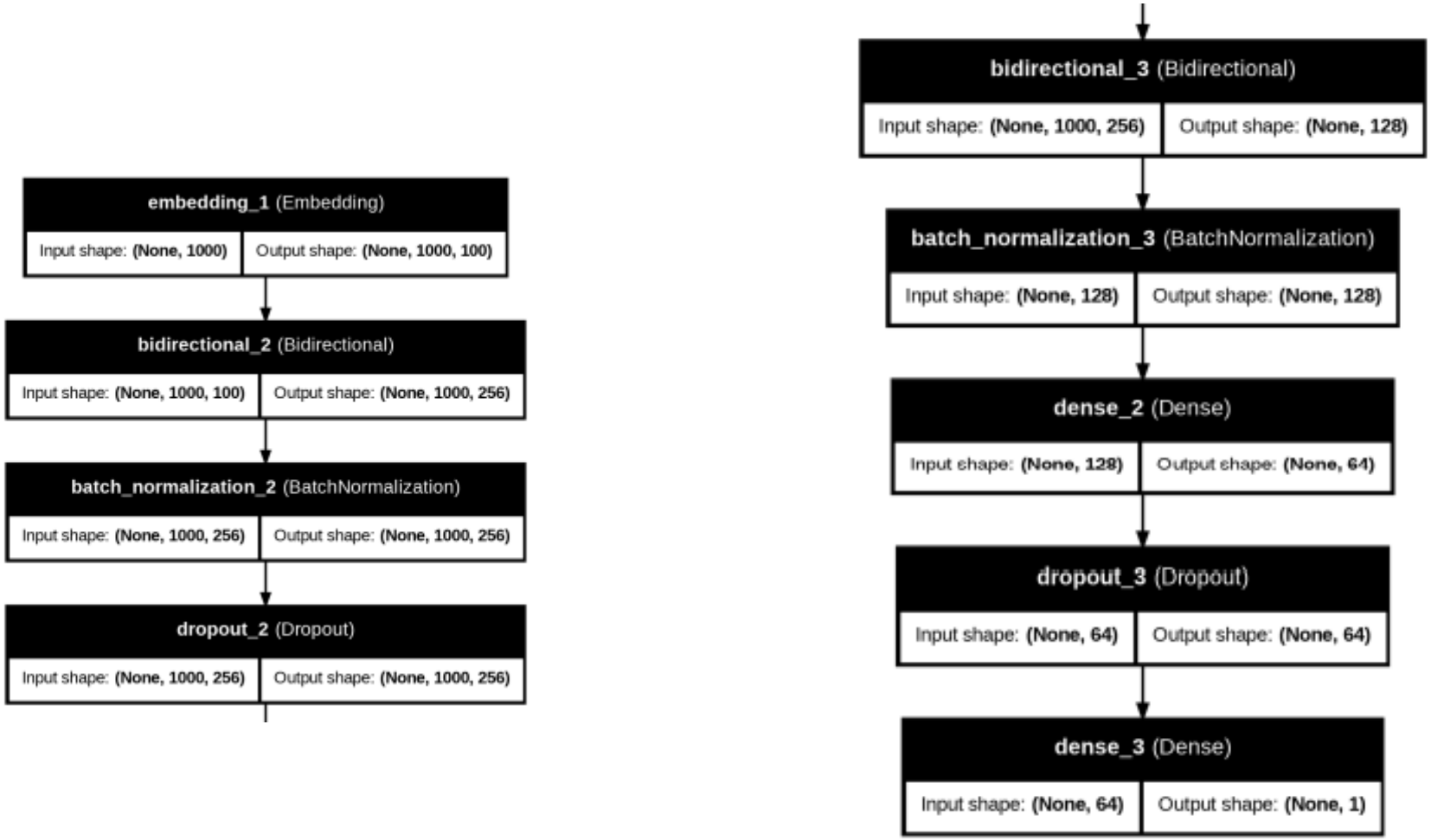

Figure 2: Model Architecture - binary classifier

Binary Classifier Model Architecture - Details

- Similar architecture as the multi-class classifier, with modifications in the last layer.
- Trained for 15 epochs to avoid overfitting.
- On a validation dataset, the model achieved an accuracy of 81-83%.

Results and Discussion

- Other ML algorithms appear to be more efficient than LSTM

Algorithm	**Accuracy (%)**
Naive Bayes	58.814
SMO	87.683
LibSVM	87.565
IBk (k=1)	92.186
IBk (k=2)	92.405

Table 1: Performance Results for Multiclass Classification on the ADFA-LD Dataset

1

[1] B. Borisaniya and D. Patel, âEvaluation of modified vector space representation using ADFA-LD and ADFA-WD datasets,â Journal of Information Security, no. 6, pp. 250â264, 2015.

Results and Discussion

Algorithm	Accuracy (%)
Naive Bayes	69.94
SMO	88.32
LibSVM	87.87
IBk (k=1)	95.95
kMeans (k=2)	70.64
ZeroR	87.46
OneR (B=5)	87.15
JRip	95.82
J48	95.98

Table 2: Performance Results for Binary Classification on the ADFA-LD Dataset

[2]

[2]B. Borisaniya and D. Patel, âEvaluation of modified vector space representation using ADFA-LD and ADFA-WD datasets,â Journal of Information Security, no. 6, pp. 250â264, 2015.

Conclusion

- **Overview:** Developed an Intrusion Detection System (IDS) using LSTM-based classifiers (binary and multiclass) on the ADFA-LD dataset.
- **Results:** Achieved an accuracy of around 80-83% with both models.
- **Insights:**
  - Bi-directional LSTM showed better performance than regular LSTM.
  - LSTM models may be less effective for IDS compared to other machine learning algorithms.
  - Alternative machine learning models could potentially improve IDS performance.