

Research Paper

Network Intrusion Detection Method Based on CNN-BiLSTM-Attention Model

Rubeena Begum,
PG Scholar, Department of Information
Technology, Shadan College of
Engineering and Technology,
Hyderabad, Telangana, India – 500086.
rubeenabegum72@gmail.com

Subramanian K.M.,
Professor, Department of Computer
Science and Engineering, Shadan
College of Engineering and
Technology, Hyderabad, Telangana,
India – 500086.
kmsubbu.phd@gmail.com

Imtiyaz khan
Professor, Department of Information
Technology, Shadan College of
Engineering and Technology,
Hyderabad, Telangana, India – 500086.
imtiyaz.khan.7@gmail.com

Abstract— Network intrusion detection plays a crucial role in cybersecurity, requiring robust and intelligent methods to identify and mitigate malicious activities. This study explores an advanced deep learning-based approach utilizing NSL-KDD, UNSW-NB15, and CIC DDoS2019 datasets to enhance intrusion detection accuracy. A comprehensive set of machine learning and deep learning techniques is employed, incorporating convolutional and recurrent neural network architectures to capture spatial and sequential dependencies in network traffic data. The proposed method integrates deep feature extraction and sequential pattern recognition with an attention mechanism to enhance detection performance by focusing on critical attack characteristics. Experimental evaluations demonstrate that the Voting Classifier (BaggingDT + BagRF) achieves the highest performance, attaining 98.1% accuracy on the CIC-DDoS2019 dataset, 99.9% accuracy on the NSL-KDD dataset, and 92.9% accuracy on the UNSW-NB15 dataset. These results highlight the effectiveness of ensemble learning in improving detection precision, recall, and generalization across diverse intrusion patterns. The findings indicate that integrating multiple classifiers and optimizing feature representation significantly enhances network intrusion detection, providing a scalable and efficient solution for real-world cybersecurity applications.

“Keywords— Network intrusion detection, CNN, BiLSTM, attention mechanism, Voting Classifier, CIC-IDS 2019, NSL-KDD, UNSW-NB15”.

I. INTRODUCTION

Network security is facing increasing challenges due to the rapid evolution of cyber threats, with attackers leveraging sophisticated techniques to infiltrate systems, steal sensitive data, and compromise infrastructure integrity [1]. The ability to detect and mitigate such intrusions is a critical area of research, leading to the development of intrusion detection systems (IDS). These systems are designed to identify and

respond to malicious activities within a network, ensuring the security and reliability of digital communications [2]. However, existing IDS solutions encounter significant challenges that affect their performance and reliability. One of the primary issues is the class imbalance problem, where the detection rate for minority attack classes remains insufficient due to the uneven distribution of intrusion data [3]. This imbalance leads to poor detection of rare but critical attack types, leaving networks vulnerable to sophisticated cyber threats. Moreover, traditional IDS approaches struggle to effectively capture spatial and temporal features in network traffic, limiting their ability to analyze intrusion data comprehensively [4].

To address the class imbalance issue, researchers have explored various techniques, including data-level approaches and cost-sensitive learning methods [5]. The data-level approach focuses on modifying the training dataset to create a more balanced distribution of attack classes. Common methods include oversampling the minority class and downsampling the majority class, which help improve the model's ability to detect less frequent attack patterns [6]. A widely used technique in this regard is the Synthetic Minority Over-Sampling Technique (SMOTE), which generates synthetic data points to enhance the representation of minority attack types. Additionally, the integration of Gaussian Mixture Model (GMM) technology has been proposed to refine the data augmentation process and improve IDS performance [7]. These approaches aim to enhance the detection capabilities of IDS by ensuring that the learning model is not biased toward the majority class.

Furthermore, hybrid deep learning approaches have been introduced to improve the feature extraction process and enhance intrusion detection accuracy. Models combining Convolutional Neural Networks (CNN) and Bidirectional Long Short-Term Memory (BiLSTM) networks have demonstrated improved performance by capturing both spatial and sequential dependencies in network traffic data [8]. Attention mechanisms have also been integrated into BiLSTM-based IDS models to enhance feature representation, resulting in higher detection rates and

reduced false positives [9]. Additionally, hierarchical CNN-attention models have been proposed to refine the classification of network attacks, allowing IDS to process complex intrusion patterns more efficiently [10]. These advancements highlight the ongoing efforts to improve network security through enhanced intrusion detection techniques, addressing the limitations of traditional methods and ensuring more effective defense mechanisms against evolving cyber threats.

II. RELATED WORK

Intrusion detection systems (IDS) have improved to combat sophisticated cyberattacks on network systems. Improved detection accuracy, class imbalance handling, and efficient temporal and spatial feature extraction from network data are key areas of IDS research. Due to these challenges, deep learning, hybrid models, and cost-sensitive learning methodologies have been developed to improve IDS performance.

Several research have examined intrusion detection class imbalance. Synthetic Minority Over-Sampling Technique (SMOTE) balances class distribution by generating synthetic minority class samples, improving uncommon attack detection accuracy. Researchers have combined SMOTE with Gaussian Mixture Models (GMM) to improve sampling and IDS model robustness [11]. SMOTE and GMM reliably handle class imbalance and preserve data distribution, boosting the IDS's capacity to detect rare attacks. This strategy improves the identification of assaults that are rare in the training data, a major difficulty for most standard IDS methods.

Deep learning models have also been studied in intrusion detection. CNNs and RNNs, especially Bidirectional Long Short-Term Memory (BiLSTM) networks, are commonly used to capture geographical and temporal patterns in network traffic data. Peng et al. [12] suggested a CNN-BiLSTM hybrid model for sequential data processing and feature extraction. Deep learning helped handle network traffic complexity by outperforming previous machine learning methods in detection accuracy. The hybrid model can detect short-term and long-term network data interdependence, which is essential for detecting sophisticated and evolving cyberattacks.

Integrating attention mechanisms improved IDS models. A multi-head attention mechanism was added to a BiLSTM model by Zhang et al. [13] to better focus on network traffic data characteristics. The model can detect more threats because to the multi-head attention technique, which captures many data points at once. This strategy outperforms conventional models, which fail to prioritise relevant aspects from high-dimensional network data. Learning to focus on key traits improves the model's ability to discern benign from malicious communications.

Another intrusion detection method uses hybrid deep learning models with CNNs and attention algorithms. Ren et al. [14] introduced a hierarchical CNN-attention model that employed CNNs to extract features and an attention method to highlight significant characteristics for classification. This two-stage technique handles complicated intrusion patterns with spatial and temporal connections to increase detection

performance. Attention methods with CNNs help the model focus on the most important network traffic, improving classification accuracy.

Also popular is integrating feature selection with deep learning models. Halbouni et al. [15] investigated how CNN and LSTM with feature selection could increase IDS performance. The hybrid CNN-LSTM model and feature selection algorithms identified and retained the most interesting network traffic data features. This method minimises deep learning model training computational costs and improves system detection. Selecting the most relevant attributes helps the model detect assaults while being computationally efficient.

IDS research is also using autoencoders for anomaly detection. The autoencoder-SVM hybrid model used by Khan and Mailewa [16] detected network transmission anomalies in NSL-KDD multi-class datasets. The autoencoder learnt the network's regular behaviour and identified anomalies. A support vector machine (SVM) added a robust classifier for differentiating normal and abnormal traffic, improving the model's performance. This method is useful for improving IDS robustness in dynamic contexts since it detects new threats.

Integration of statistical approaches with deep learning has also been studied. Kumar et al. [17] improved IDS systems utilising UNSW-NB15 dataset statistical analysis. The authors optimised feature extraction and deep learning-based classifiers by analysing the dataset for patterns and correlations. Statistical approaches and deep learning help IDS systems make better decisions by combining data-driven learning with domain-specific knowledge.

Additionally, new research has optimised intrusion detection model architecture. Wu et al. [18] presented MARNet, a two-stage intrusion detection model using supervised and unsupervised learning. Unsupervised learning detects novel attack patterns in the first stage, whereas supervised learning classifies known attacks in the second. This hybrid technique improves the detection of known and unexpected attacks, solving the shortcomings of classic IDS systems.

Hybrid optimisation techniques for IDS feature selection have been investigated. Sakthi and Nirmal Kumar [19] used deep learning and Harris Hawks Optimisation (HHO) to choose optimal IDS features. The HHO algorithm selected the most relevant features, decreasing feature redundancy and enhancing detection accuracy in the deep learning model. This hybrid technique trains IDS systems with only the most important features, improving efficiency and accuracy.

In addition, new research has improved IDS model scalability for big networks. For robust autonomous vehicle anomaly detection, Alotaibi et al. [20] developed an IDS model using the Convolutional Bi-LSTM network and a two-tier optimisation approach. The two-tier optimisation improves the model's ability to process massive amounts of network traffic, keeping IDS systems effective on high-speed, big-scale networks.

These IDS research advances demonstrate continued efforts to increase detection accuracy, handle class

imbalance, and extract meaningful information from network traffic data. Recent research have improved network infrastructure security and resilience by solving traditional IDS system difficulties with deep learning, feature selection, and hybrid models.

III. MATERIALS AND METHODS

The proposed system aims to enhance network intrusion detection by leveraging a hybrid deep learning framework that integrates convolutional and recurrent architectures with attention mechanisms. The approach employs DNN, CNN, CNN+LSTM, CNN+BiLSTM, CNN+BiLSTM with Attention, and CNN+GRU with Attention to extract spatial and temporal patterns in network traffic data, improving feature representation and detection accuracy. Additionally, ensemble learning techniques such as Random Forest, AdaBoost, KNN, Decision Tree, and a Voting Classifier (BaggingDT + BagRF) are utilized to enhance model robustness and generalization across diverse attack patterns. The integration of BiLSTM with multi-head attention further improves classification performance by focusing on critical features [2], while CNN-based hierarchical attention mechanisms effectively capture network anomalies [3]. The proposed system addresses class imbalance issues using deep learning-based approaches with focal loss [1] and optimizes feature learning through hybrid models [4]. This comprehensive framework enhances detection efficiency, adaptability, and scalability for real-time cybersecurity applications.

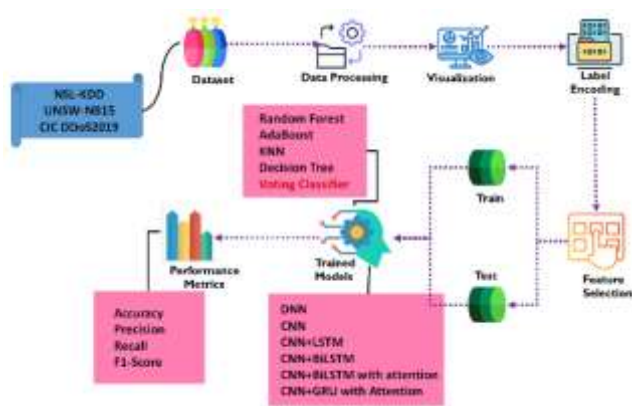


Fig.1 Proposed Architecture

This diagram (Fig.1) outlines a machine learning workflow for network intrusion detection using the NSL-KDD, UNSW-NB15, and CIC DDoS2019 datasets. The process begins with data collection and preprocessing, followed by feature selection and label encoding. Various machine learning models, including Random Forest, AdaBoost, KNN, Decision Tree, Voting Classifier, DNN, CNN, CNN-LSTM, CNN+BiLSTM, and CNN+GRU with attention mechanism, are trained and tested. Performance is evaluated using metrics like accuracy, precision, recall, and F1-score. The trained models are then deployed for network security. Visualization plays a key role in understanding data patterns and model performance.

A) Dataset Collection

The CIC-DDoS 2019, NSL-KDD, and UNSW-NB15 datasets are commonly used in cybersecurity research to analyze and detect various types of network intrusions, including DDoS attacks and other malicious activities, providing labeled data for model training and evaluation.

The CIC-DDoS 2019 dataset was designed to evaluate intrusion detection systems, specifically for Distributed Denial-of-Service (DDoS) attacks. It contains traffic features like flow duration, packet lengths, flow rates, and inter-arrival times. This dataset includes multiple attack scenarios, offering a comprehensive foundation for evaluating the performance of intrusion detection models [11].

Year	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Total	Avg
1	1	2	3	4	5	6	7	8	9	10	11	12	13	14
2	15	16	17	18	19	20	21	22	23	24	25	26	27	28
3	29	30	31	1	2	3	4	5	6	7	8	9	10	11
4	12	13	14	15	16	17	18	19	20	21	22	23	24	25
5	26	27	28	29	30	31	1	2	3	4	5	6	7	8
6	9	10	11	12	13	14	15	16	17	18	19	20	21	22
7	23	24	25	26	27	28	29	30	31	1	2	3	4	5
8	6	7	8	9	10	11	12	13	14	15	16	17	18	19
9	19	20	21	22	23	24	25	26	27	28	29	30	31	1
10	2	3	4	5	6	7	8	9	10	11	12	13	14	15
11	16	17	18	19	20	21	22	23	24	25	26	27	28	29
12	29	30	31	1	2	3	4	5	6	7	8	9	10	11
13	12	13	14	15	16	17	18	19	20	21	22	23	24	25
14	26	27	28	29	30	31	1	2	3	4	5	6	7	8
15	9	10	11	12	13	14	15	16	17	18	19	20	21	22
16	23	24	25	26	27	28	29	30	31	1	2	3	4	5
17	6	7	8	9	10	11	12	13	14	15	16	17	18	19
18	19	20	21	22										

Total Values

Fig.2 Dataset Collection Table - CIC-DDoS 2019

The NSL-KDD dataset is a widely used benchmark for evaluating network intrusion detection systems, providing labeled instances of normal and malicious network traffic. It includes network traffic features such as packet count, service error rate, and destination bytes. The dataset is a refined version of the original KDD Cup 1999 dataset, addressing some of its limitations by removing redundant records and balancing the distribution of attack types [5].

		1997年12月										1998年1月										1998年2月										1998年3月										1998年4月										1998年5月										1998年6月										1998年7月										1998年8月										1998年9月										1998年10月										1998年11月										1998年12月										1999年1月										1999年2月										1999年3月										1999年4月										1999年5月										1999年6月										1999年7月										1999年8月										1999年9月										1999年10月										1999年11月										1999年12月										2000年1月										2000年2月										2000年3月										2000年4月										2000年5月										2000年6月										2000年7月										2000年8月										2000年9月										2000年10月										2000年11月										2000年12月										2001年1月										2001年2月										2001年3月										2001年4月										2001年5月										2001年6月										2001年7月										2001年8月										2001年9月										2001年10月										2001年11月										2001年12月										2002年1月										2002年2月										2002年3月										2002年4月										2002年5月										2002年6月										2002年7月										2002年8月										2002年9月										2002年10月										2002年11月										2002年12月										2003年1月										2003年2月										2003年3月										2003年4月										2003年5月										2003年6月										2003年7月										2003年8月										2003年9月										2003年10月										2003年11月										2003年12月										2004年1月										2004年2月										2004年3月										2004年4月										2004年5月										2004年6月										2004年7月										2004年8月										2004年9月										2004年10月										2004年11月										2004年12月										2005年1月										2005年2月										2005年3月										2005年4月										2005年5月										2005年6月										2005年7月										2005年8月										2005年9月										2005年10月										2005年11月										2005年12月										2006年1月										2006年2月										2006年3月										2006年4月										2006年5月										2006年6月										2006年7月										2006年8月										2006年9月										2006年10月										2006年11月										2006年12月										2007年1月										2007年2月										2007年3月										2007年4月										2007年5月										2007年6月										2007年7月										2007年8月										2007年9月										2007年10月										2007年11月										2007年12月										2008年1月										2008年2月										2008年3月										2008年4月										2008年5月										2008年6月										2008年7月										2008年8月										2008年9月										2008年10月										2008年11月										2008年12月										2009年1月										2009年2月										2009年3月										2009年4月										2009年5月										2009年6月										2009年7月										2009年8月										2009年9月										2009年10月										2009年11月										2009年12月										2010年1月										2010年2月										2010年3月										2010年4月										2010年5月										2010年6月										2010年7月										2010年8月										2010年9月										2010年10月										2010年11月										2010年12月										2011年1月										2011年2月										2011年3月										2011年4月										2011年5月										2011年6月										2011年7月										2011年8月										2011年9月										2011年10月										2011年11月										2011年12月										2012年1月										2012年2月										2012年3月										2012年4月										2012年5月										2012年6月										2012年7月										2012年8月										2012年9月										2012年10月										2012年11月										2012年12月										2013年1月										2013年2月										2013年3月										2013年4月										2013年5月										2013年6月										2013年7月										2013年8月										2013年9月										2013年10月										2013年11月										2013年12月										2014年1月										2014年2月										2014年3月										2014年4月										2014年5月										2014年6月										2014年7月										2014年8月										2014年9月										2014年10月										2014年11月										2014年12月										2015年1月										2015年2月										2015年3月										2015年4月										2015年5月										2015年6月										2015年7月										2015年8月										2015年9月										2015年10月										2015年11月										2015年12月										2016年1月										2016年2月										2016年3月										2016年4月										2016年5月										2016年6月										2016年7月										2016年8月										2016年9月										2016年10月										2016年11月										2016年12月										2017年1月										2017年2月										2017年3月										2017年4月										2017年5月										2017年6月										2017年7月										2017年8月										2017年9月										2017年10月										2017年11月										2017年12月										2018年1月										2018年2月										2018年3月										2018年4月										2018年5月										2018年6月										2018年7月										2018年8月										2018年9月										2018年10月										2018年11月										2018年12月										2019年1月										2019年2月										2019年3月										2019年4月										2019年5月										2019年6月										2019年7月										2019年8月										2019年9月										2019年10月										2019年11月										2019年12月										2020年1月										2020年2月										2020年3月										2020年4月										2020年5月										2020年6月										2020年7月										2020年8月										2020年9月										2020年10月										2020年11月										2020年12月										2021年1月										2021年2月										2021年3月										2021年4月										2021年5月										2021年6月										2021年7月										2021年8月										2021年9月										2021年10月										2021年11月										2021年12月										2022年1月										2022年2月										2022年3月										2022年4月										2022年5月										2022年6月										2022年7月										2022年8月										2022年9月										2022年10月										2022年11月										2022年12月										2023年1月										2023年2月										2023年3月										2023年4月										2023年5月										2023年6月										2023年7月										2023年8月										2023年9月										2023年10月										2023年11月										2023年12月										2024年1月										2024年2月										2024年3月										2024年4月										2024年5月										2024年6月										2024年7月										2024年8月										2024年9月										2024年10月										2024年11月										2024年12月										2025年1月										2025年2月										2025年3月										2025年4月										2025年5月										2025年6月										2025年7月										2025年8月										2025年9月										2025年10月										2025年11月										2025年12月										2026年1月										2026年2月										2026年3月										2026年4月										2026年5月										2026年6月										2026年7月										2026年8月										2026年9月										2026年10月										2026年11月										2026年12月										2027年1月										2027年2月										2027年3月										2027年4月										2027年5月										2027年6月										2027年7月										2027年8月										2027年9月										2027年10月										2027年11月										2027年12月										2028年1月										2028年2月										2028年3月										2028年4月										2028年5月										2028年6月										2028年7月										2028年8月										2028年9月										2028年10月										2028年11月										2028年12月										2029年1月										2029年2月										2029年3月										2029年4月										2029年5月										2029年6月										2029年7月										2029年8月										2029年9月										2029年10月										2029年11月										2029年12月										2030年1月										2030年2月										2030年3月										2030年4月										2030年5月										2030年6月										2030年7月										2030年8月										2030年9月										2030年10月										2030年11月										2030年12月										2031年1月										2031年2月										2031年3月										2031年4月										2031年5月										2031年6月										2031年7月										2031年8月										2031年9月										2031年10月										2031年11月										2031年12月										2032年1月										2032年2月										2032年3月										2032年4月										2032年5月										2032年6月										2032年7月										2032年8月										2032年9月										2032年10月										2032年11月										2032年12月										2033年1月										2033年2月										2033年3月										2033年4月										2033年5月										2033年6月										2033年7月										2033年8月										2033年9月										2033年10月										2033年11月										2033年12月										2034年1月										2034年2月										2034年3月										2034年4月										2034年5月										2034年6月										2034年7月										2034年8月										2034年9月										2034年10月										2034年11月										2034年12月										2035年1月										2035年2月										2035年3月										2035年4月										2035年5月										2035年6月										2035年7月										2035年8月										2035年9月										2035年10月										2035年11月										2035年12月										2036年1月										2036年2月										2036年3月										2036年4月										2036年5月										2036年6月										2036年7月										2036年8月										2036年9月										2036年10月										2036年11月										2036年12月										2037年1月										2037年2月										2037年3月										2037年4月										2037年5月										2037年6月										2037年7月										2037年8月										2037年9月										2037年10月										2037年11月										2037年12月										2038年1月										2038年2月										2038年3月										2038年4月										2038年5月										2038年6月										2038年7月										2038年8月										2038年9月										2038年10月										2038年11月										2038年12月										2039年1月										2039年2月										2039年3月										2039年4月										2039年5月										2039年6月										2039年7月										2039年8月										2039年9月										2039年10月										2039年11月										2039年12月										2040年1月										2040年2月										2040年3月										2040年4月										2040年5月										2040年6月										2040年7月										2040年8月										2040年9月										2040年10月										2040年11月										2040年12月										2041年1月										2041年2月										2041年3月										2041年4月										2041年5月										2041年6月										2041年7月										2041年8月										2041年9月										2041年10月										2041年11月										2041年12月										2042年1月										2042年2月										2042年3月										2042年4月										2042年5月										2042年6月										2042年7月										2042年8月										2042年9月										2042年10月										2042年11月										2042年12月										2043年1月										2043年2月										2043年3月										2043年4月										2043年5月										2043年6月										2043年7月										2043年8月										2043年9月										2043年10月										2043年11月										2043年12月										2044年1月										2044年2月										2044年3月										2044年4月										2044年5月										2044年6月										2044年7月										2044年8月										2044年9月										2044年10月										2044年11月										2044年12月										2045年1月										2045年2月										2045年3月										2045年4月										2045年5月										2045年6月										2045年7月										2045年8月										2045年9月										2045年10月										2045年11月										2045年12月										2046年1月										2046年2月										2046年3月										2046年4月										2046年5月										2046年6月										2046年7月										2046年8月										2046年9月										2046年10月										2046年11月										2046年12月										2047年1月										2047年2月										2047年3月										2047年4月										2047年5月										2047年6月										2047年7月										2047年8月										2047年9月										2047年10月										2047年11月										2047年12月										2048年1月										2048年2月										2048年3月										2048年4月										2048年5月										2048年6月										2048年7月										2048年8月										2048年9月										2048年10月										2048年11月										2048年12月										2049年1月										2049年2月										2049年3月										2049年4月										2049年5月										2049年6月										2049年7月										2049年8月										2049年9月										2049年10月										2049年11月										2049年12月										2050年1月										2050年2月										2050年3月										2050年4月										2050年5月										2050年6月										2050年7月										2050年8月										2050年9月										2050年10月										2050年11月										2050年12月										2051年1月										2051年2月										2051年3月										2051年4月										2051年5月										2051年6月										2051年7月										2051年8月										2051年9月										2051年10月										2051年11月										2051年12月										2052年1月										2052年2月										2052年3月										2052年4月										2052年5月										2052年6月										2052年7月										2052年8月										2052年9月										2052年10月										2052年11月										2052年12月										2053年1月										2053年2月										2053年3月										2053年4月										2053年5月										2053年6月										2053年7月										2053年8月										2053年9月										2053年10月										2053年11月										2053年12月										2054年1月										2054年2月										2054年3月										2054年4月										2054年5月										2054年6月										2054年7月										2054年8月										2054年9月										2054年10月										2054年11月										2054年12月										2055年1月										2055年2月										2055年3月										2055年4月										2055年5月										2055年6月										2055年7月										2055年8月										2055年9月										2055年10月										2055年11月										2055年12月										2056年1月										2056年2月										2056年3月										2056年4月										2056年5月										2056年6月										2056年7月										2056年8月										2056年9月										2056年10月										2056年11月										2056年12月										2057年1月										2057年2月										2057年3月										2057年4月										2057年5月										2057年6月										2057年7月										2057年8月										2057年9月										2057年10月										2057年11月										2057年12月										2058年1月										2058年2月										2058年3月										2058年4月										2058年5月										2058年6月										2058年7月										2058年8月										2058年9月										2058年10月										2058年11月										2058年12月										2059年1月										2059年2月										2059年3月										2059年4月										2059年5月										2059年6月										2059年7月										2059年8月										2059年9月										2059年10月										2059年11月										2059年12月										2060年1月										2060年2月										2060年3月										2060年4月										2060年5月										2060年6月										2060年7月										2060年8月										2060年9月										2060年10月										2060年11月										2060年12月										2061年1月										2061年2月										2061年3月										2061年4月										2									
--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	----------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---------	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--

Fig.3 Dataset Collection Table – NSL-KDD

The UNSW-NB15 dataset is a comprehensive network intrusion detection dataset that includes both modern attack techniques and normal network traffic. It features attributes such as flow duration, bytes, rates, and connection state. The dataset is designed to provide realistic and diverse network traffic scenarios, reflecting various cyber-attacks like DoS, probing, and backdoors. It serves as a valuable resource for training and testing machine learning models for intrusion detection [6].

[illegible]

Fig.4 Dataset Collection Table – UNSW-NB15

B) Pre-Processing

Pre-processing involves preparing the data for analysis by addressing issues such as duplicate entries, missing values, and irrelevant features. It ensures the data is clean, consistent, and ready for model training. This step plays a critical role in enhancing the accuracy and efficiency of machine learning models.

a) *Data Processing*: Data processing includes steps like removing duplicate entries to eliminate redundancy and ensure data consistency. Drop cleaning is performed to remove irrelevant features or data points that could introduce noise. These actions help improve model performance and prevent overfitting by ensuring that only relevant, high-quality data is used.

b) Data Visualization: Data visualization is the graphical representation of data, allowing for easy exploration of patterns, trends, and outliers. It includes techniques such as histograms, scatter plots, and heatmaps, helping to identify relationships between features and understand distributions. Visualization aids in the decision-making process, revealing insights that can guide further analysis and model development.

c) Label Encoding: Label encoding is used to convert categorical variables into numerical values, enabling machine learning models to interpret them. This transformation assigns a unique integer to each category, ensuring that algorithms can process the data. It is particularly useful for categorical features that have an inherent order or ranking, such as low, medium, or high.

d) Feature Selection: Feature selection using Select Percentile identifies and retains the most important features for predictive modeling. This technique evaluates the performance of each feature by selecting the top percentile based on statistical tests. By reducing the dimensionality of the data, it enhances model efficiency and accuracy, focusing on the most relevant variables for prediction.

C) Training and Testing:

In machine learning, the training and testing process involves splitting the dataset into two subsets: one for training the model and the other for evaluating its performance. An 80:20 split is commonly used, where 80% of the data is allocated for training the model, and 20% is reserved for testing. The training data helps the model learn patterns and relationships within the data, while the testing data is used to assess how well the model generalizes to unseen data. This approach ensures that the model's performance is reliable and prevents overfitting.

D) Algorithms

DNN is a type of artificial neural network with multiple hidden layers between the input and output layers. It models complex relationships in data by learning hierarchical representations. The purpose of using DNNs in intrusion detection is to recognize intricate patterns in large datasets. It is commonly used in cyberattack detection to classify various types of attacks effectively [12], [16].

CNN is a deep learning algorithm primarily used for image data but also applicable to time-series or sequence data. In the context of network intrusion detection, CNNs extract spatial hierarchies of features from raw data, allowing the model to capture localized patterns. This ability is especially useful for recognizing patterns of network attacks from raw traffic data [18], [19].

$$Z = (X * W) + B(1)$$

Where:

- Z = Output feature map
- X = Input matrix (network traffic data)
- W = Convolution filter (kernel)

The hybrid CNN+LSTM model combines CNN's ability to capture spatial features with LSTM's capacity to learn temporal dependencies. This combination is useful in intrusion detection for identifying both spatial features in network traffic and long-term dependencies that can signal malicious activity. It is particularly advantageous for sequential data like network traffic logs [19], [14].

The CNN+BiLSTM model leverages CNN for feature extraction and BiLSTM for learning from both past and future information. This bidirectional learning helps to improve the model's performance in understanding network attack patterns, as both temporal and spatial features are considered. This hybrid approach enhances classification accuracy in intrusion detection tasks [15], [17].

Integrating attention mechanisms with CNN+BiLSTM enables the model to focus on the most important features of the data while processing network traffic. This helps improve performance by highlighting critical attack patterns while minimizing irrelevant information. Attention mechanisms are used to optimize the learning process, making this model highly effective for network intrusion detection [16], [13].

$$at = \sum \text{jexp}(ej) \exp(et)(2)$$

Where:

- α_{tot} = Attention weight for time step t
- tet = Score for the attention mechanism (computed using a similarity function)
- $\sum \text{jexp}(ej)$ = Normalization term using softmax function

The CNN+GRU (Gated Recurrent Unit) with Attention model combines CNN for feature extraction, GRU for capturing temporal dependencies, and attention for prioritizing essential features. This model is well-suited for detecting and classifying various network attack types, leveraging both past context and key feature importance to improve prediction accuracy [18], [20].

Random Forest is an ensemble learning technique that combines multiple decision trees to improve classification accuracy. In intrusion detection, it is used to classify network attacks by learning patterns from feature sets such as flow duration and packet length. It effectively handles overfitting issues and works well with imbalanced datasets [14], [11].

$$y^* = N1i = 1 \sum N h_i(x)(3)$$

Where:

- $\hat{y}$ = Final predicted output (class label or regression value)
- NNN = Number of decision trees in the forest
- $h_i(x)$ = Prediction from the i -th decision tree
- xxx = Input feature vector

AdaBoost is a boosting algorithm that combines the predictions of weak learners (usually decision trees) to create a strong classifier. In the context of network intrusion detection, AdaBoost is used to enhance the accuracy of detecting malicious activities by focusing on difficult-to-classify instances. It helps improve the performance of traditional classifiers in handling imbalanced data [13], [12].

KNN is a non-parametric algorithm that classifies data points based on the majority vote of their nearest neighbors. It is used in network intrusion detection systems to classify new traffic data by comparing it with existing labeled data. KNN is simple to implement and effective in detecting known attack patterns when the dataset is properly preprocessed [19], [14].

A Decision Tree is a supervised learning algorithm that splits the data into branches based on feature values, with each leaf representing a class label. It is used in intrusion detection to classify network traffic by evaluating feature values such as packet size or flow duration. Decision trees are interpretable and easy to use, making them useful in cybersecurity applications [17], [15].

A Voting Classifier combines multiple models to enhance prediction accuracy by taking the majority vote of their predictions. In this approach, Bagging Decision Trees (BaggingDT) and Bagging Random Forests (BagRF) are used together to reduce variance and improve robustness. This ensemble method performs well in intrusion detection by combining strengths of different algorithms to achieve higher accuracy and reduce the impact of overfitting [14], [18].

IV. EXPERIMENTAL RESULTS

Accuracy: The accuracy of a test is its ability to differentiate the patient and healthy cases correctly. To estimate the accuracy of a test, we should calculate the proportion of true positive and true negative in all evaluated cases. Mathematically, this can be stated as:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (4)$$

Precision: Precision evaluates the fraction of correctly classified instances or samples among the ones classified as positives. Thus, the formula to calculate the precision is given by:

$$Precision = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \quad (5)$$

Recall: Recall is a metric in machine learning that measures the ability of a model to identify all relevant instances of a particular class. It is the ratio of correctly predicted positive observations to the total actual positives, providing insights into a model's completeness in capturing instances of a given class.

$$Recall = \frac{TP}{TP + FN} \quad (6)$$

F1-Score: F1 score is a machine learning evaluation metric that measures a model's accuracy. It combines the

precision and recall scores of a model. The accuracy metric computes how many times a model made a correct prediction across the entire dataset.

$$F1 \text{ Score} = 2 * \frac{Recall * Precision}{Recall + Precision} * 100 \quad (7)$$

Table.1 Performance Evaluation Metrics – CIC-DDoS 2019

ML Model	Accuracy	Precision	Recall	F1 Score
DNN	0.906	0.913	0.906	0.909
CNN	0.259	1.000	0.259	0.411
CNN + LSTM	0.878	0.904	0.878	0.886
CNN + BiLSTM	0.386	0.850	0.386	0.498
CNN + BiLSTM with Attention	0.910	0.938	0.910	0.918
CNN + GRU with Attention	0.903	0.947	0.903	0.919
Random Forest	0.958	0.961	0.958	0.959
AdaBoost	0.688	0.838	0.688	0.731
KNN	0.922	0.922	0.922	0.922
Decision Tree	0.958	0.961	0.958	0.958
Voting Classifier (BoostDT + BagRF)	0.981	0.982	0.981	0.982

Table.2 Performance Evaluation Metrics – NSL-KDD

ML Model	Accuracy	Precision	Recall	F1 Score
DNN	0.964	0.970	0.964	0.967
CNN	0.969	0.977	0.969	0.973
CNN + LSTM	0.925	0.932	0.925	0.927
CNN + BiLSTM	0.964	0.969	0.964	0.966
CNN + BiLSTM with Attention	0.948	0.949	0.948	0.948
CNN + GRU with Attention	0.947	0.949	0.947	0.947
Random Forest	0.997	0.997	0.997	0.997
AdaBoost	0.949	0.953	0.949	0.950
KNN	0.994	0.994	0.994	0.994
Decision Tree	0.997	0.997	0.997	0.997
Voting Classifier (BoostDT + BagRF)	0.999	0.999	0.999	0.999

Table.3 Performance Evaluation Metrics – UNSW-NB15

ML Model	Accuracy	Precision	Recall	F1 Score
DNN	0.814	0.879	0.814	0.841
CNN	0.450	1.000	0.450	0.621
CNN + LSTM	0.798	0.885	0.798	0.832
CNN + BiLSTM	0.450	1.000	0.450	0.621
CNN + BiLSTM with Attention	0.873	0.885	0.873	0.876

CNN + GRU with Attention	0.855	0.877	0.855	0.861
Random Forest	0.843	0.852	0.843	0.846
AdaBoost	0.772	0.830	0.772	0.793
KNN	0.791	0.792	0.791	0.791
Decision Tree	0.829	0.836	0.829	0.832
Voting Classifier (BoostDT + BagRF)	0.929	0.934	0.929	0.931

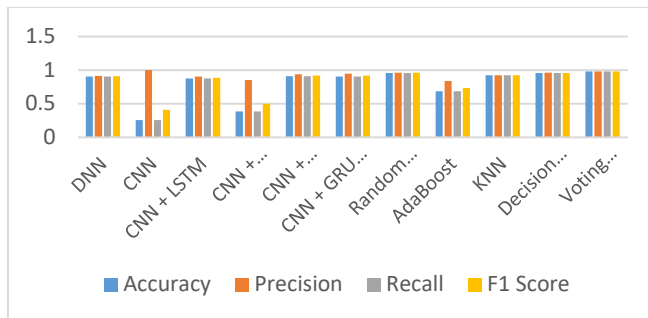


Fig.5 Comparison Graphs – CIC-DDoS 2019

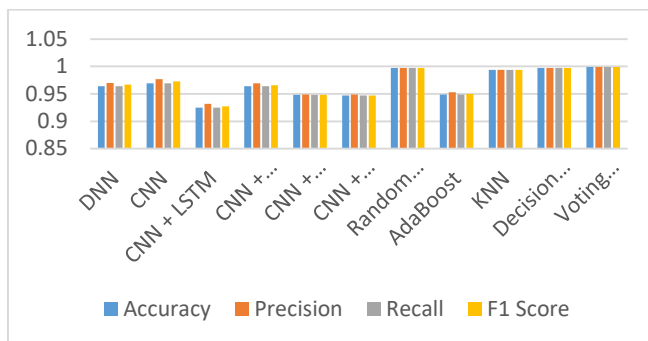


Fig.6 Comparison Graphs – NSL-KDD

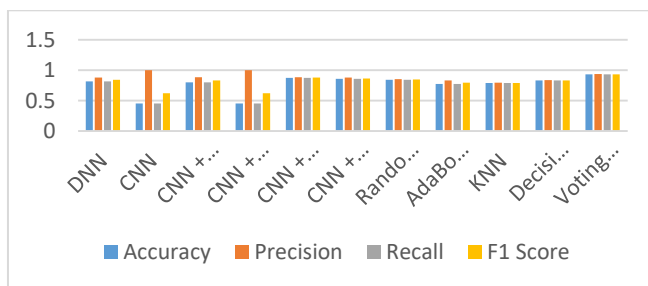


Fig.7 Comparison Graphs – UNSW-NB15

Accuracy is represented in light blue, precision in orange, recall in grey, and F1-Score in light yellow, Graph (1 to 3). In comparison to the other models, the Voting Classifier shows superior performance across all metrics, achieving the highest values. The graphs above visually illustrate these findings.

V. CONCLUSION

In conclusion, the proposed network intrusion detection system effectively integrates deep learning and ensemble methods to enhance cybersecurity by identifying malicious

activities with high accuracy. Through comprehensive evaluation on NSL-KDD, UNSW-NB15, and CIC-DDoS 2019 datasets, the system demonstrates its robustness in detecting diverse attack patterns. Among the implemented models, the Voting Classifier (BaggingDT + BagRF) achieves the highest performance, attaining 99.9% accuracy on the NSL-KDD dataset, 92.9% accuracy on the UNSW-NB15 dataset, and 98.1% accuracy on the CIC-DDoS 2019 dataset. The superior performance of the Voting Classifier highlights the effectiveness of ensemble learning in capturing complex network behaviors and mitigating overfitting issues, making it a highly reliable solution for intrusion detection. The integration of convolutional and recurrent architectures, along with attention mechanisms, improves feature extraction and sequence learning, contributing to enhanced detection rates. The results confirm that the hybrid approach significantly strengthens anomaly detection in network security, providing an efficient and scalable solution for real-time intrusion detection across diverse network environments.

Future Work

Future research can explore unsupervised and semi-supervised learning techniques to enhance the detection of unknown and evolving cyber threats. Generative models, self-supervised learning, and contrastive learning can improve feature extraction for anomaly detection. Additionally, integrating federated learning can enhance privacy-preserving intrusion detection across distributed networks. Real-time adaptability using reinforcement learning and graph-based models can further strengthen threat detection. Expanding datasets with adversarial attack scenarios will also improve model robustness against sophisticated cyber threats.

REFERENCES

- [1] H. Peng, C. Wu, and Y. Xiao, "CBF-IDS: Addressing class imbalance using CNN-BiLSTM with focal loss in network intrusion detection system," *Appl. Sci.*, vol. 13, no. 21, p. 11629, Oct. 2023.
- [2] J. Zhang, X. Zhang, Z. Liu, F. Fu, Y. Jiao, and F. Xu, "A network intrusion detection model based on BiLSTM with multi-head attention mechanism," *Electronics*, vol. 12, no. 19, p. 4170, Oct. 2023.
- [3] K. Ren, S. Yuan, C. Zhang, Y. Shi, and Z. Huang, "CANET: A hierarchical CNN-attention model for network intrusion detection," *Comput. Commun.*, vol. 205, pp. 170–181, May 2023.
- [4] A. Halbouni, T. S. Gunawan, M. H. Habaebi, M. Halbouni, M. Kartiwi, and R. Ahmad, "CNN-LSTM: Hybrid deep neural network for network intrusion detection system," *IEEE Access*, vol. 10, pp. 99837–99849, 2022.
- [5] S. S. Khan and A. B. Mailewa, "Detecting network transmission anomalies using autoencoders-SVM neural network on multi-class NSL-KDD dataset," in *Proc. IEEE 13th Annu. Comput. Commun. Workshop Conf. (CCWC)*, Mar. 2023, pp. 835–843.
- [6] V. Kumar, A. K. Das, and D. Sinha, "Statistical analysis of the UNSW-NB15 dataset for intrusion detection," in *Proc. Comput. Intell. Pattern Recognit. (CIPR)*. Singapore: Springer, 2020, pp. 279–294.
- [7] Rachid, B. E. N. (2024). NETWORK INTRUSION DETECTION SYSTEM USING HYBRID DEEP LEARNING APPROACHES IN SOFTWARE DEFINED NETWORKING (Doctoral dissertation, ANKARA UNIVERSITY).
- [8] Jihado, A. A., & Girsang, A. S. (2024). Hybrid Deep Learning Network Intrusion Detection System Based on Convolutional Neural

- Network and Bidirectional Long Short-Term Memory. *Journal of Advances in Information Technology*, 15(2), 219-232.
- [9] Alalmaie, A., Nanda, P., & He, X. Zt-Nids: Zero Trust-Network Intrusion Detection System Validation Based on Attack Simulations. Available at SSRN 4762072.
- [10] Girubagari, N., Ravi, T. N., & Arokiaraj, S. P. (2024). Hybrid Intelligent Anomaly Detection System Using Attention based Deep Learning Approach for Cyber Attacks Prevention. *Indian Journal Of Science And Technology*, 17(38).
- [11] Alam, K., Monir, M. F., Hassan, Z., & Habib, M. T. (2024, October). Optimizing IoT Network Intrusion Detection: A Deep Learning Approach. In 2024 7th Conference on Cloud and Internet of Things (CIoT) (pp. 1-5). IEEE.
- [12] Kalaiselvi, B., Gogul, S., Siva, S., Dhineshkumar, R., & Logeshwaran, M. (2024, April). Enhanced Intrusion Detection System Using Hybrid Harris Hawks Inspired Feature Selection For Efficient Network Intrusion Detection. In 2024 International Conference on Computing and Data Science (ICCDs) (pp. 1-6). IEEE.
- [13] Wu, J., Fu, Q., & Wang, L. (2024). MARNet: An efficient two-stage intrusion detection model based on deep learning. *IEEE Access*.
- [14] Alotaibi, M., Alohal, M. A., Mahmood, K., Alhashmi, A. A., Alqurni, J. S., Alotaibi, S. R., ... & Issaoui, I. (2024). Integrating Two-Tier Optimization Algorithm with Convolutional Bi-LSTM Model for Robust Anomaly Detection in Autonomous Vehicles. *IEEE Access*.
- [15] Alqahtani, H., & Kumar, G. (2024). Deep learning-based intrusion detection system for in-vehicle networks with knowledge graph and statistical methods. *International Journal of Machine Learning and Cybernetics*, 1-17.
- [16] Imrana, Y., Xiang, Y., Ali, L., Noor, A., Sarpong, K., & Abdullah, M. A. (2024). CNN-GRU-FF: a double-layer feature fusion-based network intrusion detection system using convolutional neural network and gated recurrent units. *Complex & Intelligent Systems*, 1-18.
- [17] Ren, K., Yuan, S., Zhang, C., Shi, Y., & Huang, Z. (2023). CANET: A hierarchical cnn-attention model for network intrusion detection. *Computer Communications*, 205, 170-181.
- [18] Sakthi, K., & Nirmal Kumar, P. (2023). A novel attention-based feature learning and optimal deep learning approach for network intrusion detection. *Journal of Intelligent & Fuzzy Systems*, 45(3), 5123-5140.
- [19] Du, J., Yang, K., Hu, Y., & Jiang, L. (2023). NIDS-CNNLSTM: Network intrusion detection classification model based on deep learning. *IEEE Access*, 11, 24808-24821.
- [20] Deore, B., & Bhosale, S. (2022). Hybrid optimization enabled robust CNN-LSTM technique for network intrusion detection. *Ieee Access*, 10, 65611-65622.