MMM-DADCL-Net: An Integrated Multi-Model Deep Attention Network based on Machine Learning and Recommendation System

Anthony. Kwame Ardiabah[1], Dr. Jaimeel Shah[2], Dr. Amit Ganatra[3], Naval captain kwame Osei[4]

[1]*Ph.D Researcher, Faculty of Engineering and Technology, Parul University, Gujarat, India*
[2]*Associate Professor, Computer Science and Engineering, Parul University, Gujarat, India*
[3]*Provost, Parul University, Gujarat, India*
[4]*Electrical Engineer, Naval college of engineering, Gujarat, India*
Email: antonykwameardiabah63@gmail.com

Abstract: The Gulf of Guinea has been in the center of attention since 2010, when the International Maritime Organization labeled it one of the most dangerous areas due to the persistence of piracy and armed robbery against ships. This study presented a novel way to identifying attacks in the GoG. Initially, the dataset is pre-processed using standard methodologies, and features are retrieved using statistical methods. The retrieved characteristics are used to choose features using a self-adaptive bacterial foraging optimization algorithm. Finally, classification is performed using a novel MMM-DADCL Net approach that classifies ship type, ship state, protection level, and attacks in four phases using the self-attention mechanism and CNN models such as Alex-Net, Google-Net, LeNet, VGG-19, LSTM, and Densely connected FCN. The proposed model achieved an Accuracy of 97.09% when implemented on a Python platform. The results demonstrate the proposed model's superior efficiency on comparisons with the current techniques.

Keywords: Gulf of Guinea, Bacterial Foraging Optimization Algorithm, Alex-Net, Google-Net, LeNet, VGG-19, LSTM, densely connected FCN.

1. Introduction

The maritime transportation system is responsible for ensuring safe transit in this huge area, as water covers approximately 70% of the earth's surface. Ship development is inextricably linked to global trends and the integration of cutting-edge Industry 4.0 technologies. To deal with the growing massive volume of shipping, increasing numbers of ports place significant emphasis on automation at ports [1-3].

The Gulf of Guinea (GoG) is a huge coastal region extending from Senegal to Angola, with a coastline of more than 6,000 kms [4]. It is a critical geopolitical chokepoint for shipping and maritime transit of oil derived from the Niger Delta, as well as the transportation of goods to and from other African and international locations. The GoG covers about eleven coastal countries in West and Central Africa, including Senegal, Ghana, and Angola [4]. Despite its importance as a resource provider and key international trade route, the GoG has recently gained notoriety for a variety of illicit operations. These activities include illicit piracy, illegal fishing, human trafficking, drug and weapons smuggling and others [5]. The region has seen an increase in instability, with incidences of robbery in local waterways and deep-sea piracy becoming more regular [6]. Furthermore, there is rising concern about the possible confluence of cybercrime and maritime crime in the Gulf of Guinea, which will provide considerable issues for law enforcement, shipping companies, and the maritime industry as a whole. The increased use of technology in marine operations, along with the high incidence of organized criminal activity at sea, suggests a greater danger of cyber-enabled maritime crimes in the region [7].

Research into marine mishaps is critical for improving maritime safety management and lowering the frequency of such incidents [8]. Ship type is an important aspect in maritime traffic since it influences maneuverability and physical qualities such as length, width, draft, and handling capabilities. Specialized vessel types such as passenger ships, oil tankers, and LNG carriers have unique operational needs for safe navigation. Recognizing the presence and kind of ships is critical for marine surveillance, security, traffic control, and preventing intrusions in sensitive regions and timeframes [9]. Accurate ship identification not only improves navigation safety but also vessel traffic services (VTS) by allowing for speedy identification of various ship types, which improves port navigation and rescue capabilities [10].

Machine learning (ML) is increasingly used in a variety of fields, including reliability engineering and safety [11]. Deep learning (DL), a subfield of ML, has been an important factor of technological growth [12-13]. AI models are developed more precisely by combining training data from numerous ships. Deep neural network classification is widely used in AIS trajectory analysis, particularly for future trajectory prediction and ship type classification [14]. As a result, for autonomous ships, detailed risk assessment throughout the initial design stages is critical. This approach must take into account safety, security, and cybersecurity, as well as potential causes and effects across the many operating phases [15]. The goal of this study is to appraise maritime security challenges in Gulf of Guinea which is done by the following process. The main contribution of the research are as follows,

- Pre-processing and Feature Extraction is done based on the standard approaches.
- Features are selected by a novel Lyre fish Optimization Algorithm.
- Classification is done by MMM-DADCL Net approach which identifies the results using four phases.

The other part of the article incorporates a literature review in Section 2, an in-depth description of the recommended technique in Section 3, Results and findings in Section 4, and Conclusions in Section 5.

2. RELATED WORK

Liu et.al [16] analyzed current ship anomaly detection approaches, revealing their shortcomings, and offers a new strategy based on the GRU neural network. It incorporated an attention mechanism and optimizes using a genetic algorithm. The simulation results compare the performance of the GRU model before and after optimization, indicating that the optimized model has higher accuracy and convergence speed in spotting aberrant ship behavior.

Rong et.al [17] introduced a unique method for detecting and classifying aberrant ship behavior based on AIS data is presented. The process entailed studying ship trajectories, extracting motion parameters, and using a sliding window technique. Abnormal behavior is characterized by features such as speed deviation and drift angle, which are then clustered into comparable patterns. These features are used to train a random forest classification model. The approach, which was tested off the coast of Portugal, successfully recognized and classified anomalous actions.

Bowen et.al [18] employed a deep learning for a revolutionary ship detection and tracking method. This framework uses a deep residual network and cross-layer jump connection policy to extract sophisticated ship features. This increases classification accuracy and object identification performance. The suggested ship recognition and tracking system outperformed state-of-the-art algorithms across multiple ship video datasets.

Lee et.al [19] provided an approach for ship awareness based on camera images that included ship detection, localization, and tracking. A deep learning-based model trained on a virtual picture dataset detects ships, which are then localized in spatial coordinates. The tracking is performed using an extended Kalman filter to estimate the target ships' position, course over ground, and speed over ground. In addition, a DL technique is presented to calculate the ship's heading in the image. The approach is proven with genuine ship videos and performs well.
Yang et.al [20] proposed a novel architecture for ship categorization based on AIS data. The ship motion and trajectory shape elements are retrieved, resulting in trajectory photos for experimentation. The CNN is fine-tuned by changes to network depth, composition, and characteristics such as batch size. A high-performing network structure is chosen depending on a variety of criteria. A ship categorization model is developed and verified using self-created datasets and ship division methods. The efficiency of classification improves significantly as the model's structure and parameters are refined.
Escorcia-Gutierrez et.al [21] offered OMRCNN-SHD, an efficient technique for detecting small ships in autonomous shipping. Data augmentation uses limited real-world samples to improve detection accuracy. For ship detection, the method employs Mask RCNN with Squeeze- Net model, which has been optimized using the Adagrad optimizer. The Colliding Body's Optimization, along with the weighted normalized learning correctly categorizes recognized ships, resulting in better reports.
Baeg et al. [22] established a ship-type categorization scheme based on ship trajectory data, extracting elements such as behavioral, geographic, and ship appearance factors. They utilize machine learning algorithms to classify ships into four categories: fishing, passenger, tanker, and cargo. The findings suggest that the proposed characteristics can accurately depict ship trajectories for categorization. The Random Forest method outperforms the others, reaching 84.05% accuracy.
Yildirim et.al [23] created a technique that integrates frequency domain parameters with an image-based DL system. The method determines the fast Fourier transform of ship audio files and generates frequency magnitude graphs in a series of visuals. The images are passed into ResNet50 system for classification. A public dataset encompassing different ship types is used to assess the proposed method's effectiveness as a result of increased accuracy.
Detecting and classifying aberrant ship behavior is a major task for the maritime industry to assure marine safety, security, and operational efficiency. Existing methods vary in effectiveness, prompting the creation of more precise and robust approaches. This includes improving anomaly detection accuracy, categorization, and efficiency using novel algorithms, advanced machine learning approaches, and the integration of several data sources. Overcoming limitations of present methods, such as data quality, scalability, and real-time processing, is vital, as is contributing to Security, and operational excellence through improved categorization approaches.

3. PROPOSED MODEL

For detecting attacks in the GOG, a novel approach is utilized which undergoes the following process. Initially, the dataset is pre-processed using standard methodologies, and features are retrieved using statistical methods. The retrieved characteristics are used to choose features using a self-adaptive bacterial foraging optimization algorithm. Finally, classification is performed using a novel MMM-DADCL Net approach that classifies ship type, ship state, protection level, and attacks in four phases using the self-attention mechanism and CNN models such as Alex-Net, Google-Net, LeNet, VGG-19, LSTM, and Densely connected FCN. Figure 1 shows the Architecture of the suggested model.

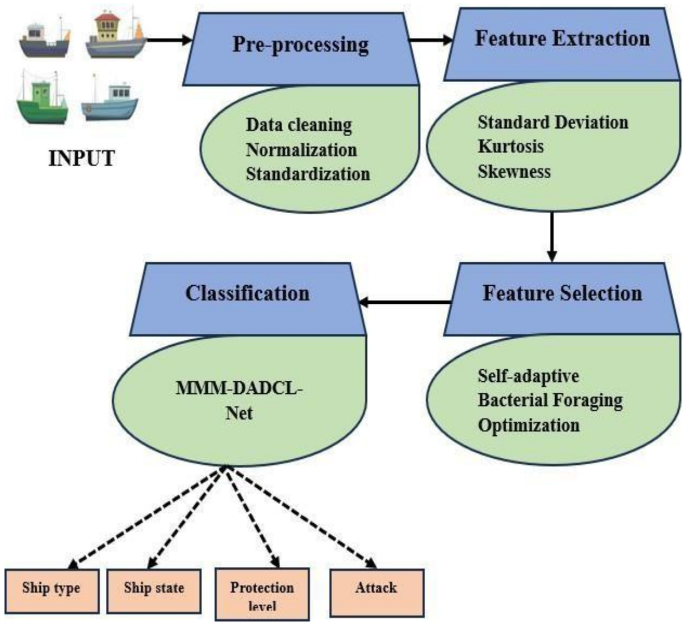

Figure 1: Proposed model

3.1. Pre processing

The initial step is the model is Pre-processing. It is a significant step for transforming the raw dataset into a most suitable format. Consider a dataset as $x = DS \epsilon DS1, DS2, \ldots DSn$. In this work, the data is pre-processed by data cleaning, standardization and normalization techniques.

3.1.1. Data cleaning

In image processing, data cleaning is critical for refining data before analysis. This method involves deleting null values and duplicates. To handle null values, remove any extraneous rows or columns. Duplicates are discovered and removed to guarantee that each unique data entry appears only once, hence increasing accuracy.

3.1.2. Standardization

It is the process of rescaling the features. The μ and σ will rescale the data, resultant features have zero μ and a unit σ.

$$x'(x) = \frac{x-\mu}{\sigma} \quad (1)$$

Where μ, σ are Mean and Standard deviation

3.1.3. Normalization

It is the process of rescaling values between [0, 1] or [-1,1]. The use of a normalization method will improve analysis for some models.

$$x'(x) = \frac{x-\min(x)}{m(x)-\min(x)} \quad (2)$$

3.2. Feature Extraction

It is the procedure of extracting a set of characteristics from the pre-processed results. This allows us to obtain features that will be useful in detecting anomalies. In this work, the characteristics are extracted using the statistical features described in this section.

- Standard deviation

The standard deviation measures the spread or dispersion of a pixel range. It determines the variance of pixel ranges from the mean. The image's contrast and texture are evaluated using the standard deviation. Low values indicate a more uniform region, but high values reveal a wide variety of pixel intensities, which are typically associated with texture or edges.

$$Standard\ deviation = \sqrt{\frac{\sum_{i=1}^{N}(I_i - Mean)^2}{N}} \quad (3)$$

$$Mean = \frac{\sum_{i=1}^{N} I_i}{N} \quad (4)$$

Ii, N represents Intensity values of the pixel and Total pixels.

- Kurtosis

Kurtosis is a statistical approach used to obtain non-Gaussian values. It adjusts the estimated evaluation of a physical parameter using a Gaussian. Kurtosis measures the probability distribution curve of a real-valued random variable.

$$K = \frac{1}{N}\sum_{i=1}^{n}(x_i - \mu)^4 \quad (5)$$

- Skewness

Skewness is a statistical measure of asymmetry or lack of symmetry in a dataset in which the weights assigned to each data point vary depending on particular criteria. Skewness assesses the extent of a data distribution.

$$S_w = \frac{1}{N}\sum_{i=1}^{n}(x_i - \mu)^3 \quad (6)$$

3.3. Feature Selection

It is a popular and important process that seeks to find key aspects while also deleting redundant and irrelevant features. Thus, it entails selecting a subset of significant traits from the original collection, retaining the most informative ones while rejecting the less important ones. The characteristics are accurately taken by the self-adaptive Bacterial Foraging Optimization Algorithm (BFOA).

BFOA is a swarm intelligence search technique designed to address large-scale optimization events generally complicated dynamic, and have constraints. Its biomimetic premise is based on Escherichia coli's foraging process, which consists of three stages: chemotaxis, reproduction, and dispersal.

- Chemotaxis

During bacterial foraging, differentiating the environment is critical for effective food seeking and population growth. Bacteria commonly alter orientation by flipping to reconfigure their foraging pattern in regions where food is harmful. In contrast, in food-rich areas, they continue to swim in the same direction. The merging of swimming and flipping actions is known as the chemotaxis stage. To sustain population expansion, bacteria destroy inefficient foragers, reproduce those in food-rich locations, and enhance populations through replication phases.

- Reproduction

Throughout the chemotaxis process, multiple fitness values can be used to evaluate which locations are best suited for bacterial survival. Then, half of the bacteria in the area with low nutrition levels are destroyed, while the remaining half replicate themselves. As a result, the number of bacteria remains constant during the reproductive process. However, the algorithm's convergence rate can be accelerated following the reproduction procedure.

- Dispersal

Furthermore, in response to severe natural disasters, bacteria with lesser survival capabilities in the community die as a result of fast environmental changes or food scarcity. Meanwhile, bacteria with higher survival rates migrate through the population, a process known as dispersion. Throughout the foraging phase, the quality of the environment continues to influence bacteria's survival tactics. The fitness function F measures the quality of bacterial sites.

Assuming V is the original population of bacteria, spreading a single bacterium in the group is the best solution to the associated practical applications problem. The position data of a single bacterium, including an M-dimensional vector, can be written as follows,

$$Sa = [Sa_1, Sa_2, \ldots, Sa_M]; \; a = 1,2, \ldots, V \qquad (7)$$

Bacterial position after chemotactic, replication, and dispersion steps can be represented by (b, c, d) given as,

$$(b, c, d) = [Sa_1, Sa_2, \ldots, Sa_M] \qquad (8)$$

The optimal fitness for a given position can be expressed by (b, c, d) and the position update of chemotactic action is done by,

$$(b + 1, c, d) = Sa(b, c, d) + E(a).\emptyset(a) \qquad (9)$$

(a) is a measure of duration of steps while bacterial swimming which is >0, $\emptyset(a)$ is the random direction selected for rolling stated by,

$$\emptyset(a) = \frac{\Delta(a)}{\sqrt{\Delta T(a).\Delta(a)}} \qquad (10)$$

$\Delta(a)$ is randomly generated in range [-1,1].

During the reproductive stage of bacteria, the fitness function (b, c, d) ranks individuals based on their locations. Half of those with greater fitness undergo self-replication, while the other half are removed. As a result, the bacterial population includes individuals with doubled fitness. The system dynamically adjusts the reproduction rate to balance exploration and exploitation, with lower rates encouraging exploration and higher rates encouraging exploitation. This modification is based on the observed increase in fitness in the bacterial population. The reproduction rate throughout the reproductive stage determines the percentage of individuals who replicate themselves. Improvements in the fitness of the bacterial population between iterations permit changes to the reproduction rate Rr.

$$Rr = Initial\ Rr \times \frac{(1 + \Delta)}{Favg} \qquad (11)$$

$$Initial\ Rr = V \times Rr\% \qquad (12)$$

ΔFa improvement in fitness value for each iteration, $Favg$ is average fitness, $Rr\%$ is taken by user (if 30% is considered, then $Rr\% = 0.3$).

Bacterial dispersal can be simulated using the migration probability, Pmd. When this reaches migration probability threshold, a new individual is randomly created at any point in space, while the original disappears. This process produces results with varying positions and fitness levels, allowing some to explore new places and improve global search skills by swimming and flipping.

3.4. Classification

For a successful classification, a suitable classification system needs to be employed. A sufficient number of training samples and their representativeness are critical for the classifications. In this research, a novel approach called a multi-level-based classification is performed to improve accuracy of the proposed model.

In the three phases of classification, a variation of CNN, Alex-Net, Google-Net, LeNet, and VGG-19 are used, and the final classification is done with LSTM-Densely linked FCN. CNN is a neural system comprised of numerous filter stages and a classification phase. It was driven by the concept of the visual system and developed for image processing; it is still widely used in computer vision applications. It is intended to extract characteristics from inputs. The detailed explanation is given as,

a) Convolution layer

To generate the output features, this layer convolves the input local regions with filter kernels, which are then passed to the activation unit. Each filter employs the same kernel to extract local features from the input local region, a process known as weight-sharing. This operation is given as follows,

$$y = \sum_{i=1}^{d} (ki * Wc + bc); c = 1,2,3..m \quad (13)$$

Where ki is the input, Wc, bc are weights and bias of filter

Every convolutional block must have an activation function layer. It allows the network to describe the input signal in a nonlinear way. There are several activation functions which is described by,

$$ReLU = \{ \begin{matrix} 0; k \leq 0 \\ 1; k > 0 \end{matrix} \quad (14)$$

$$Tanh = \frac{e^{k} - e^{-k}}{e^{k} + e^{-k}} \quad (15)$$

b) Pooling layer

$$Sig = 1/1+e-k \quad (16)$$

Pooling layers are commonly used to minimize feature size and accelerate learning. The applies the local max operation on the input features with a specific kernel/pool size, resulting in location-invariant features.

$$Max\ po(kab) = maxl-1 \quad k_{a+p,b+q} (17)$$

kab represents the input feature maps at position (a, b), where a, b are row and column index, l is window size.

c) Fully connected layer

All outputs from the preceding layer are connected to all inputs of the FC layer, which predicts the label. This layer predicts the target class using activation functions like SoftMax.

$$z = k. W + b \quad (18)$$

$$SoftMax \frac{e^{z_i}}{\sum_{j=1}^{k} e^{z_j}} \quad (19)$$

k are the total input vectors.

Self-attention mechanisms (SAM) work in tandem or sequentially to acquire multiple components of input data. It calculates attention scores and generates weighted outputs based on the data it receives. The i/p sequence is divided into two halves: the query sequence (Q) and the key sequence. Q is multiplied by the weight matrix to provide a set of attention weights. To obtain the output values (V), the K is weighted with the attention weights and then multiplied by the weight matrix a second time.

A score = y. WQ1. (y. WK1)^T (20)

A weight = Softmax(A score) (21)

A output = A1 weight. (y. WV1) (22)

$WQ1, WK1, . WV1$are learnable weight matrices for Attention.

- Alex Net

The Alex Net model consists of five convolutional, three maximum pooling, and three FC layers. The Alex Net model receives an input and passes it through the convolution block before using the ReLU (Rectified Linear Unit) activation function to remove nonlinearities. The findings are then down-sampled using pooling layers to extract useful features. The feature vector is then flattened with the first FC layer, which removes 50% of the features, and the final FC layer produces 1000 outputs. The network's output is normalized to a probability distribution over the projected output class by using the SoftMax activation function or a normalized exponential function.

- Google-Net

Google-Net, a version of the Inception model, consisting of 22 layers with several stacked Inception modules, each followed by average pooling and fully connected layers. The module's architecture takes into account two CL with a patch size of 3x3 and two unique strides. To accelerate processing after convolutional layers, the Inception model was modified by dividing the 5 × 5 layers into several 3 × 3 and ReLU layers. The max-pooling methodology extracts critical characteristics, such as edges, while the average pooling method extracts features gradually. When one or more pooling layers are added in the CL-generated feature maps, computational difficulties are reduced. Finally, classification is finished using the SoftMax in the fully linked layer.

- LeNet

Along with the input and output layers, LeNet-5 includes three convolutional layers and two fully connected layers. In the convolutional layer, the core size is set to 5 × 5, whereas in the pooling layer it is set to 2 × 2. The fully connected layer minimizes the number of neurons, which reduces parameter training.

- VGG-19

VGG-19 is a deep CNN architecture with 19 layers. It consists of 16 convolutional layers and three fully connected. Each convolution layer detects patterns in the input image using a tiny window (3x3 pixels) and ReLU. The spatial size of the pooling layer is then lowered to prevent overfitting. The network's end contains three fully connected levels. They assess all of the characteristics extracted by the convolutional layers before making a final conclusion.

3.4.1. Multi-level classification

This approach involves four stages in the classification process as provided in Table 1 and the architecture of the multi-level classification is illustrated in Figure . The detailed explanation of each stage of the model is as follows.

Table 1: Multi-level Classification

Classification level	Classification type	Model
1	Ship type	Self-Attention based Alex-Google Net
2	Ship state	Self-Attention based Google- Le Net
3	Protection level	Self-Attention based Le-VGG-19 Net
4	Attack	LSTM-Densely connected FCN

- Stage 1

Phase 1 involves classifying the type of ship using an Alex-Google Net based on self- attention. The input data is first fed into Alex Net's convolutional layer, which extracts features from the input, in this suggested classification method. The next step is to pool the features in order to get an output by down sampling the features. Subsequently, self-attention techniques are utilized to enable weight sharing among various regions inside the feature map. Self-attention improves the model's representation of the input by enabling it to suppress unimportant elements and concentrate on significant ones. Another convolutional neural network architecture called Google Net receives the output of the self-attention mechanism. To extract hierarchical features, the feature map is subjected to further convolution and pooling processes within Google Net. Ultimately, a fully connected layer is used to classify the processed features.

- Stage 2

A Google-LeNet architecture based on Self-Attention in the second phase to forecast the ship's status. The features from the Google Net are fed to the to the self-attention followed by the LeNet which extract features and pooling processes takes place within LeNet and finally the fully connected layer classifies the ship status.

- Stage 3

The Self-Attention based Le-VGG-19 Net technique used to classify the ship's protection level. The output from the LeNEt model is given to the, self-attention techniques which are utilized to enable weight sharing among various regions inside the feature map. Another CNN variant, VGG-19, receives the output of the self-attention mechanism and extracts hierarchical features, by the convolution and pooling procedures within VGG-19. Finally, the processed features are classified using a fully connected layer.

- Stage 4

Finally, in phase 4, the attack is classified using an LSTM with a densely connected FCN. The majority of the LSTM neural network is made up of three gated units: the forget gate, the i/p gate, and the o/p gate. To preserve distant reliance on time series details and accomplish high-precision prediction, certain gated units learn and memorize sequence data. The input gate handles the majority of I/P data. The forget gate controls the present neuron's ability to remember past information. The o/p gate represents the neuron's output or projection. Assuming the i/p sequence is $x1$, $x2$,..,, the LSTM neuron parameter at time t can be determined as,

$$it = (Wi * [ht{-}1, xt] + bi) \quad (23)$$

$$ft = (Wf * [ht{-}1, xt] + bf) \quad (24)$$

$$ot = (Wo * [ct, ht{-}1, xt] + bo) \quad (25)$$

$$ct = ft * Ct{-}1 + it * tanh(Wc * [ht{-}1, xt]) \quad (26)$$

$$ht = ot * tanh(ct) \quad (27)$$

Where $[it, ot, ft\ ]$, $[Wi, Wo, Wf\ ]$, $[bi, bo, bf]$are input, output and forget gates, and its weights and bias respectively, $xt$is the i/p to the LSTM neuron at time t, $ht{-}1$ is the o/p state of the hidden layer at time t-1, W_c is the weight between the i/p and the cell unit, $h_t$represents the o/p of the hidden layer at time t and S is the sigmoid function. Finally, using the densely connected FCN, 6 fully connected layers are utilized by which the final classification can be dene which results in successful and unsuccessful outputs.

4. RESULTS AND DISCUSSIONS

Using the dataset, the results are assessed using evaluation parameters. The results are computed based on comparisons between the suggested technique and current methods Google-Net, Alex-Net, Vgg-19 and LSTM using the dataset [24-25] containing 38 variables by implementing in a python platform.

4.1. Dataset Description and Evaluation metrics

The dataset [24-25] includes csv file of the piracy dataset containing 38 variables by which the attacks are detected using the recommended model. The parameters used for the analysis are detailed below. Table 2 and 3 provides the results of the proposed and current methods with a learning rate of 70/30, 80/20 for training and testing respectively.

Table 2: Results for 70% Training and 30% Testing

Metric	Proposed	VGG-19	Google Net	Alex Net	LSTM
Accuracy	0.96259	0.94588	0.94828	0.92154	0.93856
Precision	0.96051	0.94828	0.94082	0.92296	0.93458
F-Score	0.96164	0.94804	0.94122	0.92319	0.93753
Specificity	0.96163	0.94683	0.94958	0.92226	0.93852

Sensitivity	0.96312	0.94714	0.9423	0.9225	0.93159
MCC	0.96632	0.94763	0.94066	0.92287	0.93854
FPR	0.04727	0.0598	0.0678	0.06509	0.07204
FNR	0.04778	0.0566	0.06542	0.06846	0.07684

Table 3: Results for 80% Training and 20% Testing

Metric	Proposed	VGG-19	Googlenet	Alexnet	LSTM
Accuracy	0.97099	0.95241	0.95775	0.93629	0.94584
Precision	0.97085	0.95292	0.95879	0.93703	0.94012
F-Score	0.97122	0.95319	0.95816	0.93721	0.94245
Specificity	0.97114	0.95226	0.95683	0.93685	0.94634
Sensitivity	0.97118	0.9525	0.95769	0.93703	0.94842
MCC	0.9712	0.95287	0.95763	0.93696	0.94612
FPR	0.03781	0.05172	0.05983	0.05655	0.06573
FNR	0.03433	0.04762	0.0566	0.04913	0.05174

4.1.1. Accuracy

Accuracy is a measure of the proportion of correctly categorized cases among all occurrences.

$$\boldsymbol{Accuracy} = \frac{\boldsymbol{Tp+Tn}}{\boldsymbol{Tp+Tn+Fp+Fn}} \quad (28)$$

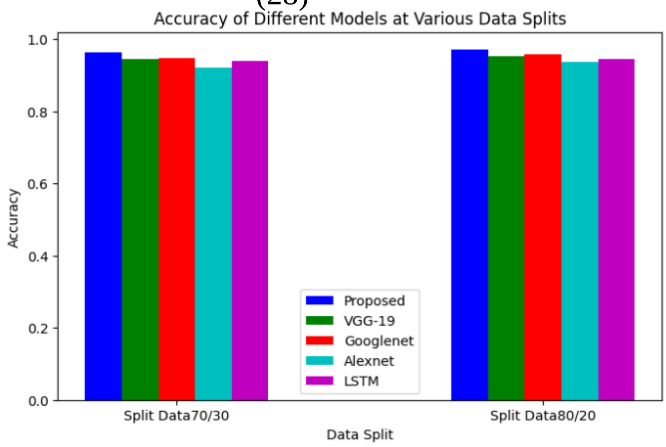

Figure 2: Examination of Accuracy for the models

Figure 2 depicts a graphical representation of the approaches with relation to accuracy. Tables 1 and 2 presents the numerical findings for the models. Based on the illustration, the proposed technique has an accuracy of about 97.09%.

4.1.2. Precision

Precision is a metric that measures how often a model correctly predicts the positive class.

$$\boldsymbol{Precision} = \frac{\boldsymbol{Tp}}{\boldsymbol{Tp+Fp}} \quad (29)$$

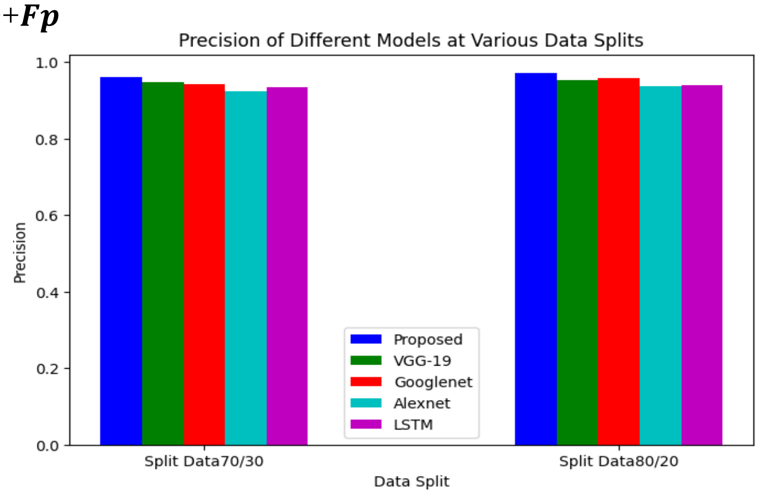

Figure 3: Analysis of models in terms of Precision

With a precision of 97.08%, the proposed approach has a low rate of false positives, indicating that when it predicts a positive occurrence, it is extremely likely to be true. Thus, the suggested is proven to be better considering Precision which is depicted from Figure 3.

4.1.3. Sensitivity

Sensitivity measures the proportion of true positives that are accurately recognized. It is given as follows

$$\boldsymbol{Sensitivity} = \frac{\boldsymbol{Tp}}{\boldsymbol{Tp+Fn}} \qquad (30)$$

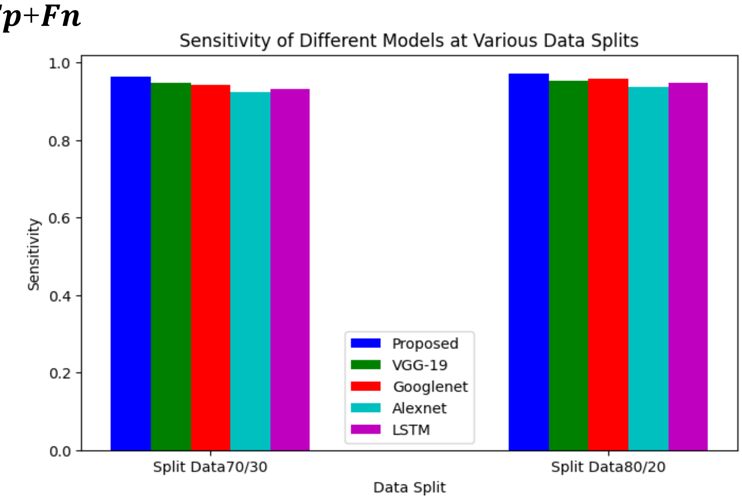

Figure 4: Assessment of Sensitivity

Figure 4 shows the graphical representation of Sensitivity. From the figure, it is seen that the proposed model attained a Sensitivity of 97.11% which shows the better performance of the model in comparison to the other approaches.

4.1.4. Specificity

Specificity is a measure of the percentage of true negatives that are correctly determined.

$$\boldsymbol{Specificity} = \frac{\boldsymbol{Tn}}{\boldsymbol{Tn+Fp}} \qquad (31)$$

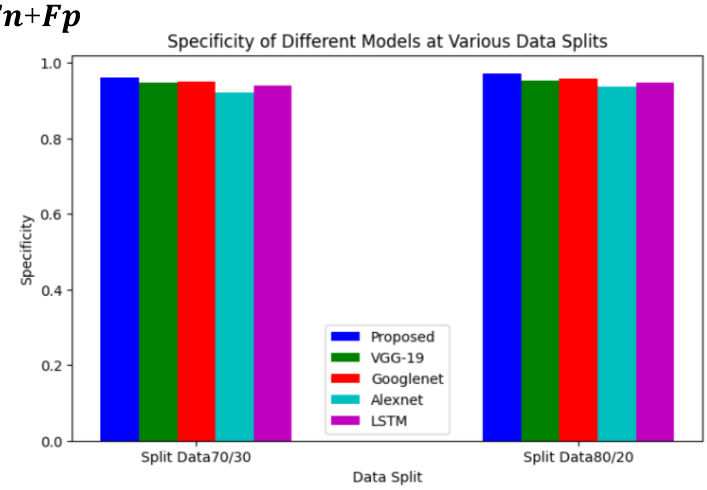

Figure 5: Evaluation of models with regard to Specificity

Figure 5depicts an illustration of Specificity for the proposed and existing models. The fig demonstrates that the proposed model achieved a Specificity of 97.11%, indicating the model executed better than other techniques.

4.1.5. F measure

A general score for performance evaluation, it is a combination statistic that combines Precision and recall.

$$\textbf{F measure} = \frac{2 * Precision * Recall}{Precision + Recall} \quad (32)$$

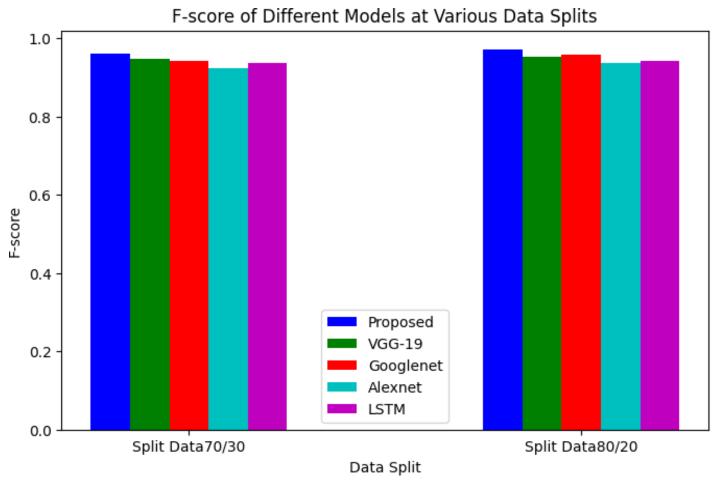

Figure 6: Comparison of models with regard to F-measure

The F-measure assessment of the current and proposed models is shown in Figure 6. The visual illustration makes it clear that the recommended model outperforms the proposed model in terms of F-measure as proven by its 97.08% F-measure

4.2.6. Mathew's Correlation Coefficient (MCC)

MCC assesses the level of correlation between predicted and real outcomes.

$$MCC = \frac{(Tp \times Tn) - (Fp \times Fn)}{\sqrt{(Tp+Fp)(Tp+Fn)(Tn+Fp)(Tn+Fn)}} \quad (33)$$

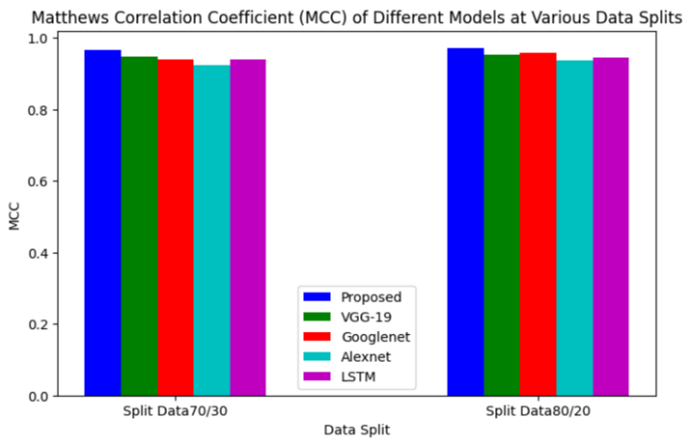

Figure 7: Comparison of models with regard to MCC

Figure 7 presents the MCC analysis of suggested and present models. It is evident from the graphical representation that the suggested model achieved a MCC of 97.12% which demonstrated that the recommended model performs better in terms of MCC.

4.2.7. FNR and FPR

FPR refers to statistics that are actually negative but are predicted to be positive, whereas

FNR refers to values that are actually positive but predicted to be negative.

$$FPR = \frac{Fp}{Fp+Tn} \quad (34)$$

$$FNR = \frac{Fn}{Fn+Tp} \quad (35)$$

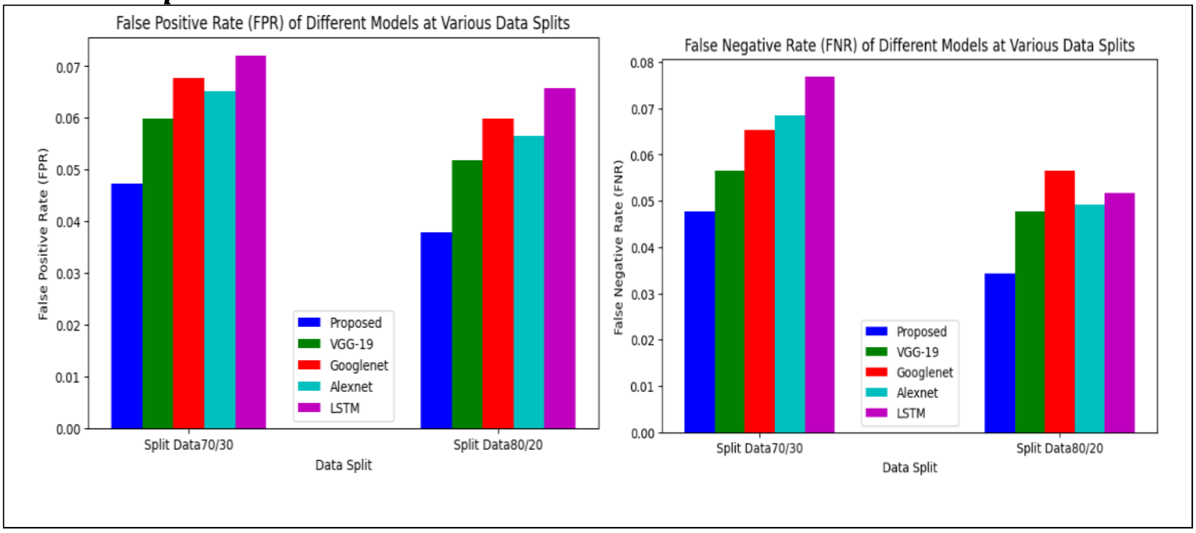

Figure 8: Analysis of FPR and FNR of the models

Figure 8 illustrates the shows the FPR and FNR for the models. From the graph, it is evident that the proposed model has a less FPR and FNR rates of about 3.7% and 3.4% which results in superior performance of the recommended model in comparisons to the other models.

5. CONCLUSION

Since 2010, the Gulf of Guinea has been identified as one of the hazardous shipping areas due to the continuing threat of piracy and armed robbery against ships, as noted by the International Maritime Organization. In response to this continuous concern, this study presented a novel strategy for detecting and responding to attacks in the Gulf of Guinea. The data is pre-processed by the standard methods. Subsequently, features are retrieved from the dataset using statistical methods. To improve feature selection, a self-adaptive bacterial foraging optimization technique is used. This method dynamically modifies its settings to efficiently choose the most useful features. Finally, classification is done by MMM-DADCL Net technique. This novel approach takes advantage of a self-attention mechanism and employs CNN models, including Alex-Net, Google-Net, LeNet, VGG-19, LSTM, and Densely connected FCN. Using the capability of these complex models, the classification process can precisely identify ship types, statuses, protection levels, and prospective threats in four distinct stages. Using Python platform, the suggested model achieves a high performance with Accuracy of 97.09%, Sensitivity of 97.11%, Specificity of 97.11%, Precision of 97.08%, F-measure of 97.12%, FPR and FNR of 3.7% and 3.4% and MCC of 97.12%, indicating its effectiveness in detecting and mitigating marine security concerns in the Gulf of Guinea. Comparative research with existing methodologies confirms the proposed model's higher efficiency, highlighting its potential to considerably improve marine safety and security in the region.

DECLARATIONS:
ETHICS APPROVAL
No ethics approval is required.
HUMAN AND ANIMAL ETHICS:
Not Applicable.
DECLARATION OF INTERESTS
The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

DATA AVAILABILITY STATEMENT
All the data is collected from the simulation reports of the software and tools used by the authors. Authors are working on implementing the same using real world data with appropriate permissions.
FUNDING
On Behalf of all authors the corresponding author states that they did not receive any funds for this project.
CONFLICTS OF INTEREST
The authors declare that we have no conflict of interest.
COMPETING INTERESTS
The authors declare that we have no competing interest.
INFORMED CONSENT
Not Applicable.
CONSENT FOR PUBLICATION
Not Applicable
ACKNOWLEDGMENTS
The authors would like to thank the Deanship of Parul University for supporting this work.

References

1. Gupta, Brij Bhooshan, Akshat Gaurav, Ching-Hsien Hsu, and Bo Jiao. "Identity-based authentication mechanism for secure information sharing in the maritime transport system." IEEE Transactions on Intelligent Transportation Systems 24, no. 2 (2021): 2422-2430.
2. Ma, Xindi, Qi Jiang, Mohammad Shojafar, Mamoun Alazab, Sachin Kumar, and Saru Kumari. "Disbezant: secure and robust federated learning against byzantine attack in iot-enabled mts." IEEE Transactions on Intelligent Transportation Systems 24, no. 2 (2022): 2492-2502.
3. Miao, Yang, Changan Li, Zhan Li, Yipeng Yang, and Xinghu Yu. "A novel algorithm of ship structure modeling and target identification based on point cloud for automation in bulk cargo terminals." Measurement and control 54, no. 3-4 (2021): 155-163.
4. Ogoun, Paul Mandela. "Maritime Security Challenge in Nigeria and the Gulf of Guinea." European Journal of Science, Innovation and Technology 2, no. 3 (2022): 69-74.
5. Abiodun, Temitope Francis, and M. Y. Dahiru. "Maritime insecurity in the Gulf of Guinea (GoG) and the quest for security intelligence deployment in combating the menace." International Journal of Advanced Academic Research 6, no. 4 (2020): 79-99.
6. Ofosu-Boateng, N., and Zhang Jiping. "A Pestle Analysis of Maritime Piracy and Maritime Security in the Gulf of Guinea." Advances in Social Sciences Research Journal 7, no. 1 (2020): 472-482.
7. Tachie-Menson, Elsie Amelia. "Investigating the Intersection of Maritime and Cyber Crime in the Gulf of Guinea." Scientia Militaria: South African Journal of Military Studies 51, no. 3 (2023): 89-112.
8. Zhang, Yang, Yujia Zhai, Shanshan Fu, Meiyu Shi, and Xuanyu Jiang. "Quantitative analysis of maritime piracy at global and regional scales to improve maritime security." Ocean & Coastal Management 248 (2024): 106968.
9. Pristrom, Sascha, Zaili Yang, Jin Wang, and Xinping Yan. "A novel flexible model for piracy and robbery assessment of merchant ship operations." Reliability Engineering & System Safety 155 (2016): 196-211.
10. Cao, Xiufeng, Shu Gao, Liangchen Chen, and Yan Wang. "Ship recognition method combined with image segmentation and deep learning feature extraction in video surveillance." Multimedia Tools and Applications 79 (2020): 9177-9192.

11. Majnarić, Darin, Sandi Baressi Šegota, Nikola Anđelić, and Jerolim Andrić. "Improvement of Machine Learning-Based Modelling of Container Ship's Main Particulars with Synthetic Data." Journal of Marine Science and Engineering 12, no. 2 (2024): 273.
12. Murray, Brian, and Lokukaluge Prasad Perera. "An AIS-based deep learning framework for regional ship behavior prediction." Reliability Engineering & System Safety 215 (2021): 107819.
13. Jiang, Yan, Xian-Rui Hou, Xue-Gang Wang, Zi-Hao Wang, Zhao-Long Yang, and Zao-Jian Zou. "Identification modeling and prediction of ship maneuvering motion based on LSTM deep neural network." Journal of Marine Science and Technology 27, no. 1 (2022): 125- 137.
14. Chen, Xiang, Yuanchang Liu, Kamalasudhan Achuthan, and Xinyu Zhang. "A ship movement classification based on Automatic Identification System (AIS) data using Convolutional Neural Network." Ocean Engineering 218 (2020): 108182.
15. Bolbot, Victor, Gerasimos Theotokatos, L. Andreas Wennersberg, Jerome Faivre, Dracos Vassalos, Evangelos Boulougouris, Ørnulf Jan Rødseth, Pål Andersen, Ann-Sofie Pauwelyn, and Antoon Van Coillie. "A novel risk assessment process: Application to an autonomous inland waterways ship." Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability 237, no. 2 (2023): 436-458.
16. Liu, Hongdan, Yan Liu, Bing Li, and Zhigang Qi. "Ship abnormal behavior detection method based on optimized GRU network." Journal of Marine Science and Engineering 10, no. 2 (2022): 249.
17. Rong, H., A. P. Teixeira, and C. Guedes Soares. "A framework for ship abnormal behaviour detection and classification using AIS data." Reliability Engineering & System Safety (2024): 110105.
18. Liu, Bowen, Sheng Zheng Wang, Z. X. Xie, JianSen Zhao, and MingFeng Li. "Ship recognition and tracking system for intelligent ship based on deep learning framework." TransNav: International Journal on Marine Navigation and Safety of Sea Transportation 13 (2019).
19. Lee, Won-Jae, Myung-Il Roh, Hye-Won Lee, Jisang Ha, Yeong-Min Cho, Sung-Jun Lee, and Nam-Sun Son. "Detection and tracking for the awareness of surroundings of a ship based on deep learning." Journal of Computational Design and Engineering 8, no. 5 (2021): 1407-1430.
20. Yang, Tianyu, Xin Wang, and Zhengjiang Liu. "Ship type recognition based on ship navigating trajectory and convolutional neural network." Journal of Marine Science and Engineering 10, no. 1 (2022): 84.
21. Escorcia-Gutierrez, José, Margarita Gamarra, Kelvin Beleño, Carlos Soto, and Romany F. Mansour. "Intelligent deep learning-enabled autonomous small ship detection and classification model." Computers and Electrical Engineering 100 (2022): 107871.
22. Baeg, Siyeol, and Tracy Hammond. "Ship Type Classification Based on The Ship Navigating Trajectory and Machine Learning." In Proceedings of the ACM IUI Workshops. 2023.
23. Yildirim, Mustafa Eren. "Ship type recognition using deep learning with FFT spectrums of audio signals." El-Cezeri 10, no. 1 (2023): 57-65.
24. https://springernature.figshare.com/articles/dataset/Processed_csv_file_of_the_piracy_dat aset/24119643
25. https://www.kaggle.com/code/teeyee314/classification-of-ship-images/input