Identifying Suspicious Fishing Activity based on AIS Disabling Events

Anmaya Agarwal (anmaya.agarwal02@nmims.edu.in)
MPSTME NMIMS
Jay Gala
MPSTME NMIMS
Saketh Mantha
MPSTME NMIMS
Yash Katariya
MPSTME NMIMS
Prashasti Kanikar
MPSTME NMIMS

Research Article

Keywords: fishing activity, AIS, global fishing watch, illegal, unreported, unregulated, class imbalance, AIS disabling, cost-sensitive learning, oversampling, undersampling, neural networks

Posted Date: April 10th, 2023

DOI: https://doi.org/10.21203/rs.3.rs-2782178/v1

Abstract

A staggering loss of US$10 billion to US$23 billion is incurred each year due to illegal, unreported, and unregulated (IUU) fishing activities along with a severe loss to biodiversity. The Automatic Identification System (AIS), is a tool used to track vessel activity and avoid collisions. It is now being used to detect IUU activities as well, but it has a major drawback as the AIS transponders could be disabled due to various reasons, either illegal or otherwise, hence reducing its effectiveness. According to Welch et al. (2022), more than 55,000 suspected intentional disabling events (> 4.9M hours) occurred between 2017 and 2019. Thus the need for much more sophisticated global surveillance has increased and algorithms to analyze such huge amounts of data are required. We present a machine learning solution based on historical data to detect vessels of interest using the AIS Disabling Events dataset obtained from the Global Fishing Watch combined with the Regional Fisheries Management Organizations (RFMOs) datasets containing details of vessels caught in IUU fishing activities previously within their respective regions. One of our best models is the XGBoost with cost-sensitive learning boasting a minority recall of 0.79 and a majority recall of 0.76.

I. Introduction

Supervision of marine vessels and fishing practices is becoming an increasing necessity. The amount of economic loss caused by Illegal, Unreported, and Unregulated (IUU) fishing activities annually ranges from USD 10 Billion to USD 23 Billion, along with severe effects on marine biodiversity. Over 26 Million tonnes of fish are lost annually due to IUU activity, according to the United Nations Food and Agriculture Organization (FAO)[2]. The Automatic Identification System (AIS) is a tool for detecting and monitoring vessel locations to prevent collisions at sea. It is now being used to detect vessels of interest and prevent IUU activities. However, a major drawback of this system is that the AIS transponders could be disabled due to various reasons, either illegal or otherwise, which is a common occurrence. According to Global Fishing Watch (GFW), which is an international non-profit organization, more than 4.6 million hours of AIS disabling events were recorded between 2017–2019. Thus it's becoming increasingly important to address this issue which has a direct implication on fisheries, government authorities, and consumers.

Due to a large amount of data being available, machine learning can be leveraged in this domain to take a step toward solving this problem. Welch et al. (2022) have published a dataset on the GFW website that has intentional AIS Disabling Events[4]. Then we needed to label the dataset to make it suitable for machine learning, the label would be binary - ‘1’ for the vessels that had previously been caught doing IUU fishing by the authorities and ‘0’ for the vessels that haven’t been caught by the authorities, we obtained the list of vessels that had been caught previously from various RFMOs, research papers, and news articles. We then added several features based on existing attributes to improve the performance of the models. The labeled dataset we obtained was highly imbalanced and that is why we had to apply machine learning techniques like oversampling (SMOTE), undersampling, and cost-sensitive learning.

Ii. Dataset Preparation

In this study, we utilized the AIS Disabling Events dataset obtained from the Global Fishing Watch[3][4]. The dataset contained the Maritime Mobile Service Identity (MMSI) numbers of ships that disabled their Automatic Identification System (AIS), along with their vessel class, flag, vessel tonnage, vessel length, latitude/longitude of when they disabled and enabled their AIS, number of hours for which AIS was disabled (gap hours), and the time when AIS disabled and enabled.

To identify vessels of interest we needed labeled data of which vessels were caught doing IUU fishing previously so that we could train our machine-learning model on that data. To do that, we collected the vessel identity information from various Regional Fisheries Management Organizations (RFMOs), including the South Pacific Regional Fisheries Management Organisation (SPRFMO)[5], the Commission for the Conservation of Antarctic Marine Living Resources (CCAMLR) [6], the Commission for the Conservation of Southern Bluefin Tuna (CCSBT)[7], General Fisheries Commission for the Mediterranean (GFCM)[8], the Inter-American Tropical Tuna Commission (IATTC) [9], the Indian Ocean Tuna Commission (IOTC)[10], Northwest Atlantic Fisheries Organization (NAFO) [11], North East Atlantic Fisheries Commission (NEAFC)[12], The North Pacific Fisheries Commission (NPFC)[13], South East Atlantic Fisheries Organisation (SEAFO)[14], Southern Indian Ocean Fisheries Agreement (SIOFA)[15], Western and Central Pacific Fisheries Commission (WCPFC)[16] manually, along with these RFMOs we also used the IUU Vessel List provided by TM-Tracking is a non-profit organisation that provides national fisheries authorities and international organisations with fisheries intelligence, analysis, and capacity building with the goal of reducing illegal fishing and improving ocean governance more broadly.[17]. We also tried finding such vessels in news articles or other research papers[4][27][28]. These datasets contained details of vessels that had been caught previously in Illegal, Unreported, and Unregulated (IUU) fishing activities within their respective regions. However, the datasets were disorganized and some lacked the identity information of some of the vessels and a lot of the RFMOs had cross-listed the vessels meaning a lot of the vessels were common for various RFMOs.

We combined the vessel identity information we gathered from the RFMO datasets into a single excel file and used a python script to obtain the MMSI numbers of the respective International Maritime

Organization (IMO) numbers from the Global Fishing Watch API. After obtaining the MMSI numbers, we combined them with the AIS Disabling Events Dataset and found 13 MMSI numbers that had been caught doing IUU fishing according to various RFMOs and also disabled their AIS according to the Global Fishing Watch dataset.

The existing attributes in the GFW AIS Disabling Events dataset weren't enough to train machine learning models. To generate more features from the available attributes, we wrote some python scripts[34]. The features generated included -

- The gap hours, which meant the number of hours the vessel was dark, this feature was included in the original GFW AIS Disabling Events dataset.

- The spherical distance traveled during the AIS disabling period, for this, we used the latitude/longitude of when the AIS was disabled and when it was enabled again.
- To determine whether or not AIS was disabled within the Exclusive Economic Zone (EEZ), we used the distance from shore attribute from the GFW AIS Disabling Dataset. The EEZ is a 200 nautical miles (370400 meters) band from a country's shore in which only the said country can conduct economic activities.
- The exact gear type of the vessel, which was obtained by passing the MMSI to the GFW API.
- The average speed of the vessel during the disabling event which was obtained by dividing the spherical distance by the gap hours.
- To determine during which part of the day the disabling event took place, here we divided the day into 6 parts each of 4 hours starting from 00:00 hours to 24:00 hours and classified all values into one of these 6 categories.
    - 00:00 hours – 04:00 hours = twilight
    - 04:00 hours – 08:00 hours = dawn
    - 08:00 hours – 12:00 hours = morning
    - 12:00 hours – 16:00 hours = afternoon
    - 16:00 hours – 20:00 hours = evening
    - 20:00 hours – 24:00 hours = night

As seen in Fig. 1. there are certain times of the day when AIS gets disabled more often.

- The identification of the ocean in which AIS was disabled, for this we used the start latitude and longitude of the disabling event to identify the ocean. To accurately identify the ocean, initially, we utilized GeoJSON files obtained from the FAO[18] which contained the major fishing areas and ocean names. Then we wrote a python script to find in which zone the lat/long for each disabling event lay, but due to the complexity of the GeoJSON, we were getting thousands of events as False meaning they didn't lie in any ocean. So then we went to the International Hydrographic Organization (IHO) website, which contained the GeoJSON for all the oceans and so we downloaded the JSON for all the oceans separately - Arctic ocean[19], Atlantic ocean[20], Baltic sea[21], Indian ocean[22], Mediterranean sea[23], Pacific ocean[24], South China sea[25], Southern ocean[26] and ran a for loop in python checking each point in every ocean's JSON. But, getting only the ocean name wasn't enough as most of the disabling events took place in 2 oceans namely the Pacific and the Atlantic, so we extracted the exact ocean name, every ocean is divided into small parts or seas like the Atlantic can be divided into North Atlantic, South Atlantic, etc. This gave a more detailed view of the ocean giving us 36 values for the ocean name attribute as compared to the previous 8.

The final dataset consisted of 55,129 entries of vessels not caught in IUU fishing activities and 239 entries of vessels caught in IUU fishing activities. The dataset was highly imbalanced. We then performed one-hot encoding on 3 of our categorical variables to make them viable for machine learning - gear type,

time of the day when AIS was disabled, and the ocean where AIS was disabled. Finally, we split our dataset into training and testing data with a 60/40 train/test split.

Iii. Machine Learning

After preparing the dataset, we started implementing machine learning models, since our dataset is highly imbalanced with 55,129 entries for the majority class v/s only 239 entries for the minority class, we applied methods such as oversampling, undersampling, and cost-sensitive learning. Our test set has 22,097 samples out of which 22,001 samples are negative samples '0' (low suspicion of IUU activity) and 96 samples are positive samples '1' (high suspicion of IUU activity). For our project, we would be using 2 performance measurement indicators namely -

- The Receiver Operating Characteristics Area Under the Curve (ROC_AUC) which is a measure of classification problem performance at various threshold settings, the top left-most point of the curve gives the optimal threshold setting for the given model, higher the ROC_AUC score the better the model is at classifying the samples.
- Recall, in our project the cost of misclassifying a positive case (vessel conducting IUU activity) as a negative case (vessel not conducting IUU activity) is more. The top left-most point of the ROC_AUC curve gives us the best trade-off between the recall of the majority and minority classes. The higher the recall, the better the model is performing.

Oversampling is a technique of creating artificial samples for the minority class, one of the popular oversampling techniques is the Synthetic Minority Over-sampling Technique (SMOTE)[29], which creates synthetic samples by randomly sampling the characteristics of the minority class. We oversampled our training data's minority class using SMOTE and kept the testing data separate so that the testing data doesn't get contaminated. We then trained various models like -

- Artificial Neural Network (ANN) with 3 dense layers (including 1 input and 1 output layer), the first layer had 50 neurons with Relu activation, the second layer had 15 neurons with Relu activation, and the output layer had 1 neuron with Sigmoid activation since we wanted to classify the sample as binary (0 - low suspicion of IUU activity, 1 - high suspicion of IUU activity). We used the Adam optimizer and 'binary cross entropy loss function. After training the model on X_train, we plotted the ROC_AUC graph, using which we found the best threshold value for classifying the predicted values of the test set. The recall for the majority and minority classes are 0.75 and 0.71 respectively for the test set.

Table 1
Confusion Matrix of ANN (oversampling)

		Actual Values	
		0	1
Predicted Values	0	16,524	5,477
	1	28	68

- XGBoost stands for “Extreme Gradient Boosting”, is an optimized distributed gradient boosting library designed for efficient training of machine learning models[30]. We used a high gamma value and a low max depth value because the model was overfitting. The recall for the majority and minority classes are 0.78 and 0.67 respectively for the test set.

Table 2. Confusion Matrix of XGBoost (oversampling)

		Actual Values	
		0	1
Predicted Values	0	17,134	4,867
	1	32	64

- Logistic Regression estimates the probability of an event occurring, in this case, was the vessel conducting IUU (1) or not (0), based on a given dataset of independent variables. The recall for the majority and minority classes are 0.91 and 0.40 respectively for the test set.
  Table 3. Confusion Matrix of Logistic Regression

  (oversampling)

		Actual Values	
		0	1
Predicted Values	0	20,073	1,928
	1	58	38

- Ensemble learning is a process in which multiple models are combined to solve a given problem and produce better results than the individual models[31]. Here, we did ensemble learning of XGBoost and Logistic Regression, with soft voting wherein the probabilities of each prediction in each model are combined and the prediction with the highest total probability is picked. The recall for the majority and minority classes are 0.90 and 0.47 respectively for the test set.

Table 4. Confusion Matrix of Ensemble Learning (oversampling)

		Actual Values	
		0	1
Predicted Values	0	19,773	2,228
	1	51	45

- Stacking is a method to explore different models for the same problem, here, we take some base models and train them on the training set, then we append the predictions (of the training set) of each of the base model to the training set, finally, we train the meta-model on the new training set containing the results of the base models. Here, we took the base models as ANN and Logistic Regression and the meta-model as XGBoost. The recall for the majority and minority classes are 0.91 and 0.40 respectively for the test set.

Table 5
Confusion Matrix of Stacking model (oversampling)

		Actual Values	
		0	1
Predicted Values	0	20,073	1,928
	1	58	38

As seen in Fig. 3. above, the best tradeoff for the recall of the majority and minority classes belongs to ANN and XGBoost for oversampling.

Undersampling is a technique to randomly remove samples from the majority class of the training dataset, resulting in a better class distribution, which can reduce the skew from a 1:100 to a 1:10, or like in our case to a 1:1[33]. We performed random undersampling on our training data's majority class and kept the testing data separate so that the testing data doesn't get contaminated. We then trained models similarly as we did while oversampling-

- Artificial Neural Network (ANN), with a similar configuration as used for oversampling. We used the Adam optimizer and 'binary cross entropy loss function. The recall for the majority and minority classes are 0.64 and 0.74 respectively for the test set.

Table 6
Confusion Matrix of ANN (undersampling)

		Actual Values	
		0	1
Predicted Values	0	14,033	7,968
	1	25	71

- XGBoost, for which the recall for the majority and minority classes are 0.77 and 0.77 respectively for the test set.

Table 7. Confusion Matrix of XGBoost (undersampling)

		Actual Values	
		0	1
Predicted Values	0	16,938	5063
	1	22	74

- Logistic Regression, with a similar configuration as used for oversampling The recall for the majority and minority classes, are 0.72 and 0.72 respectively for the test set.

Table 8. Confusion Matrix of Logistic Regression (undersampling)

		Actual Values	
		0	1
Predicted Values	0	15,942	6,059
	1	27	69

- Ensemble Learning with XGBoost and Logistic Regression, with soft voting. The recall for the majority and minority classes are 0.78 and 0.75 respectively for the test set.

Table 9. Confusion Matrix of Ensemble Learning (undersampling)

		Actual Values	
		0	1
Predicted Values	0	17,126	4,875
	1	24	72

- Stacking model with the base models as cost-sensitive ANN and Logistic Regression with the meta-model as XGBoost. The recall for the majority and minority classes are 0.29 and 0.96 respectively for the test set.

Table 10. Confusion Matrix of Stacking (undersampling)

		Actual Values	
		0	1
Predicted Values	0	6,315	15,686
	1	4	92

As seen in Fig. 4. above, the best trade-off for the recall of the majority and minority classes belongs to XGBoost and Ensemble Learning for undersampling.

Cost-Sensitive Learning is a method used when there is class imbalance and the cost of misclassifying a positive case as negative has serious consequences. Here, we assign weights to each of the classes, the higher the weight, the higher the cost of misclassifying that class[32]. For each of the models, we

performed a grid search to find the best weight for the minority class. We then trained various models like -

- Artificial Neural Network (ANN) with 3 dense layers (including 1 input and 1 output layer), the first layer had 50 neurons with Relu activation, the second layer had 15 neurons with Relu activation, and the output layer had 1 neuron with Sigmoid activation. We used the Adam optimizer and 'binary cross entropy loss function. The recall for the majority and minority classes are 0.75 and 0.56 respectively for the test set.

Table 11
Confusion Matrix of ANN (cost-sensitive)

		Actual Values	
		0	1
Predicted Values	0	16,430	5,571
	1	42	54

- Cost-Sensitive ANN, here with 4 dense layers (including 1 input and 1 output layer), the first layer had 63 neurons with Relu activation, the second layer had 30 neurons with Relu activation, the third layer had 10 neurons with Relu activation, and the output layer had 1 neuron with Sigmoid activation. We used the Adam optimizer and 'binary cross entropy loss function. The weight for the majority class is '1' whereas the weight for the minority class is '750'. The recall for the majority and minority classes are 0.81 and 0.69 respectively for the test set.

Table 12
Confusion Matrix of Cost-sensitive ANN

		Actual Values	
		0	1
Predicted Values	0	17,889	4,112
	1	30	66

- XGBoost with the parameter scale_pos_weight to implement class weighted XGBoost. After performing a grid search, we found the optimal value of the parameter to be 1000. The recall for the

majority and minority classes are 0.76 and 0.79 respectively for the test set.

Table 13. Confusion Matrix of XGBoost (cost-sensitive)

		Actual Values	
		0	1
Predicted Values	0	16,656	5345
	1	20	76

- Logistic Regression with the parameter class_weight to assign different class weights, here, we assigned the class weight for the majority class as '1' and '230' for the minority class. The recall for the majority and minority classes are 0.74 and 0.74 respectively for the test set.

Table 14. Confusion Matrix of Logistic Regression (cost-sensitive)

		Actual Values	
		0	1
Predicted Values	0	16,355	5,646
	1	25	71

- Ensemble Learning with XGBoost and Logistic Regression, with soft voting. The recall for the majority and minority classes are 0.79 and 0.76 respectively for the test set.

Table 15. Confusion Matrix of Ensemble Learning (cost-sensitive)

		Actual Values	
		0	1
Predicted Values	0	17,328	4,673
	1	23	73

- Stacking model with the base models as cost-sensitive ANN and Logistic Regression with the meta-model as XGBoost. The recall for the majority and minority classes are 0.78 and 0.71 respectively

for the test set.

Table 16. Confusion Matrix of Ensemble Learning (cost-sensitive)

		Actual Values	
		0	1
Predicted Values	0	17,156	4,845
	1	28	68

As seen in Fig. 7. above, the best trade-off for the recall of the majority and minority classes belong to XGBoost and Ensemble Learning for cost-sensitive learning.

As seen in Fig. 8. above, after comparing the best models of all the methods Oversampling (OS), Undersampling (US), Cost-Sensitive Learning (CS), the best trade-offs of the majority and minority recall belong to XGBoost (Cost-sensitive), Ensemble Learning (Cost-sensitive), and XGBoost (Undersampling).

Iv. Conclusion

In conclusion, this research explored the use of machine learning techniques to take a step towards solving the problem of IUU fishing, by identifying which AIS Disabling Events are suspicious. A unique dataset was created using feature engineering and manual data collection, which allowed us to train and test several models, including Logistic Regression, Artificial Neural Networks, XGBoost, Ensemble Learning, and Stacking. We also used advanced techniques like oversampling, undersampling, and cost-sensitive learning to tackle class imbalance in our dataset. Through the analysis of the results, we observed that the XGBoost method provided the best results with a recall value of up to 0.79 for the minority class for cost-sensitive learning, effectively addressing the high-class imbalance in the dataset. However, it is important to acknowledge the drawbacks of some of the methods - that synthetic data via oversampling does not fully capture the complexities and nuances of real-world scenarios such as this one and may lead to biases or inaccuracies in the model, and undersampling deletes examples from the majority class at random, which can result in the loss of information vital to a model.

V. Future Work

The more data is available, the better the model is going to perform. Getting more labelled disabling events will lead to better performance in the models. Getting a more comprehensive list of vessels that have been caught in IUU activities by the authorities can help label our current dataset in a better manner and might improve the performance significantly. A new attribute that takes in consideration the closest EEZ of where the disabling event took place and compares the flag of the vessel with that of the EEZ would be really helpful as the basis of Illegal fishing is when a vessel of a different flag/country conducts fishing activity in the EEZ of another country. Adding more meaningful features that can be derived from

existing attributes can improve the model's performance. These additions will enable us to make new breakthroughs in this domain and reduce our dependence on synthetically generated data. In addition, an interdisciplinary partnership between academics, policymakers, and stakeholders is necessary to assure the ethical and socially just deployment of machine learning in this field.

Vi. Statements

The authors would like to acknowledge that no funding was received for this research study.

Anmaya Agarwal wrote the dataset preparation, machine learning, conclusion, and future work. Jay Gala and Saketh Mantha assisted by making the tables and figures along with writing the abstract and introduction. Yash Katariya wrote the keywords and references. Prashashti Kanikar helped write the introduction and helped in the formatting. All authors reviewed the manuscript.

The authors declare that they have no conflicts of interest to report.

The dataset used in this research is publicly available at the Global Fishing Watch website[3], the RFMO datasets can be found at the links in the references [5] to [17], the GeoJSONs can be found at the links given in the references [18] to [26], and the source code can be found at the link [34]

References

1. What is IUU fishing? “https://www.fao.org/iuu-fishing/background/what-is-iuu-fishing/en/”
2. The toll of illegal, unreported and unregulated fishing “https://www.un.org/en/observances/end-illegal-fishing-day#:~:text=According%20to%20the%20UN%20Food,of%20US%2410%E2%80%9323%20billion”
3. Hotspots of unseen fishing vessels “https://globalfishingwatch.org/data-download/datasets/public-welch-et-al-disabling-events:v20221102”
4. Heather Welch, Tyler Clavelle, Timothy D. White, Megan A. Cimino, Jennifer Van Osdel, Timothy Hochberg, David Kroodsma, and Elliott L. Hazen “Hot spots of unseen fishing vessels”, 2022, Science Advances, Vol 8, Issue 44, DOI: 10.1126/sciadv.abq2109
5. South Pacific Regional Fisheries Management Organisation (SPRFMO) “https://www.sprfmo.int/fisheries/conservation-and-management-measures/cmm-04-iuu-fishing/iuu-lists#SPRFMO”
6. The Commission for the Conservation of Antarctic Marine Living Resources (CCAMLR) “http://www.ccamlr.org/en/compliance/illegal-unreported-and-unregulated-iuu-fishing”
7. The Commission for the Conservation of Southern Bluefin Tuna (CCSBT) “https://www.ccsbt.org/en/content/iuu-vessel-lists”
8. General Fisheries Commission for the Mediterranean (GFCM) “http://www.fao.org/gfcm/data/iuu-vessel-list”

9. The Inter-American Tropical Tuna Commission (IATTC) “https://www.iattc.org/VesselRegister/IUU.aspx?Lang=en”
10. The Indian Ocean Tuna Commission (IOTC) “http://www.iotc.org/vessels#iuu”
11. Northwest Atlantic Fisheries Organization (NAFO) “https://www.nafo.int/Fisheries/IUU”
12. North East Atlantic Fisheries Commission (NEAFC) “http://www.neafc.org/mcs/iuu”
13. The North Pacific Fisheries Commission (NPFC) “https://www.npfc.int/npfc-iuu-vessel-list”
14. South East Atlantic Fisheries Organisation (SEAFO) “http://www.seafo.org/Management/IUU”
15. Southern Indian Ocean Fisheries Agreement (SIOFA) “http://www.apsoi.org/mcs/iuu-vessels”
16. Western and Central Pacific Fisheries Commission (WCPFC) “http://www.wcpfc.int/wcpfc-iuu-vessel-list”
17. Combined IUU Vessel List “https://www.iuu-vessels.org/Home/Download”
18. FAO Statistical Areas for Fishery Purpose - FAO_Areas_CWP.geojson “https://data.apps.fao.org/map/catalog/srv/eng/catalog.search#/metadata/ac02a460-da52-11dc-9d70-0017f293bd28”
19. Arctic ocean “https://www.marineregions.org/gazetteer.php?p=details&id=1906”
20. Atlantic ocean “https://www.marineregions.org/gazetteer.php?p=details&id=1902”
21. Baltic sea “https://www.marineregions.org/gazetteer.php?p=details&id=2401”
22. Indian ocean “https://www.marineregions.org/gazetteer.php?p=details&id=1904”
23. Mediterranean sea “https://www.marineregions.org/gazetteer.php?p=details&id=4278”
24. Pacific ocean “https://www.marineregions.org/gazetteer.php?p=details&id=1903”
25. South China sea “https://www.marineregions.org/gazetteer.php?p=details&id=4331”
26. Southern ocean “https://www.marineregions.org/gazetteer.php?p=details&id=1907”
27. Analysis of the Southeast Pacific Distant Water Squid Fleet “https://globalfishingwatch.org/wp-content/uploads/GFW-2021-FA-SQUID2020-EN1-4.pdf”
28. News articles “https://www.reuters.com/article/us-argentina-defense-china-idUSKCN0WH2QL”, “https://osf.io/preprints/marxiv/juh98/”, “https://es.mongabay.com/2020/05/oceanos-pesca-ilegal-en-argentina/”
29. Chawla, N. V., Bowyer, K. W., Hall, L. O., and W. P. Kegelmeyer. "SMOTE: Synthetic Minority Over-sampling Technique." *ArXiv*, (2002). Accessed February 18, 2023. https://doi.org/10.1613/jair.953.
30. Chen, Tianqi, and Carlos Guestrin. "XGBoost: A Scalable Tree Boosting System." *ArXiv*, (2016). Accessed February 18, 2023. https://doi.org/10.1145/2939672.2939785.
31. F. Huang, G. Xie and R. Xiao, "Research on Ensemble Learning," 2009 International Conference on Artificial Intelligence and Computational Intelligence, Shanghai, China, 2009, pp. 249-252, doi: 10.1109/AICI.2009.235.
32. Elkan, Charles. "The foundations of cost-sensitive learning." In International joint conference on artificial intelligence, vol. 17, no. 1, pp. 973-978. Lawrence Erlbaum Associates Ltd, 2001.

33. Anand, Ashish, Ganesan Pugalenthi, Gary B. Fogel, and P. N. Suganthan. "An approach for classification of highly imbalanced data using weighting and undersampling." *Amino acids* 39 (2010): 1385-1391.
34. Source Code “https://github.com/Anmaya1856/iuu-fishing”

Figures

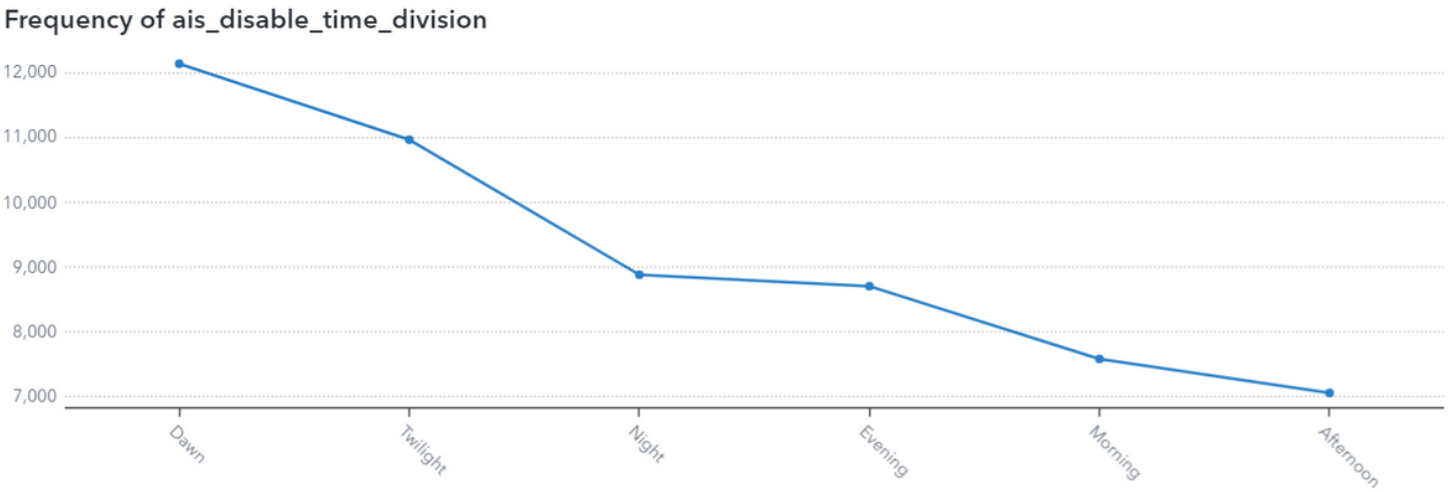

Figure 1

The frequency of AIS disabling events during different times of the day

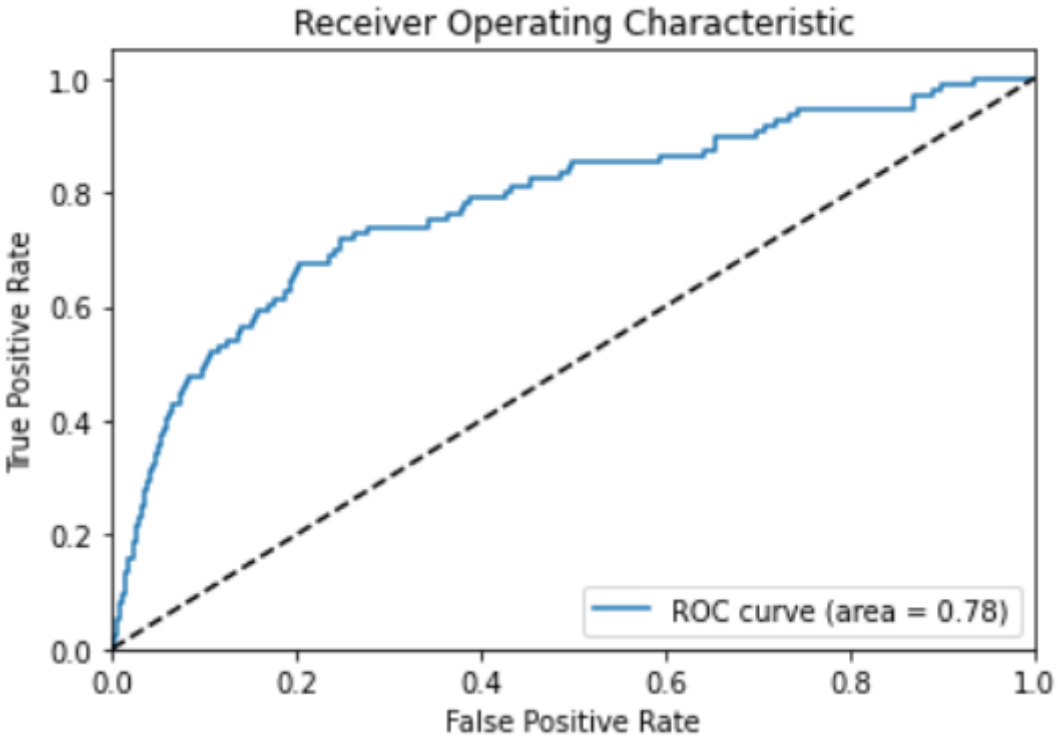

Figure 2

ROC_AUC curve of ANN (oversampling)

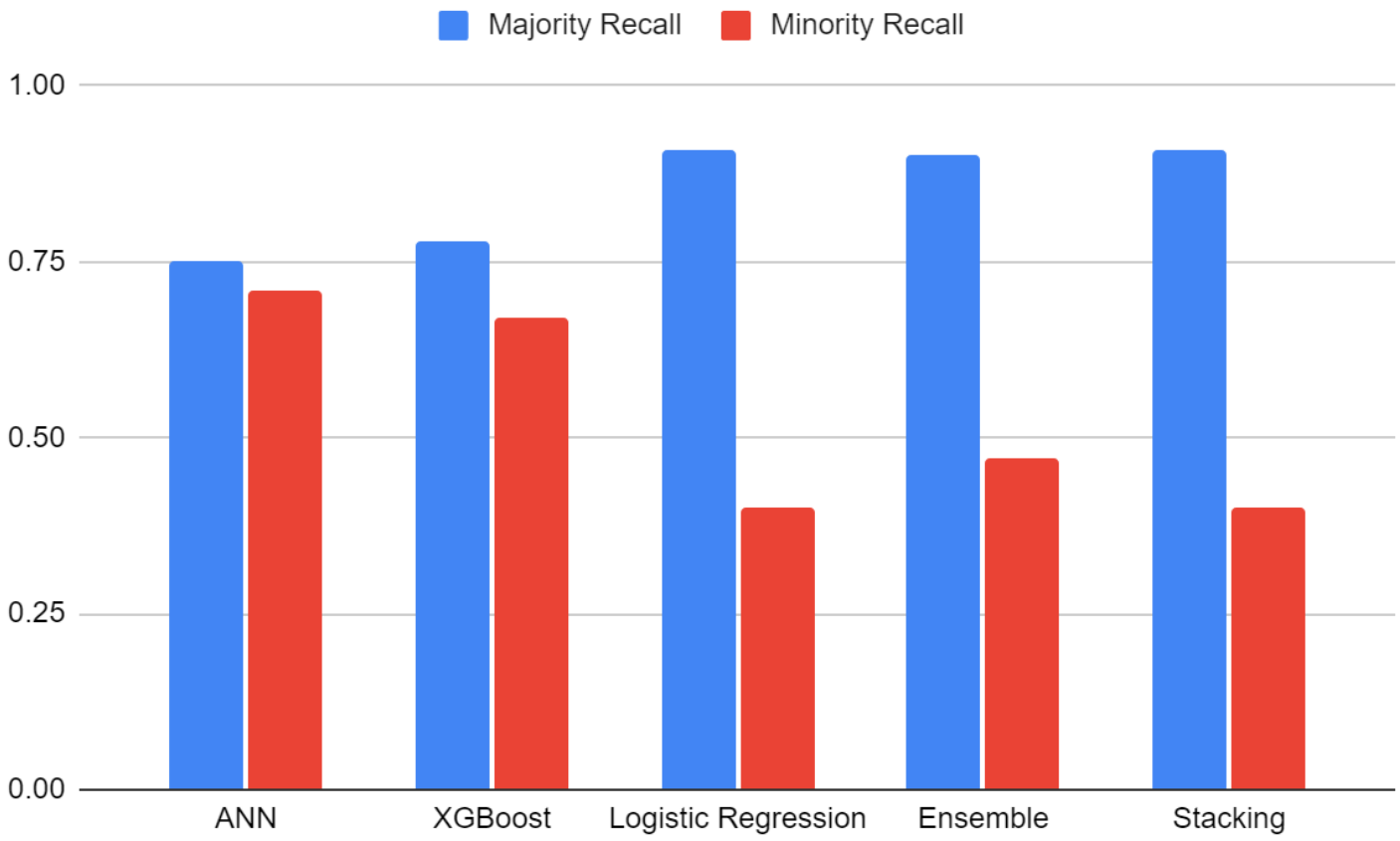

Figure 3

Recall comparison of various models after oversampling

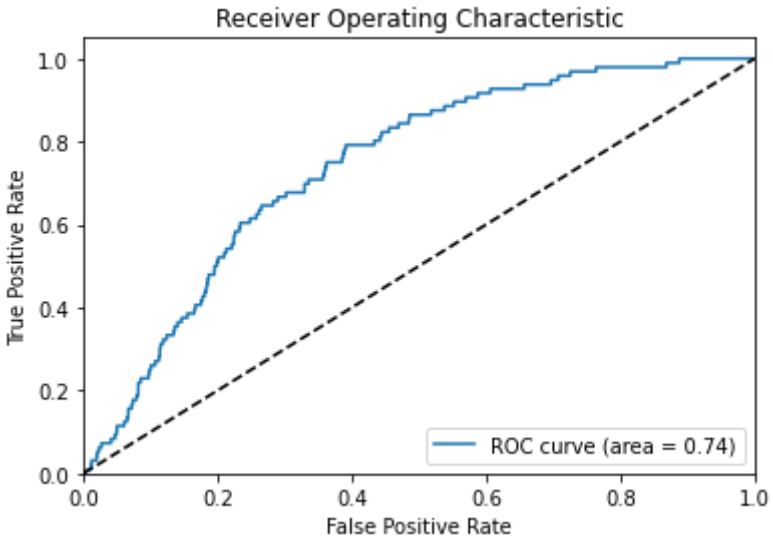

Figure 4

ROC_AUC curve of ANN (undersampling)

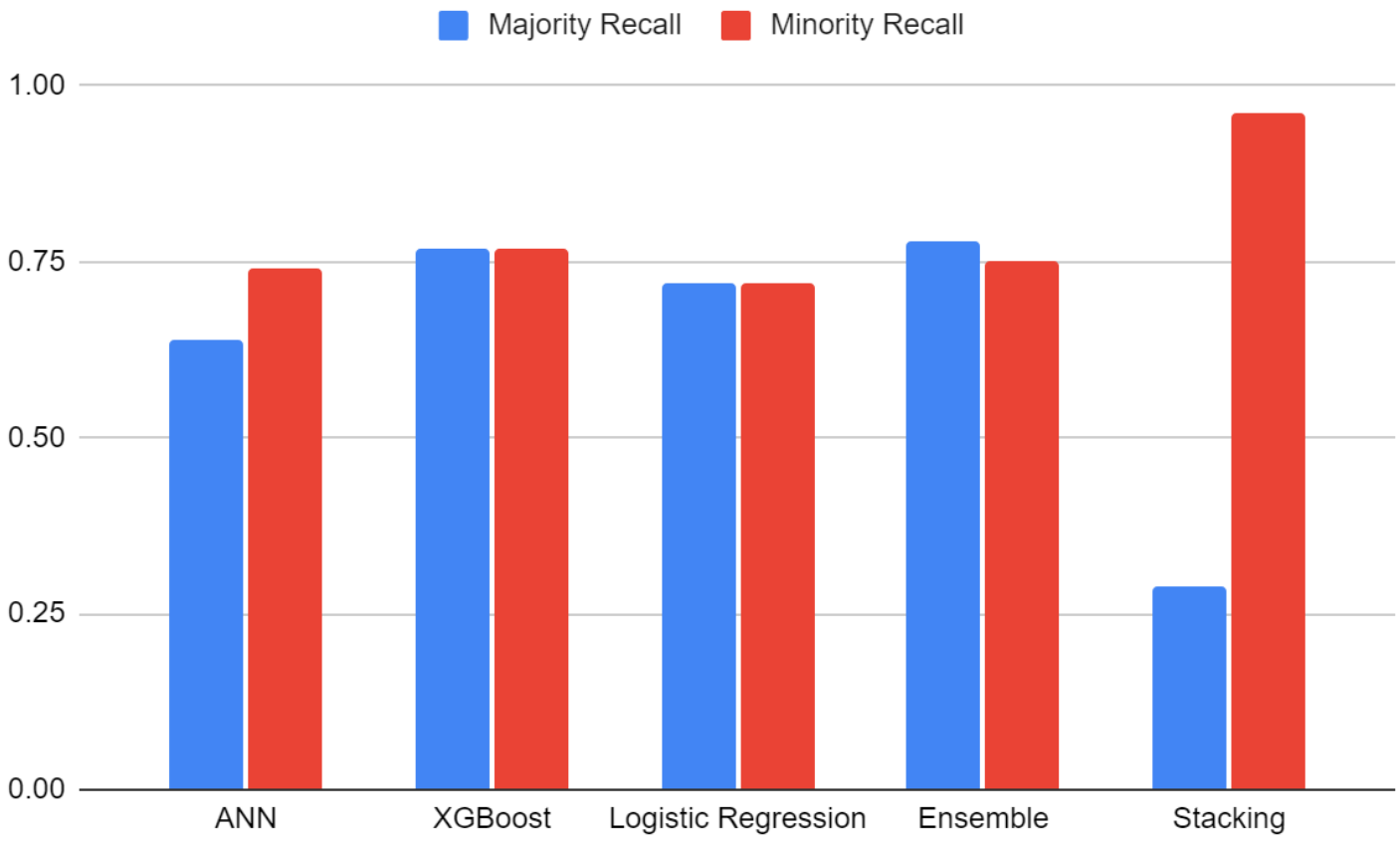

Figure 5

Fig. 4. Recall comparison of various models after undersampling

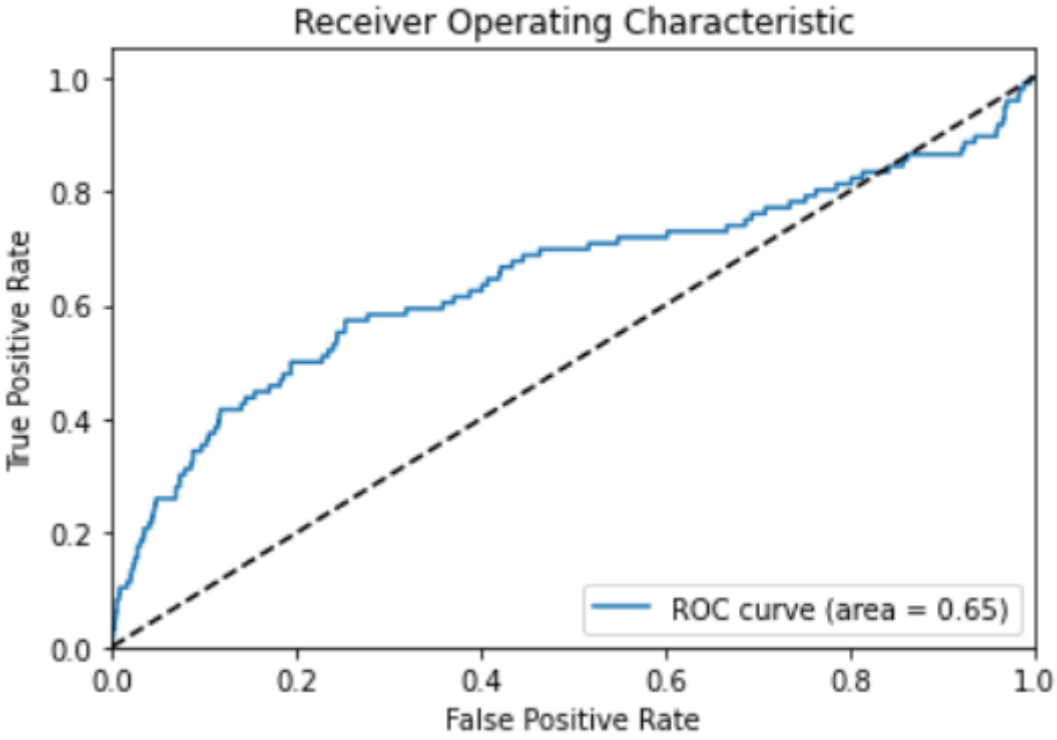

Figure 6

Fig. 5. ROC_AUC curve of ANN (cost-sensitive)

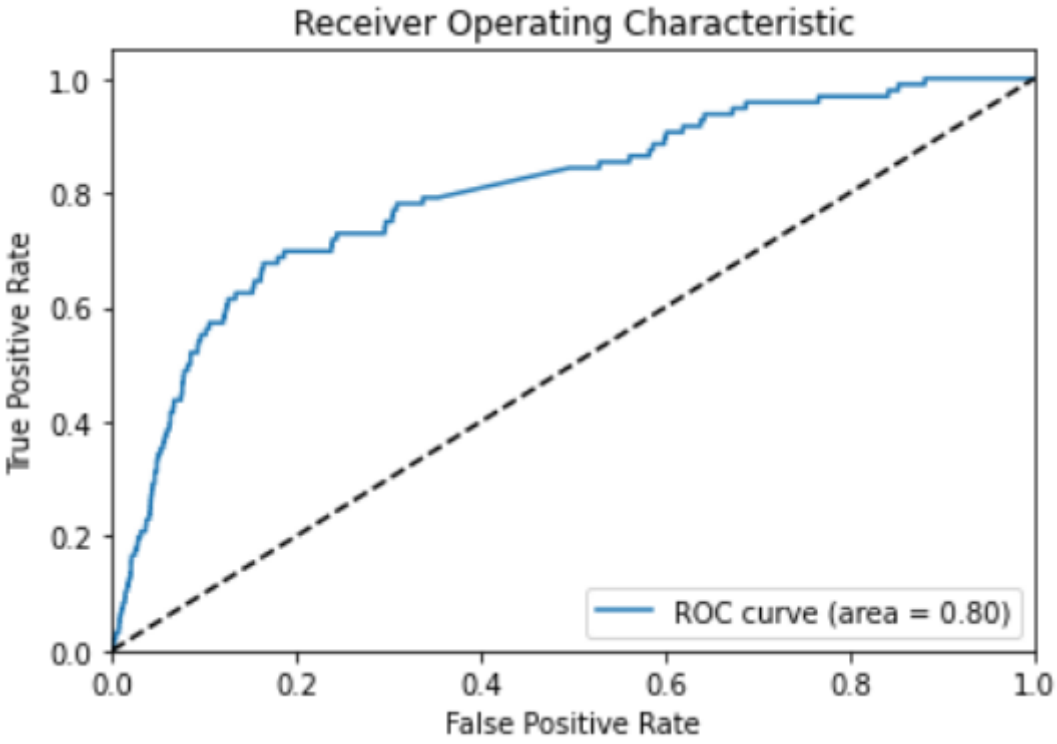

Figure 7

Fig. 6. ROC_AUC curve of Cost-sensitive ANN

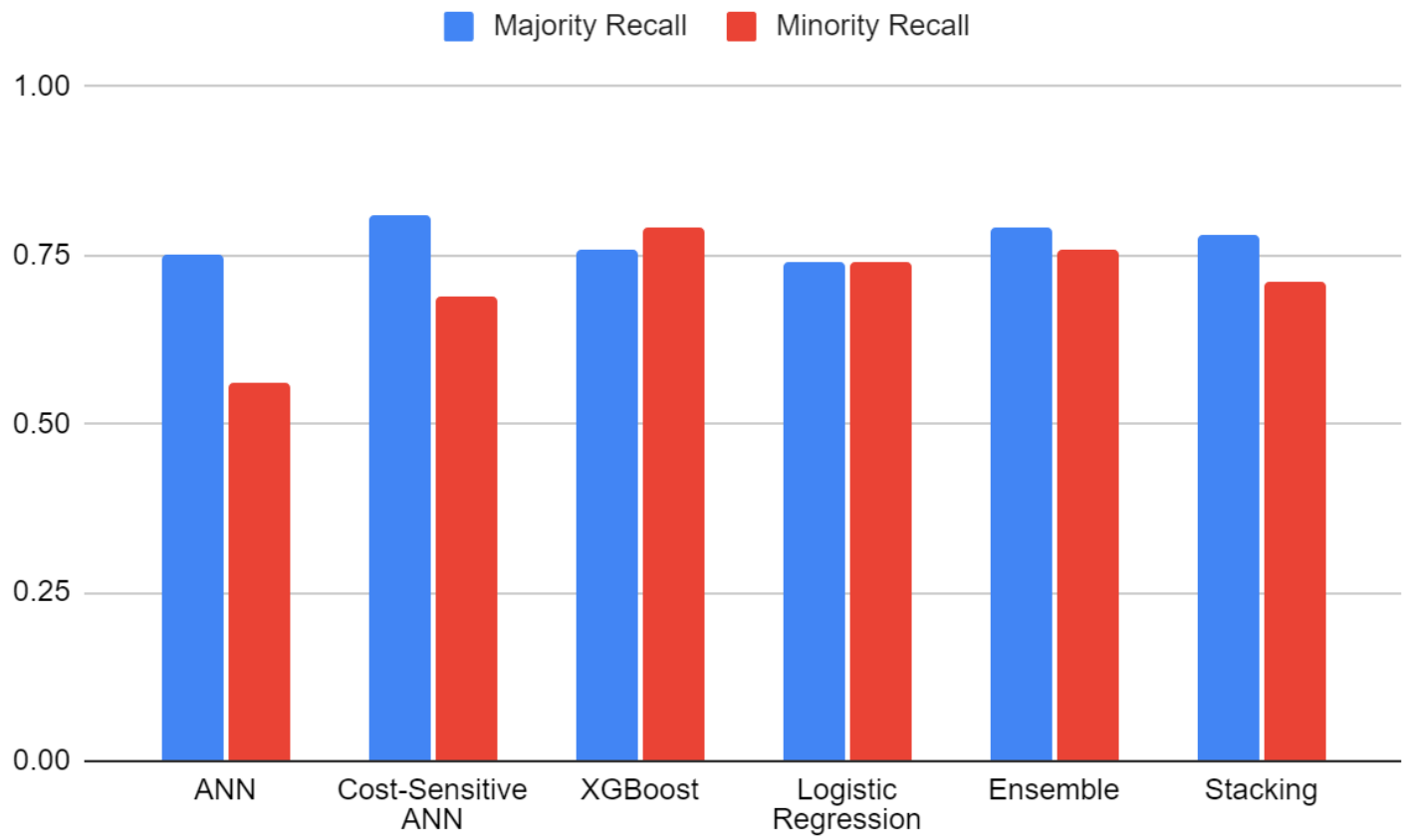

Figure 8

Fig. 7. Recall comparison of various models using cost-sensitive learning

Figure 9

Fig. 8. Recall comparison of the best models of all the methods