

Article

An Unmanned Underwater Vehicle Torpedoes Attack Behavior Autonomous Decision-Making Method Based on Model Fusion

Liqiang Guo ^{1,*} , Liang Ma ¹, Hui Zhang ¹, Jing Yang ², Zhuo Cheng ¹ and Wenmin Jiang ¹¹ Naval Submarine Academy, Qingdao 266199, China² School of Systems Engineering, National University of Defense and Technology, Changsha 410073, China

* Correspondence: javis019_ai@126.com

Abstract: The autonomous technology of unmanned platforms is the most dynamic frontier among fields of technology and, inevitably, is trending towards future development. Aiming at the dual requirements of reliable and real-time autonomous decision-making of unmanned underwater vehicles in complex and unfamiliar environments, this article proposes an intelligent decision-making method of attack behavior based on model fusion. The experimental dataset is generated through simulation modeling, and an appropriate amount of noise is added to simulate the observation error in a real situation. The threshold of weapon-hit probability is set according to the requirements of combat missions, and the decision-making of attack behavior is transformed into the problem of imbalanced sample classification with noisy data. Through theoretical analysis and experimental testing, the classification effects of algorithms such as Logistic Regression (LR), K-nearest neighbor (KNN), support vector machine (SVM), multilayer perceptron (MLP), decision tree (DT), and ensemble learning are compared. On this basis, the intelligent decision model is constructed by using synthetic minority oversampling technique resampling and three model fusion methods of voting, stacking, and blending. The experimental results show that compared with traditional simulation decision-making and common classification algorithms, the proposed method has higher accuracy, recall rate, area-under-the-curve value, and model generalization ability. It can not only effectively identify the impact of noise data on attack-behavior decision-making, but also ensures the decision-making speed through offline training, and provides references for the research in the field of equipment development and intelligent decision-making in the future.

Keywords: autonomous decision-making; unmanned underwater vehicle; imbalanced classification; model fusion; machine learning



Citation: Guo, L.; Ma, L.; Zhang, H.; Yang, J.; Cheng, Z.; Jiang, W. An Unmanned Underwater Vehicle Torpedoes Attack Behavior Autonomous Decision-Making Method Based on Model Fusion. *Electronics* **2022**, *11*, 3097. <https://doi.org/10.3390/electronics11193097>

Academic Editors: Hongbin Deng, Jifeng Guo and Chengchao Bai

Received: 22 August 2022

Accepted: 24 September 2022

Published: 28 September 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Unmanned platforms represented by unmanned aerial vehicles (UAVs), unmanned ground vehicles (UGVs), and unmanned underwater vehicles (UUVs) have become an important force in modern military equipment, and are promoting great changes to traditional forms and modes of combat. From the perspective of operational concept, new operational styles such as unmanned cluster operations and multi-domain collaborative operations [1,2] have been put forward one after another. From the perspective of combat missions, the mission field of unmanned platforms began to expand from logistics support to main attack and main battle. From the perspective of combat subjects, unmanned platforms have gradually replaced some manned platforms and have become the key force determining the outcome of war. Typical representatives are the Valkyrie UAV, the Manta large displacement unmanned underwater vehicle (LDUUV), the Talon UGV [3], etc.

Compared with manned platforms, military UUVs have obvious advantages in terms of concealment, flexibility, production cost, and operational risk. In most cases, they will be deployed to perform tasks in unknown battlefields. The combat mission behaviors of unmanned platforms need to be carried out as directed by the platforms' independent

decision-making, which requires them to be able to make independent and reliable decisions and execute combat missions efficiently in complex and uncertain environments.

At present, autonomous attack decision-making of UUVs mainly refers to manned platforms, while the traditional torpedo-attack decision-making of manned platforms mostly relies on computer simulation technology, establishes the task behavior model of various entities through logical rules, and analyzes and makes decisions according to the optimization results [4–7]. However, with the increase in the complexity of the battlefield environment, this method of simulating target dispersion and observation system error with random samples leads to an exponential increase in the calculation workload, which impedes rapid and accurate decision-making in practical operations. If the current methods of torpedo-attack decisions of manned platforms continue to be used, the operational potential of unmanned platforms cannot be fully reached [8].

The development of artificial intelligence technology has led to data-driven intelligent decision-making technology introducing new ideas to solve these problems. From the perspective of machine learning, UUV attack-behavior decision-making can be transformed into a binary classification problem with unbalanced sample categories by setting the threshold of the effectiveness index, and, then, improving the classification accuracy and generalization ability of the decision model. In recent years, model fusion technology based on mature ensemble algorithms has been widely used in the industrial field due to its excellent classification performance [9–12]. In the latest version of 2017–2042 US Unmanned Systems Integrated Roadmap [13,14], the US Department of Defense also regards artificial intelligence and machine learning as the primary supporting factors for improving the autonomy of unmanned systems.

Aiming at the dual requirements of UUVs for the accuracy and speed of torpedo-attack autonomous decision-making, this paper first constructs a torpedo-attack decision-making model under typical scenarios in the first chapter, and, then, in the second chapter, the model fusion strategy is analyzed theoretically. In the third chapter, according to the evaluation results of the classifier in the confusion matrix, the intelligent decision fusion model of UUV torpedo behavior based on stacking is constructed. Finally, the effectiveness and applicability of the method are analyzed and summarized in the fourth chapter.

2. Torpedo Attack Modeling

The attack process of UUVs launching torpedoes can be expressed as follows: after receiving the command to attack the target, the unmanned platform maneuvers itself to make contact with the enemy according to the planned route. When reaching the firing position, it launches torpedoes with firing advance angle φ and speed V_w to strike. When the weapon is D_g kilometers away from the target, the target will launch acoustical decoys to interfere with the weapon's homing device.

As shown in Figure 1, assuming that the effective action range of the torpedoes' acoustic homing device is a sector, the covering center coefficient of the sector is expressed by a constant k , the torpedoes' acoustic homing action distance $r = k * r_0$, which meets the encounter condition between the midpoint a of a leading edge of the shooting sector, the target M and the hit point C , that is, $\triangle aMC$ is the encounter triangle, and the calculation formula of the firing advance angle φ is obtained as follows:

$$\begin{aligned} m &= \frac{V_m}{V_w} \\ \varphi_0 &= \arcsin(m \sin Q_m) \\ \beta &= \arctan \frac{\sin Q_m}{(D_g / mr) + \cos Q_m} \\ \varphi &= \varphi_0 - \beta \end{aligned} \quad (1)$$

where, V_m is the speed of the target, D_g is the firing distance of the torpedoes, Q_m is the chord angle of the target, and m is the ratio of the speed of the target V_m to the speed of the

torpedoes V_w . Using Equation (1), we calculate the firing advance angle φ according to the target motion law, relative situation, and equipment performance [15].

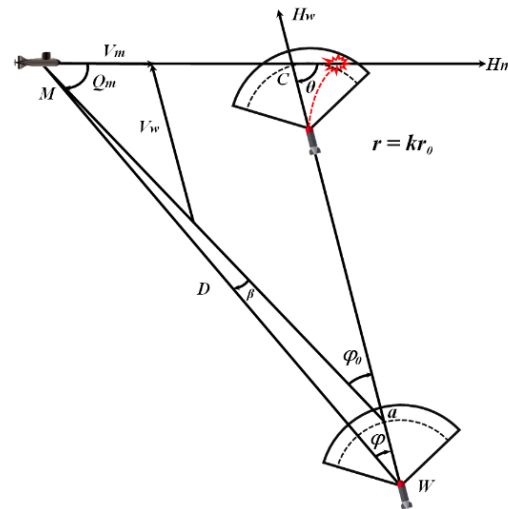


Figure 1. The diagram of torpedo firing advance angle calculation.

The effect of acoustical decoys on the torpedoes' homing devices is expressed by random error samples. Assuming that the target dispersion obeys the Gaussian (normal) distribution with coefficient θ and the number of samples is N_2 , the torpedoes' hit probability calculated by the Monte Carlo simulation method is $P = N_1 / N_2$, where N_1 is the number of torpedoes hitting the target. Taking the hit probability as the evaluation index of attack decision-making, and, then, according to the motion control logic of the UUV, target, and torpedoes, a combat simulation model based on the finite state machine [16] is established, and the attack decision-making results under different relative situations are obtained by using the model.

3. Model Fusion Theory and Strategy

Model fusion is an ensemble learning technology that trains multiple strong evaluators and fuses them with certain rules, which can further improve the classification accuracy based on the ensemble model. The essential idea of model fusion is to reduce the risk of the algorithm falling into the local optimal solution by expanding the hypothesis space of the model, so as to improve the generalization ability of the whole model.

3.1. Analysis of the Model Fusion Method

In the classification task, the commonly used model fusion methods [17] include voting, stacking, and blending.

Among them, voting means to fuse according to the majority rule, where the voting method based on the classification label category is called hard voting, and the voting method based on the predicted probability mean at a given threshold is called soft voting.

On the basis of the above voting methods, the performance improvement space of voting can be further expanded by weighted fusion. This idea is simple to operate, but it cannot guarantee the effect of fusion. Stacking fuses the results of the base learner by training the meta learner so that the fusion model is trained to minimize the loss function. The model has a strong learning ability and a wide application range, but the complexity and operation costs are high. Based on stacking, blending divides the pre-trained set into a training set and a validation set in advance, and directly inputs the prediction results of the validation set of the base learner into the meta learner layer, which improves the internal cross-training operation of stacking and greatly reduces the computational complexity of model training. However, due to artificially reducing the use of the training data, there may be some risk of overfitting.

3.2. Stacking Model Fusion Theory

Stacking is a parallel model fusion strategy with a multilayer learning structure proposed by Wolpert [18], in which the first layer learner is called the base learner and the second layer learner is called the meta learner. Stacking combines homogeneous or heterogeneous base learners, takes the original feature as the input of the first layer base learner, then takes the output prediction result as the input feature of the secondary meta learner, and outputs the final prediction result by the meta learner. On a given original dataset $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$, x_i represents the feature vector of the i -th sample, and $y_i \in \{C_1, C_2, \dots, C_j\}$ represents the category label. The training process of stacking can be divided into the following stages:

Step1: Divide the original dataset into training set D_{train} and test set D_{test} . Assuming that the number of base learners contained by the base learner layer is K , then D_{train} can be divided into k parts and recorded as $D_1 \sim D_k$.

Step2: Take D_1 to D_k as the verification set of each base learner, and the remaining sub-datasets as the training set of each base learner. After cross-validation by K-fold, vertically stack the results of cross-validation of all base learners to form a prediction set $Z = \{z_1, z_2, \dots, z_k\}$.

Step3: Use all training sets D_{train} to train all base learners in preparation for the model test.

Step4: Horizontally splice the prediction set Z of all base learners to form a new feature matrix, which combines with the original training set D_{train} to form a new dataset D_{new} , and put it into the meta learner for training.

It can be seen from the stacking training process in Figure 2 that, in order to prevent repeated learning of data, the training data of the stacking meta learner are not directly composed of the training set of the base learners, but adopt an internal cross-validation mechanism, so that each base learner retains a part of the set as the prediction set during training, and combines the results of k -numbered prediction sets into the input characteristics of the meta learner. In this way, the transformation of data from input to output is realized, which not only prevents overfitting, but also adaptively calculates the fusion rules of the base learner with the help of the meta learner algorithm.

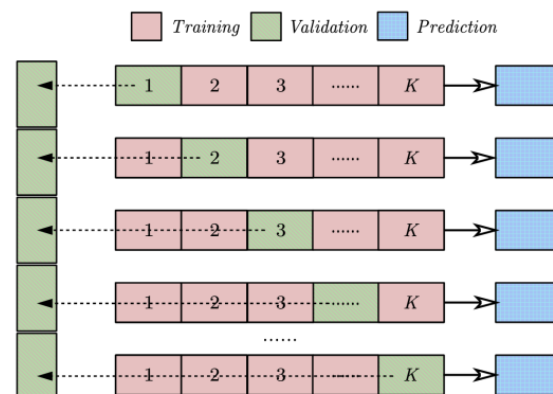


Figure 2. Diagram of the internal K-fold cross-validation process in stacking.

3.3. Stacking Model Fusion Strategy

The improvement effect of stacking is mainly determined by the design method of the base learner layer and meta learner. The generalization ability and training efficiency of the algorithm should be emphasized in the selection of meta learners. In terms of the construction of the base learning layer and because stacking is based on the idea of a high-dimensional hypothesis space, to ensure the stability of the base learner the diversity of algorithms should be increased as much as possible to improve the mutual independence of the base learners, which is conducive to reducing the square difference between the prediction results and the real label, and improving the generalization ability of the model as a whole; for example, the mixed use of a tree model, a linear model, a probability model, an ensemble model, etc. Therefore, through experiments from mature classification

algorithms such as logistic regression, K-nearest neighbor, support vector machine [19], multilayer perceptron, decision tree, random forest [20], AdaBoost [21], gradient boosting decision tree [22], eXtreme gradient boosting [23], and light gradient boosting machine [24], this paper selects the model with strong generalization ability and high computational efficiency as the meta learner and optimizes the configuration of the construction method of the base learning layer by taking into account the diversity and the stability. The principles and characteristics of the above classification algorithms are analyzed in Table 1.

Table 1. Analysis of the classification algorithm.

Algorithm	Principle	Advantages	Disadvantages
LR	Map the objective function values to the Sigmoid function and convert them into prediction probability, then conduct classification by the maximum likelihood function.	Good interpretability. Fast training speed.	Weak ability to deal with nonlinear or imbalanced sample problems.
KNN	Calculate the distance between samples and conduct classification.	Simple-structured model. Low computational complexity.	Insufficient ability to process high-dimensional features.
SVM	Map the samples to high-dimensional space by kernel function and classify by the hyperplane.	Strong learning ability for high-dimensional small sample data.	Depend too much on the kernel function for the accuracy of the model.
MLP	Based on the feed-forward network structure, apply to the artificial neural network model constructed by the back-propagation algorithm and activation function.	Able to deal with nonlinear problems. Strong and robust.	Greatly affected by discrete values. Low interpretability.
DT	Tree-structure model for top-down classification according to information-gain direction.	Strong model interpretability.	Weak ability to deal with missing data. Easy to overfit.
RF	Ensemble model based on bagging theory and classification and regression tree.	Strong ability to process high-dimensional data. Good tolerance to noise and outliers.	Weak ability to process small-sample data and low-dimensional data.
AdaBoost	Realize boosting by adaptively increasing the weight of prediction-error samples.	Flexible use and strong anti-overfitting ability.	Vulnerable to noise interference.
GBDT	Optimize the residual by gradient descent.	Flexible handling of multiple types of features. Supports parallel computing.	Difficult to parallelize. High computational complexity.
XGBoost	Based on GBDT, carry out the second-order Taylor expansion of the loss function and add the regular term.	Built-in cross-validation. Strong model-generalization ability.	Difficult parameter adjustment. High calculation cost.
LightGBM	Adopt histogram algorithm and branching strategy with depth constraints.	Fast training efficiency. Less memory consumption.	Sensitive to noise.

4. Experiments and Results Analysis

The experiment was carried out in the Windows environment, using MATLAB 2021b, JupyterLab 3.10 IDE, and python 3 kernels. The hardware configuration is an AMD Ryzen 5-5600H processor, the main frequency is 3.30 GHz, the number of cores is six, and the memory is 16 GB.

4.1. Dataset Generation and Processing

In the field of machine learning, the decision-making of attack behavior on unmanned platforms is a typical binary classification problem. The relative situation and equipment performance of UAVs, targets, and torpedoes can be summarized using the following eight characteristic parameters: torpedoes' firing distance D_g , target chord angle Q_m , target speed V_m , torpedoes' initial speed V_{w1} , torpedoes' terminal navigation speed V_{w2} , torpedoes' range L , target alarm distance D_s , torpedoes' acoustic homing action distance D_t to constitute the features matrix X of the sample. Define $y \in \{0, 1\}$ to represent the label of the sample, where 0 represents that the torpedoes' hit probability does not meet the

threshold condition and does not attack. 1 means that the torpedoes' hit probability meets the threshold condition and carries out attack behavior.

According to the requirements of combat missions, after setting the features' parameter space and the threshold conditions of the torpedoes' hit probability, the simulation model is input to generate the experimental dataset. As shown in Table 2, the number of positive samples is 4024 and the number of negative samples is 25,976, which has the obvious characteristics of sample imbalance.

Table 2. The experimental dataset.

Sample Size	Feature Dimension	Positive Sample Size	Negative Sample Size	Imbalanced Proportion	Sample Time
30,000	8	4024	25,976	1:6.46	3 min 26 s

4.2. Imbalanced Classification Evaluation Indicators

In the classification of imbalanced samples, it is not comprehensive to apply only a single index of accuracy and recall; therefore, the area under the curve (AUC) is introduced to jointly evaluate the performance of the model. The AUC is the area under the receiver operating characteristic curve (ROC), with a value between 0 and 1, which can comprehensively consider the classification accuracy of minority and majority classes.

According to the confusion matrix in Table 3, the calculation formulas of evaluation indicators of precision, recall, F1 value, and the AUC are as follows:

$$\text{precision} = \text{TP} / (\text{TP} + \text{FP}) \quad (2)$$

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (3)$$

$$\text{F1} = \frac{2 * \text{precision} * \text{recall}}{\text{precision} + \text{recall}} \quad (4)$$

$$\text{AUC} = \frac{\sum_{i \in \text{Positive Class}} \text{rank}_i - \text{NP}(\text{NP} + 1) / 2}{\text{NP} * \text{NN}} \quad (5)$$

where NP indicates the total number of positive samples (minority classes), NN indicates the total number of negative class samples (majority classes), rank_i indicates positive sample, and i indicates the confidence ranking of positive samples.

Table 3. The confusion matrix.

	Actual Positive Class	Actual Negative Class
Predicted positive class	TP (true positives)	FN (false negatives)
Predicted negative class	FP (false positives)	TN (true negatives)

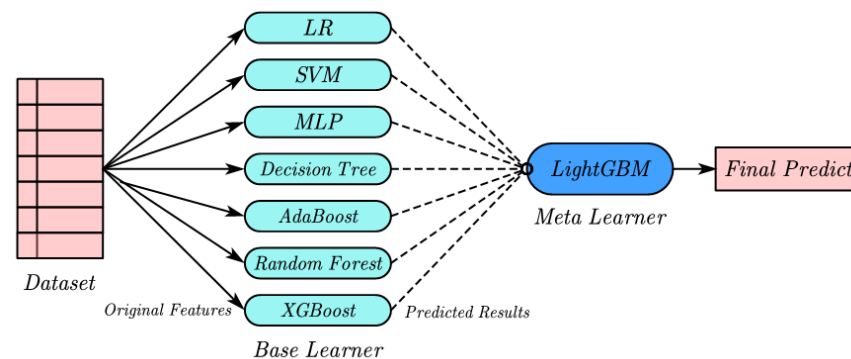
4.3. Experimental Design and Analysis

In order to simulate the measurement error of the observation system in the real situation, a certain amount of noise value is added to the characteristic matrix, and the training set and test set are randomly divided in the proportion of 7:3. Generally speaking, if the imbalanced proportion of sample categories is 4:1, the performance of the classifier will be affected. Since the imbalanced proportion of sample categories in the experimental dataset reaches 6.46:1, SMOTE [25] technology is used to adjust the imbalanced proportion of training samples to 3:1, and no adjustment is made in the test set. Taking accuracy, recall, and the AUC as the classification effect evaluation indicators, the performance of the above 10 classifiers is evaluated by the method of five-fold cross-validation, and the evaluation results are used as the benchmark for the performance evaluation of the fusion model. The evaluation results of 10 classifiers on the test set are shown in Table 4.

Table 4. The evaluation results of classifier performance.

Classifier	Accuracy	Recall	AUC	Test Time
LR	0.802	0.873	0.832	2.11 s
KNN	0.897	0.644	0.790	3.23 s
SVM	0.899	0.907	0.897	1 min 6 s
MLP	0.937	0.804	0.881	11.65 s
Decision	0.943	0.785	0.876	0.23 s
Tree	0.937	0.710	0.841	5.65 s
Random	0.951	0.831	0.900	2.39 s
Forest	0.956	0.818	0.897	5.01 s
AdaBoost	0.945	0.918	0.933	4.53 s
GBDT	0.916	0.941	0.936	2.17 s

According to the experimental results in Table 4, the ability of each classifier to recognize noise data and process imbalanced samples is different. The performance of the ensemble learning algorithm based on bagging and boosting is, obviously, better than other algorithms. Among them, LightGBM has high classification accuracy, a short time cost, and strong generalization ability. It is suitable to be used as the meta learner for the stacking and blending methods. The KNN algorithm performs weakly on this dataset, and GBDT is highly similar to XGBoost and LightGBM, so it is removed from the base learner queue. The Stacking fusion model is constructed as shown in Figure 3, and in order to enhance the anti-overfitting ability of the fusion model, the TPE [26] Bayesian optimization algorithm under the Optuna [27] framework is used to adjust the hyperparameters of the meta learner and fuzzy optimize the hyperparameters of the base learner. Three fusion methods, voting, stacking, and blending, are used to test the model fusion and decision-making effect, respectively.

**Figure 3.** Diagram of the stacking fusion model structure.

It can be seen from Figure 4 and Table 5 that the values of accuracy, recall, and the AUC of the decision-making model on the test set are significantly improved after the hyperparameters' optimization and model fusion. Among them, the blending model has the fastest prediction speed and the stacking model has the best prediction effect. The experimental results show that, compared with traditional simulation decision-making methods, the intelligent decision-making method based on model fusion can greatly speed up decision-making as well as ensure its reliability, and meet the needs of autonomous decision-making of unmanned platforms. At the same time, it has good universality and portability in tactical mission behavior decisions, such as attack and defense.

Table 5. Evaluation results of the fusion model performance.

Methods	Accuracy	Recall	AUC	Test Time
Voting	0.935	0.818	0.973	5.15 s
Stacking	0.941	0.938	0.987	5.38 s
Blending	0.927	0.909	0.984	3.12 s

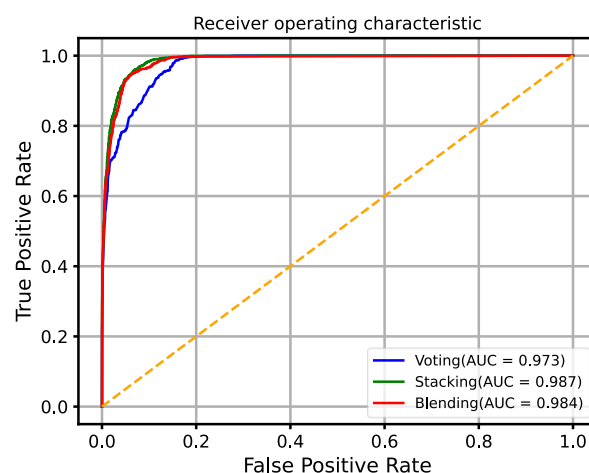


Figure 4. Diagram of the receiver operating characteristic of fusion models.

After analyzing the experimental results, it is considered that the voting fusion method has simple rules, but the model learning ability is limited, which is suitable for solving the problem of low data complexity. The stacking fusion method has strong learning ability, deep use of training data and wide application range, but the model computational cost is high and the risk of overfitting is also high, so it is necessary to reasonably configure the model combination of base learning layer and meta learner. The blending fusion method has a fast operation speed, a low cost, and a low risk of overfitting. However, due to the use of only part of the verification set for prediction, the learning space of the meta learner is very limited, so it is not suitable for a situation with fewer training data; therefore, when the number of data and amount of computing power are sufficient, the stacking fusion method is preferred. When the fusion model shows strong overfitting, it can be replaced with the voting method with its simple rules or the blending method with its shallow learning depth.

5. Conclusions

Aiming at the dual requirements of reliability and real-time performance of UAVs' autonomous decision-making in complex and unfamiliar environments, the attack-behavior decision is transformed into an unbalanced sample classification problem with noisy data. Firstly, the SMOTE method is used to adjust the unbalanced proportion of training set samples to a suitable range. Then, according to the performance of the base classifier, the configuration of the base learning layer and the meta learner are optimized, and the improvement effects of the three model fusion methods of voting, stacking, and blending are compared. The experimental results show that the AUC value of the model fusion method based on stacking reaches 98.7%, which has a stronger classification performance than other methods. Finally, it is concluded that this method is more suitable for autonomous decision-making of UAVs' torpedo attack behavior.

Due to the great intensity of the conflict between ourselves and the enemy, and the intensity of confrontation in the real combat environment, obtaining situation data is difficult and such data have large errors and represent weak real-time performance. In the next step, the time series data processing method will be introduced and explored to analyze, correct, and complete the observed and hitherto incomplete time series data, so as to improve the robustness of intelligent decision-making models.

Author Contributions: L.G.: data curation, formal analysis, and writing—original draft. L.M.: writing—review and editing, and supervision. H.Z.: validation and supervision. J.Y.: writing—review and editing, and validation. Z.C.: methodology. W.J.: validation. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Written informed consent for the publication of this paper was obtained from the Naval Submarine Academy.

Informed Consent Statement: Written informed consent for the publication of this paper was obtained from all authors.

Data Availability Statement: Data sharing not applicable to this article as no datasets were generated or analysed during the current study.

Acknowledgments: The authors would like to thank the Kunming Precision Machinery Research Institute for helpful discussions on topics related to this work.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Si, G.; Miao, Y.; Li, G. Underwater Tridimensional Attack-Defense System Technology. *Command Control Simul.* **2018**, *40*, 1–8.
- He, Y.; Qin, T.; Wang, N. Cross-domain Collaboration: New Trends in the Development and Application of Unmanned Systems Technology. *Unmanned Syst. Technol.* **2021**, *4*, 1–13.
- Wang, Y.; Yang, Y.; Wang, T.; Ge, Y. Summary of the Development of Unmanned Systems in 2019. *Unmanned Syst. Technol.* **2019**, *2*, 53–57.
- Wu, Z.; Yu, X.; Xu, Z. Analytic Formula and Employment of the Hitting Probability for Sub-Launched Torpedo. *J. Unmanned Undersea Syst.* **2021**, *29*, 203–209.
- Gong, Y.; Wu, Z.; Liu, Z. Optimal Modeling of Two-Time Rotating Angle Shoot Target for Acoustic Homing Torpedo. *Fire Control Command Control* **2018**, *43*, 82–85.
- Li, H.; Zhang, J. Operational Application Method of Two-time Turning Angle for Acoustic Homing Torpedo. *J. Unmanned Undersea Syst.* **2018**, *26*, 342–347.
- Cui, Z.; Zhang, Y.; Li, Z. Simulation Study of Surface Warship Evading Acoustic Homing Torpedo Based on Monte Carlo. *Comput. Meas. Control* **2017**, *25*, 92–94.
- Ma, L.; Guo, L.; Zhang, H.; Yang, J.; Liu, J. Demand for Intelligent Torpedo Attacking Decisions for Unmanned Under-Sea Vehicle and Method Discussions. Available online: <http://www.cnki.com.cn/Article/CJFDTotal-YLJS20220711000.htm> (accessed on 13 July 2022).
- Sun, W.; Deng, A.; Deng, M.; Liu, Y.; Cheng, Q. Fault Diagnosis of Wind Turbine Gearbox Based on Model Fusion. *Acta Energ. Sol. Sin.* **2022**, *43*, 64–72.
- Feng, X.; Xu, M.; Tang, Y.; Wu, Z. Industrial Product Quality Prediction Based on Stacking Multi-model Fusion. *Stat. Its Interface* **2021**, *51*, 266–276.
- Pan, G.; Gong, M.; He, M.; Wu, C.; Tang, X.; Yang, L.; Ouyang, J. Identification Method of Electricity Charge Recovery Risk of Specialized Transformer User Based on Stacking Model Fusion. *Electr. Power Autom. Equip.* **2021**, *41*, 152–158.
- Prediction Model for Production Well Hydraulic Fracturing Effect of Block SN in Daqing Oilfield Based on Machine Learning and Model Ensemble. Available online: <https://kns.cnki.net/kcms/detail/23.1286.TE.20220218.1456.002.html> (accessed on 21 February 2022).
- Li, L.; Wang, T.; Jiang, Q. Key Technology Develop Trends of Unmanned Systems Viewed from Unmanned Systems Integrated Roadmap 2017–2042. *Unmanned Syst. Technol.* **2018**, *1*, 79–84.
- Unmanned Systems Integrated Roadmap 2017–2042. Available online: https://cdn.defensedaily.com/wp-content/uploads/post_attachment/206477.pdf (accessed on 28 August 2018).
- Guo, L.; Ma, L.; Zhang, H.; Yang, J. Data-driven Autonomous Decision-making Method for the Effective Position of AUV Torpedo Attacks. *J. Unmanned Undersea Syst.* **2022**, *30*, 528–534.
- Cheng, J.; Zhang, H. Optimal Model and Model Solving Strategy of Submarine Torpedo Defense Using Acoustic Countermeasure Equipment. *Command Control Simul.* **2019**, *41*, 48–51.
- Ma, J. *Multi-Model Integrated Learning Methods and Applications*; Nanjing University: Nanjing, China, 2016.
- Wolpert, D.H. Stacked Generalization. *Neural Netw.* **1992**, *5*, 241–259. [[CrossRef](#)]
- Cortes, C.; Vapnik, V. Support Vector Machine. *Mach. Learn.* **1995**, *20*, 273–297. [[CrossRef](#)]
- Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5–32. [[CrossRef](#)]
- Freund, Y.; Schapire, R.E. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *J. Comput. Syst. Sci.* **1997**, *55*, 119–139. [[CrossRef](#)]
- Friedman, J.H. Greedy Function Approximation: A Gradient Boosting Machine. *Ann. Stat.* **2001**, *29*, 1189–1232. [[CrossRef](#)]
- Chen, T.; Guestrin, C. Xgboost: A Scalable Tree Boosting system. In Proceedings of the KDD '16, San Francisco, CA, USA, 13–17 August 2016; pp. 785–794.
- Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; Liu, T.Y. LIGHTGBM: A Highly Efficient Gradient Boosting Decision Tree. In Proceedings of the NIPS 2017, Long Beach, CA, USA, 4–9 December 2017; pp. 3146–3154.
- Shi, H.; Chen, Y.; Chen, X. Summary of Research on SMOTE Oversampling and Its Improved Algorithms. *CAAI Trans. Intell. Syst.* **2019**, *14*, 1073–1083.

-
26. Ozaki, Y.; Tanigaki, Y.; Watanabe, S.; Onishi, M. Multiobjective Tree-Structured Parzen Estimator for Computationally Expensive Optimization Problems. In Proceedings of the GECCO '20, Cancun, Mexico, 8–12 July 2020; pp. 533–541.
 27. Akiba, T.; Sano, S.; Yanase, T.; Ohta, T.; Koyama, M. Optuna: A Next-Generation Hyperparameter Optimization Framework. In Proceedings of the 25th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD '19, Anchorage, AK, USA, 4–8 August 2019; pp. 2623–2631.