Department of Computer Science
University of Toronto

Non-Invasive Fish Species Classification Using Deep Learning

A Hydroacoustic Approach for Sustainable Ecological Monitoring

Scarlett (Yi) Yang

scarlett.yang@mail.utoronto.ca

STA2453 Final Project Report

March 2025
Toronto, Canada

Contents

1 Abstract

Hydroacoustic monitoring offers a non-harmful way to monitor and classify fish species. This study developed a deep learning pipeline to distinguish Lake Trout (LT) and Smallmouth Bass (SMB) using hydroacoustic frequency responses. With 3,828 observations from 9 LT and 2,257 observations from 7 SMB, machine learning models including Logistic Regression, XGBoost, and Random Forest, as well as a deep learning Long Short-Term Memory (LSTM) network, were evaluated. All machine learning methods treat timesteps independently, ignoring the time-series nature data. The LSTM model leveraged sequential dependencies and achieved 73.8% accuracy under full leave-one-pair-out (LOPO) validation across all 63 LT/SMB pairs. The accuracy score falls short of the 80% target, but demonstrates significant improvement over machine learning baselines.

Feature engineering via Random Forest importance rankings and Principal Component Analysis (PCA) reduced 426 frequencies to 30 key features (Random Forest) and 10 principal components (PCA). Although the computational cost is higher than machine learning methods, the LSTM model's success in accuracy underscores the importance of considering sequential architectures for time-series hydroacoustic analysis. This work advances ecological monitoring and provides a scalable, non-invasive framework for species classification. Scalability is valuable for large aquatic ecosystems. Further work, including dataset expansion and model refinement, is recommended to bridge the gap between controlled experiments and real-world deployment.

2 Introduction

Ecological monitoring of inland fisheries is important for sustainable management. However, traditional methods of monitoring, such as trawling and netting, are expensive and harmful to aquatic life [1]. This project aims to develop a learning-based species classification pipeline using hydroacoustic data collected from Lake Trout (LT) and Smallmouth Bass (SMB). This pipeline uses models such as Long Short-Term Memory (LSTM) to analyze sequential patterns in sound waves that bounce back from fish across frequencies. The result provides a scalable and non-destructive solution to enhance fish monitoring programs, with lower costs and minimal disturbance to the environment compared to traditional methods. This research extends a project conducted by Leivesley and Professor Leos Barajas at the University of Toronto in 2024.

The dataset contains 6,085 observations, 3,828 for nine Lake Trout and 2,257 for seven Smallmouth Bass, with 483 features. The features include biological measurements such as total length and weight, as well as 426 frequency responses from 45 Hz to 260 Hz in steps of 0.5 Hz. Key challenges include sequential dependencies in frequency responses and class imbalance (63% LT v.s. 37% SMB). To ensure data quality, observations for inactive fish and those with missing values in key frequencies were removed. The main objective of this project is to build a deep-learning model that classifies fish species between LT and SMB with high accuracy using frequency-related features. Leivesley's original analysis achieved 80% balanced test accuracy, this research aims to reach the same level of accuracy to achieve at least 80% classification accuracy on the test data [1].

3 Data

The original raw dataset contained hydroacoustic recordings for four types of fish [1]. The hydroacoustic measurements were collected in a controlled experimental setup, ensuring the recordings capture the acoustic reflections over time for each fish. Since Leivesley chose LT and SMB for analysis, this research also retained only Lake Trout and Smallmouth Bass, focusing on these two species.

The dataset was preprocessed to make it suitable for model training. First, based on Leivesley's note, some inactive fish had unusual behavior during data collection. In addition, during the process of exploratory data analysis (EDA), some fish were found to have missing values in key frequencies. Those fish were considered "invalid" and were removed. Sixteen fish were retained after this step, including nine Lake Trout and seven Smallmouth Bass. For each fish, the key features are structured as shown in the following table 1. Demographic data, such as length and weight, were included for future reference.

Species	**ID**	**Time**	**Ping#**	**F45**	**F45.5**	...	**F260**
Lake Trout	10	15:11:06.1020	P_1	X_1	Y_1	...	Z_1
Lake Trout	10	15:12:07.2030	P_2	X_2	Y_2	...	Z_2
⋮	⋮	⋮	⋮	⋮	⋮	...	⋮
Lake Trout	10	15:20:12.5060	P_n	X_n	Y_n	...	Z_n

Table 1: Sample Data Table for Hydroacoustic Analysis

Below is an explanation of the key features.

- **Species**: The label/response variable, either Lake Trout or Smallmouth Bass.
- **ID**: The ID of the fish within a species.
- **Time**: The time of receiving the series of hydroacoustic responses, ranging from frequencies of 45 to 260 Hz.
- **Ping#**: The unique ID representing the series of hydroacoustic responses from frequencies of 45 to 260 Hz at a specific time.
- **F45-F260**: The hydroacoustic response for each frequency in the range of 45 to 260 Hz.

Next, fish labels were encoded: 0 for Lake Trout and 1 for Smallmouth Bass. Demographic information was removed for now because the focus is on hydroacoustic responses. For frequencies, response values were standardized using z-score normalization to ensure that all features, improving compatibility with models like LSTM. Neighboring frequencies are correlated as found in EDA. Therefore, using Random Forest importance ranking and PCA, 426 raw frequency features have been reduced to 30 key features (Random Forest) and 10 principal components (PCA). Additionally, note that there is a slight imbalance (63% LT v.s. 37% SMB). Synthetic minority oversampling techniques (SMOTE) was explored to address imbalance during training.

The most important aspect is the sequential dependency inside the hydroacoustic frequency data. Unlike traditional datasets where each row represents a single independent observation, each row is a part of a continuous temporal sequence for a fish. Whether a model can capture the time-series nature was considered when making decisions. Moreover, a traditional random train-test split would cause data leakage. Data leakage occurs when a model uses information that is not available at the time of prediction for training. To prevent this problem, cross-validation has been designed so that measurements for each fish are entirely in either the training or test set.

4 Methodology

This section walks through the model exploration.

4.1 Feature Engineering

First, the dataset contained more than 400 frequencies, many of which were highly correlated, especially neighboring frequencies. The heatmap 1 of the correlation matrix shows that most frequencies are moderately correlated. There are clusters of highly correlated frequencies, which suggest redundancy among neighboring features.

Therefore, a correlation analysis guided feature selection. Random Forest[2] importance rankings and Principal Component Analysis (PCA) [3] were then used to select the most informative features for classification. Random Forest Classifiers identified the top 30 frequencies and PCA further reduced these to 10 principal components. However, the selected ones did not explain substantially more variance than others.

A Random Forest (RF) is an ensemble learning method that combines a large number of decision trees, with the final prediction based on majority voting. It is more robust than a single decision tree. The Random Forest Classifier can identify which features are more important in a high-dimensional dataset. PCA is a dimensionality reduction technique that transforms correlated variables to uncorrelated principal components in a high-dimensional dataset to explain most of the information.

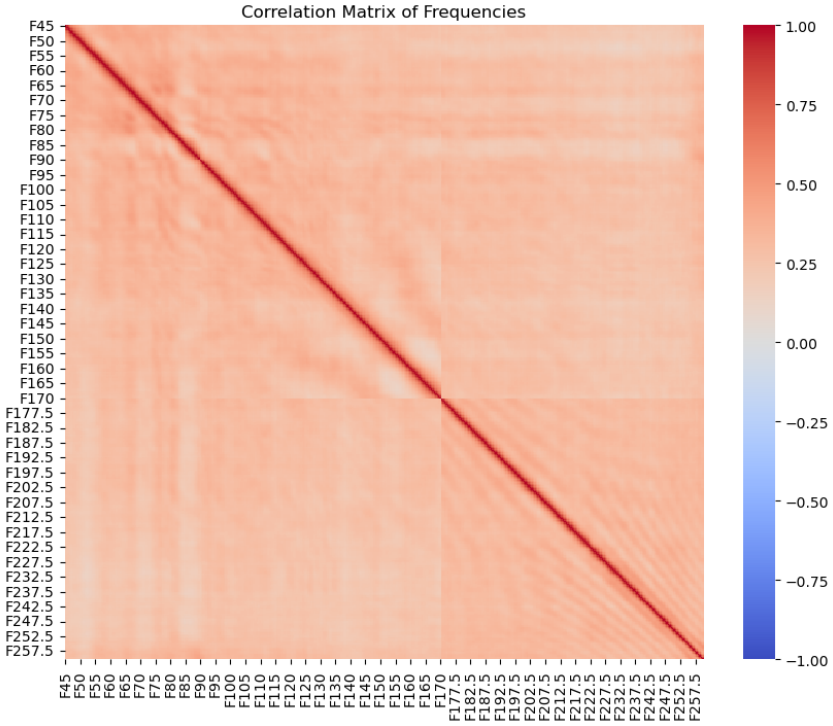

Figure 1: Correlation Matrix

4.2 Train Test Split

Since hydroacoustic data is not independent per record, using a traditional random train-test split would cause the same fish to appear in both training and test sets, leading to data leakage.

Thus, the initial splitting methods employed in this research are called Leave-One-Fish-Out (LOFO) and Leave-One-Pair-Out (LOPO). Each test set contains data for either a single fish or a fish pair (one LT and one SMB). This method ensures models are never exposed to test fish data during training.

Although LOFO and LOPO prevent data leakage, models perform poorly when trained using these splitting methods. Therefore, the stratified 5-fold method was also employed for cross-validation. Unlike the traditional K-fold method, stratified K-fold ensures that each fold has a similar distribution of classes. There are more LT than SMB, stratified K-fold puts more LT samples to the test set so that both species are balanced in the training set. Similar to LOFO and LOPO, the data for any individual fish is entirely in either the training or test dataset, preventing leakage.

4.3 Machine Learning

As stated in the Proposal, the initial approaches were machine learning models. Since the problem involves binary classification (LT v.s. SMB), Logistic Regression is an easy-to-train and interpretable model for classification problems. However, the logistic regression model trained using the key features indicated by PCA and Random Forest Classifier performed poorly, with both accuracies are 66%. These accuracies are close to random chance.

Logistic regression is a simple statistical model that aims to create a linear decision boundary. The non-linear relationship between hydroacoustic signals and class labels is the primary cause of its failure. The most important point is that fish species classification highly depends on response changes over time. Both logistic regression and Random Forest treat each time point independently. Without information from the sequential structure, it is impossible to provide a reliable result.

In addition to the two simple models, XGBoost was also trained. XGBoost builds scalable, distributed gradient-boosted decision trees [4]. Compared to Random Forest, XGBoost builds trees sequentially, with each new tree focusing on the errors made by previous trees to improve performance. It performed better than logistic regression and Random Forest, achieving an accuracy of 72% after hyperparameter tuning, PCA feature reduction, and stratified K-fold. Nonetheless, it does not capture the sequential dependencies in the time-series data and fails to meet the expected accuracy of 80%.

Synthetic Minority Oversampling Technique (SMOTE) was applied for balancing [5]. SMOTE is a method of creating simulated samples in the minority group. The main idea is to take two samples from the minority group and create additional samples between them. The new samples are similar to the original ones. However, the results did not improve significantly. Therefore, it was concluded that SMOTE may not be suitable for machine learning, and it was decided not to use it.

Table 2 summarizes the accuracy scores for each method conducted.

Method Description	**LOPO Accuracy**
Logistic Regression with top features selected by Random Forest	0.662
Logistic Regression with PCA-reduced features	0.661
XGBoost using top features selected by Random Forest	0.721
XGBoost using PCA-reduced features	0.723
XGBoost with SMOTE applied to statistical features	0.688
Random Forest with SMOTE applied to statistical features	0.688

Table 2: LOPO Accuracy of all evaluated models

4.4 Deep Learning

Since machine learning methods are not well-suited for time-series data, it is natural to move to deep learning models, which can capture sequential dependencies. Long Short-Term Memory (LSTM) is the main model explored.

LSTM is a type of Recurrent Neural Network (RNN). In addition to traditional RNNs, it uses memory cells to retain information over long sequences. Compared to the machine learning models tried before, LSTM captures patterns over time, making it a good fit for this dataset and problem. Among deep learning models, such as traditional RNNs and Transformers, LSTM was chosen for three reasons. First, LSTMs are proven to be suitable for sequential dependencies in medium-length time series [6]. Second, compared to RNNs, the memory cells in LSTM mitigate the vanishing gradient problem in simple RNNs. Thrid, Transformers require large datasets and have a higher computational cost, which is unnecessary for this small dataset. Gated recurrent units (GRUs) are similar to LSTM, but they are less widely adopted in ecological signal processing.

After reshaping the dataset from 2D (observations $\times$ features) to 3D (samples $\times$ timesteps $\times$ features), a two-layer LSTM with 64 and 32 hidden units, respectively, was trained. The LSTM architecture also included a masking layer to ignore zero-padded timesteps, and a sigmoid-activated output layer for binary classification. Input shape was standardized to the maximum observed length in the dataset timesteps, and padded with zeros for those shorter than that. The model was trained for 10 epochs using the Adam optimizer with a batch size of 2. This means the model was trained 10 times on the training data, with weights updated after processing 2 samples at a time to reduce errors. A small batch size was necessary to accommodate memory constraints during LOPO validation, which requires retaining entire fish sequences. Adaptive Moment Estimation (Adam) optimizer is a stochastic gradient descent method based on adaptive estimation of first-order and second-order moments [7].

It is cross-validated using both LOPO and stratified K-fold methods, with accuracies of 73.8% and 70%, respectively. The results are better than ML models, but not as high as expected. The improvement suggests LSTMs can capture the sequential dependencies when sufficient training data is available. Accuracy was chosen over metrics such as F1-score or AUC-ROC for two reasons. First, the class imbalance was not severe, reducing the urgency of the F1 score. This dataset is limited, but researchers can always collect more data as needed. Second, the two species are not necessarily a True/False problem. The ecological monitoring aims for overall correctness over precision/recall trade-offs. However, future work could include these metrics to assess minority-class performance if imbalance is still a problem.

Method Description	**LOPO Accuracy**	**K-Fold Accuracy**
LSTM with 64/32 hidden units, masking layer, and Adam optimizer	0.738	0.700

Table 3: LSTM Performance Comparison

5 Results & Discussion

The experiments conducted have provided insights for choosing different models to address this fish species classification problem. Traditional machine learning models, such as logistic regression, Random Forest, and XGBoost, are not appropriate. Although XGBoost achieved a somewhat satisfactory accuracy of 72%, it is still unsuitable because it cannot capture sequential dependencies.

Given the need for models that account for sequential dependencies, the deep learning model LSTM was explored. By restructuring the raw 2D frequency data into 3D sequences to preserve temporal continuity, the LSTM model achieved 73.8% accuracy, slightly falling short of the project's target. This model was validated through two methods, LOPO cross-validation and stratified K-fold cross-validation with five folds. These findings confirm that temporal dynamics in hydroacoustic signals are key to accurately classifying fish, but they require more data or model tuning to achieve better results. The original 80% accuracy target, derived from previous work [1], deserves to be examined. While the 73.8% accuracy falls short numerically, the model was trained on higher-quality data (active, no missing response fish) in a smaller dataset. If the dataset could be extended with the same standard, the framework proposed here would outperform the previous framework.

6 Limitations

Despite the success in the accuracy improvement with LSTM, there are several limitations.

Firstly, the dataset is relatively small and the data quality presents challenges. There are only 16 fish, and the two classes are significantly imbalanced. While LSTM is powerful, the small size could result in overfitting, especially considering the parametric complexity. During training, reducing units to 32/16 lowered accuracy, and increasing to 128/64 caused severe overfitting. These results suggest that the LSTM hyperparameters are sensitive. Besides, the computational cost for LSTM is notably high. It requires padded sequences and extensive tuning.

Additionally, the model lacks generalization. The training data and test data are controlled and given, field applicability is untested.

7 Conclusion

The study shows that deep learning models can address the problem of using hydroacoustic frequency responses to classify fish species between LT and SMB. While machine learning models such as XGBoost provided baseline insights, they lack the ability to capture temporal dependencies. In contrast, transitioning to LSTM-based models is an important milestone toward incorporating time-dependent information. Designed explicitly for sequential data, LSTM models achieved accuracies of 70% and 73.8% under stratified cross-validation and LOPO validation, respectively. This achievement highlights the necessity of considering sequential architectures for time-series hydroacoustic analysis.

Further research should focus on enhancing model robustness and generalization. Expanding the dataset to include more active fish would mitigate the overfitting risks. Adding diverse environmental conditions can improve generalization. In addition, integrating Fourier transforms or adopting transformer-based models, could improve accuracy. Finally, field testing in natural habitats is an important validation step before applying to the real world. With these suggestions, the framework developed in this project can be extended to a scalable, non-harmful tool for ecological monitoring.

8 Acknowledgments

The code and results for this project are available at: `https://github.com/blackchocspyyy/FishSpecies/tree/main`.

References

[1] Leivesley, J., & Barajas, L. (2024). *Hydroacoustic fish classification dataset* [Dataset]. GitHub. Retrieved March 2025 from https://github.com/WidebandPingFest/FishTetherExperiment

[2] Breiman, L. (2001). Random Forests. Machine Learning. https://link.springer.com/article/10.1023/A:1010933404324

[3] Jolliffe, I. T. (2016). Principal Component Analysis. Springer. https://link.springer.com/book/10.1007/b98835

[4] Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785–794). ACM. https://doi.org/10.1145/2939672.2939785

[5] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16, 321–357. https://arxiv.org/abs/1106.1813

[6] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. Neural Comput. 9, 8 (November 15, 1997), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

[7] Chollet, F., & Keras Team. (2015). Adam optimizer. Keras. https://keras.io/api/optimizers/adam/