サケ漁獲量予測のための深層学習に関する研究

メタデータ	言語: en 出版者: 公開日: 2023-06-07 キーワード (Ja): キーワード (En): 作成者: チャン, ユエ メールアドレス: 所属:
URL	https://doi.org/10.15118/00010890

Doctoral Dissertation

A Study on Deep Learning for Catch Forecast of Salmon

室蘭工業大学
MURORAN INSTITUTE OF TECHNOLOGY

Muroran Institute of Technology
Division of Information and Electronic Engineering

ZHANG YUE

February 2023

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background of our study

Coastal fishing is an important primary industry in Japan. Hokkaido, an island prefecture in northern Japan, is the largest fishing region in the country, accounting for approximately one-quarter of the coastal fishery products of Japan [1]. However, after peaking in 1984 at 1.282 billion tons, Japanese fishery and aquaculture production declined rapidly and, although this downward trend leveled off after 1995, it has continued to the present day [2]. In this context, accurate catch forecasts can help fishery operators make decisions and perform efficiently. In particular, fishermen and ports can effectively arrange fishing work, and relevant industrial chains such as logistics can be supported to improve logistical efficiency and ensure the freshness of aquatic products [3]. This can greatly relieve pressure on fishery workers. Therefore, catch forecasting is among the most important tasks in the fishery industry [4].

Research on fishery catch forecasting has considered both long-term catch forecasting on a scale of months or years and short-term forecasting on a scale of days. For long-term catch projections, Komiya studied annual catch changes in spiny lobsters and forecasting issues [5]. Leathwick et al. used data on water temperature and salt concentrations with a decision tree to forecast snapper catches and fishing probabilities off the coast over short periods of time [6]. Moreover, Kokaki et al. developed a fish catch forecasting method using a state-space model that described the probabilistic behavior of fish inside nets [7]. Long-term catch forecasting can reflect macrolevel trend changes in the catch, whereas short-term catch forecasting can reflect specific changes in the catch in detail. Therefore, a combination of long-term and short-term forecasts can comprehensively reflect changes in fish catches, and the two approaches can confirm and complement each other to simulate and summarize the patterns to the greatest extent. Particularly, if changes over different time scales are combined with changes over different geographical scales, macro- and microlevel changes in fishery catches can be demonstrated more effectively. However, in existing works, long-term and short-term catch forecasting studies have often been conducted independently. Hence, analyzing and studying catch data from multiple perspectives remains challenging.

Moreover, the fishery industry is subject to problems involving limitations of available data in practical applications. In fact, in Japan, particularly in Hokkaido, numerous challenges arise in catch data collection and collation. Due to work cycles or actual work conditions, fishery practitioners responsible for catch data often do not have complete and detailed records, leading to a lack of real data and inaccuracies. This situation is particularly severe in the case of short-term catch data, for which it is not easy to achieve the desired forecasting by relying only on the analysis a single catch data type.

1.2 Purpose of our study

To solve these problems, it is necessary to integrate long-term and short-term catch data to develop a more comprehensive analytical approach. To this end, performed data selection and considered the data management approach of the fishery industry in the Hokkaido region of Japan (Fig. 1.1). From the figure, it may be observed that catch data are managed in the Hokkaido region via a cascading accounting process, in which data are first collected by individual fishing vessels or ports, then aggregated to the regional fishing associations, and finally pooled from each region to construct the total catch for the Hokkaido region. This method of data management creates a difference between data recorded at high and low levels. Because the high-level data representing a large scope is composed of lower-level data representing a small scope, the statistical period for aggregating the data is longer for high-level data than for lower-level data. In other words, high-level data are generally counted only when low-level data are accumulated to a certain extent. Consequently, long-term data tend to correspond to a wide range of geographical data records. By contrast, short-term data tend to have a relatively small geographical scope.

To the best of our knowledge, the present work is the first to integrate long- and short-term data processing for fishery data. The proposed approach is implemented in the aforementioned forecasting study on the catch in Hokkaido. Monthly catch data for the entire Hokkaido province and daily catch data for the eastern ports of Hokkaido were used.

The objectives of this doctor thesis are mainly two. The first objective is to construct a suitable forecasting method based on the distribution characteristics of the catch volume data on different time scales(Fig. 1.2). The second objective is to combine long- and short-term catch volume forecasting to provide data support for different usage scenarios. Therefore, in Chapters 3 and 4, constructs forecasting models for short-term and long-term catch data, respectively. In Chapter 5, upgrades the forecasting method in Chapter 3, and also combine the long- and short-term catch forecasts.

Specifically, in Chapter 3, proposed an LSTM network for forecasting short-term catches, and add a method related to data augmentation, aiming to solve the gradient disappearance/explosion problem of LSTM networks. In Chapter 4, a model based on neural network and autoregressive integrated moving average (ARIMA) [8] is proposed to handle both short-term characteristics and long-term catch data, and is used to forecasting long-term catch data. The LSTM method with data augmentation proposed in Chapter 3 has achieved some results, but still completely fails to solve the gradient disappearance/explosion problem. Together with the single LSTM structure, there are also limitations in its overall predictive capability. Therefore, in Chapter 5, combines CNN and LSTM models to construct a neural network model with a data splitter for predicting short-term catch data. The multivariate data features are extracted by CNN, the LSTM captures the time dependence, and the data splitter structure is able to extract both long-term and short-term features from the data.

1.3 Structure of this doctor thesis

This doctor thesis consists of six chapters, and the outline of each chapter is described below. Chapter 1 describes the background and purpose of the study. Chapter 2 introduces previous research on capture prediction as a related study to this research and clarifies the problem to be addressed in this doctor thesis. In Chapter 3, proposed an LSTM-based method with a combination of data augmentation and data filtering, and the results show that it improves the prediction accuracy. In Chapter 4, proposed a

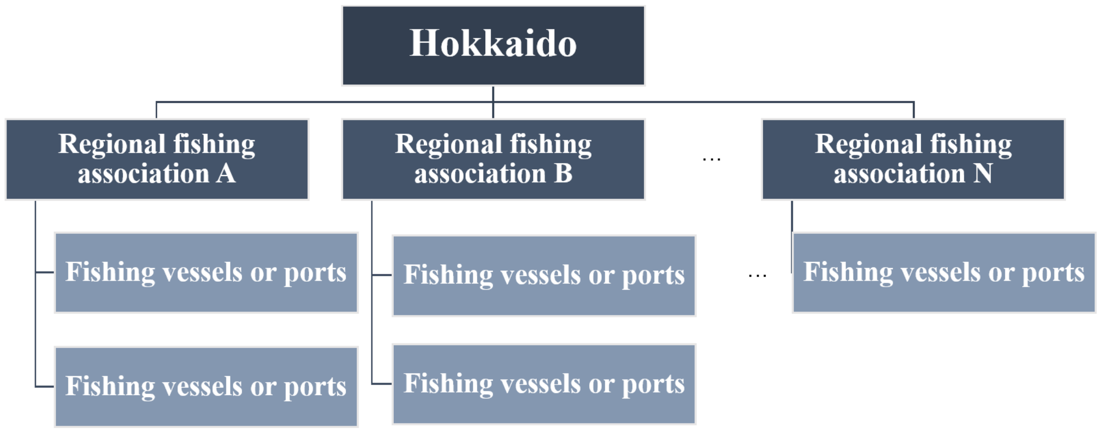


Figure 1.1: Data management approach of fishery industry in the Hokkaido.

prediction method combining ARIMA and LSTM for the data characteristics of long-term capture and water temperature. and show that high accuracy prediction can be achieved by this method. In chapter 5, as a combined long- and short-term study, proposed and implement a neural network model with multiple time-scale features based on convolutional neural networks and long- and short-term memory neural networks for predicting daily catches. Similarly, a previously proposed method for predicting long-term catches was combined to investigate the total monthly catches in Hokkaido. The daily and monthly prediction results and their correlations were also analyzed in detail. Therefore, the prediction of catch from multiple time scales using deep learning models. Not only does it have high prediction accuracy, but it is also more realistic. The In chapter 6r presents the conclusions of this doctor thesis and discusses future issues.

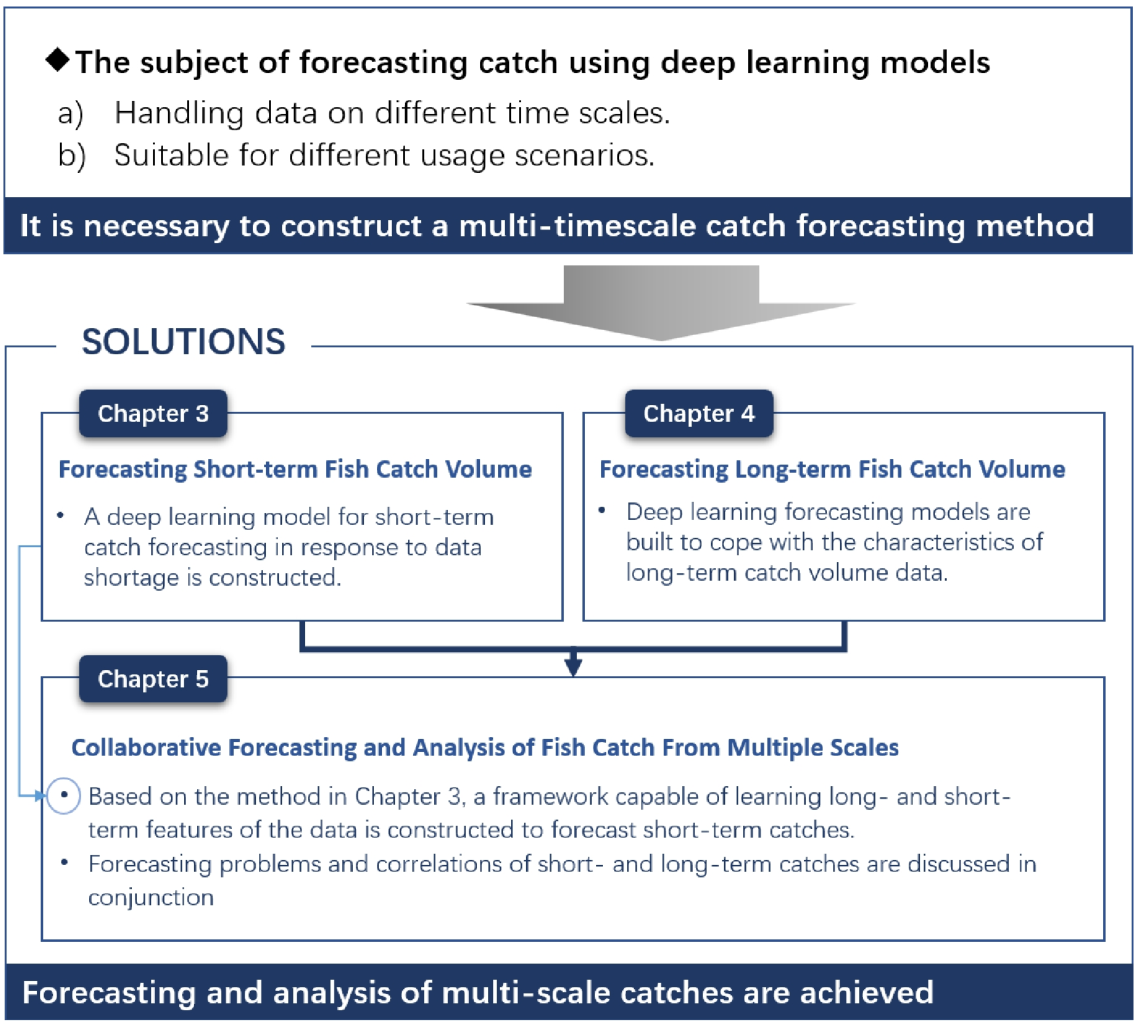


Figure 1.2: Our process summary.

Chapter 2

Related works

2.1 Introduction

In this section, described previous related research on catch forecasting. Fig. 2.1 illustrates the distribution of existing capture prediction studies. Could be seen that before 2000, the capture prediction methods were mainly traditional based on probability or constructing environmental models, and the prediction results for complex data were not satisfactory. In contrast, the latest studies between 2018 and 2022 have typically used machine learning methods with high prediction accuracy.

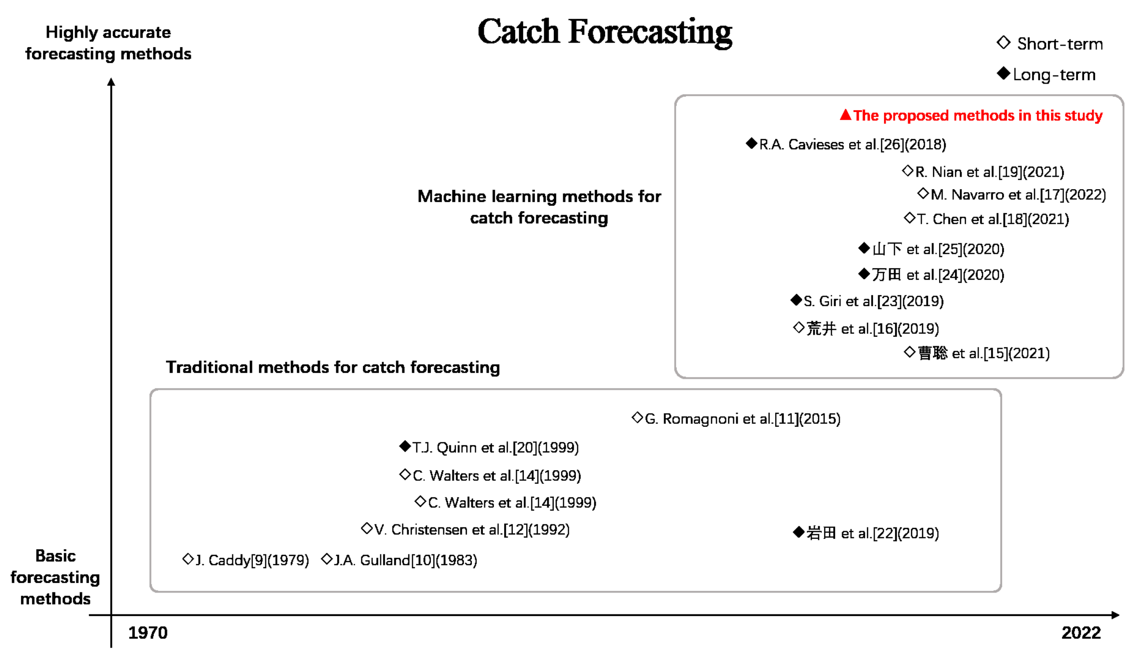


Figure 2.1: Resarch map.

In the following, described previous studies on capture prediction, which are closely related to this doctor thesis. The structure of this chapter is shown in the following figure. **2.2** describes existing studies that address short-term capture prediction. **2.3** describes the existing studies for long-term catch prediction. **2.4** describes the research on deep learning for time series data prediction so far. In **2.5**, shows the problems that can be considered from the previous studies and the solutions in this doctor thesis.

2.2 Related works of short-term catch projections

To date, various methods have been proposed to predict short-term captures [9–11, 15–19]. Among them, in literature [9–11], traditional methods such as constructing environmental models are mainly used to predict captures. In contrast, in literature [15–19], a data-driven machine learning approach is used to predict the captures.

In the following, explains the methods that have been studied so far for short-term capture prediction.

literature [9]

In the literature [9], in the estimation of fish catchability of invertebrate types, factors such as consideration of fishing gear inputs and work intensity were introduced.

literature [10]

The literature [10], similarly to [9], a consideration of the use of fishing gear within the entire sea area of the fishing ground is also included in the calculation of catch probability.

literature [11]

The literature [11], an eco spatial model based on Ecopath with Ecosim (EwE) [12–

14] food-web models was used to predict the temporal-spatial relationship between fish biomass and catch.

literature [15]

In the literature [15], based on the basis of a collection of fishes, sensors and meteorological numbers, resnet regressor and random forest regressor two types of machine learning models, used for daily measurement and each fish type.

literature [16]

In the literature [16], using satellite data, the maximum entropy of random forest, support vector machine, and machine learning is used to predict the catch of fallfish.

literature [17]

In the literature [17], spatial heterogeneity was considered in the problem of recreational fishery catch prediction, and two models, generalized additive mixture model (GAMM) and augmented regression tree (BRT), were used for comparative analysis.

literature [18]

In the literature [18], using a multi-layer perceptron (MLP), catch prediction is performed from all available data collected by fishing vessel sensors.

literature [19]

In the literature [19], short-term catches were predicted using convolutional neural networks. A new kind of knowledge, the trawler fishing chronology, is added to address the challenges posed by the stochastic nature of fishermen's behavior and the diversity of ocean weather.

2.3 Related works of long-term catch projections

To date, various methods have been proposed to predict long-term captures [20, 22–26]. Among them, in literature [20, 22], traditional methods such as constructing environmental models are mainly used to predict captures. In contrast, in literature [23–26], a data-driven machine learning approach is used to predict the captures.

In the following, explains the methods that have been studied so far for long-term capture prediction.

literature [20]

In the literature [20], predictions of annual fish stocks are made using production and yield per unit effort (CPUE). In addition, CPUE is the value obtained by dividing Catch by Effort, but in order to estimate CPUE, temporal and geographic effort data and catch data will result in increased precision [21].

literature [22]

In the literature [22], under the condition that only catch information is available, the catch probability is used to reduce the bias and thus estimate the mean annual catch.

literature [23]

In the literature [23], the monthly catches in the Gulf of Bengal northern Khailsa Bay were modeled and predicted using Bayesian structural time series (BSTS) with chlorophyll and rainfall as explanatory covariates.

literature [24]

In the literature [24], five statistical models, linear regression, support vector regression, random forest regression, neural networks, and deep learning, were tested using extensive data on weather, ocean, and other fish catches.

literature [25]

In the literature [25], the annual and monthly catches of skipjack tuna in the waters around Kochi Prefecture were predicted using a regression algorithm.

literature [26]

In the literature [26], monthly catch projections were made for the finfish fishery in the ”Baa Magadalena-Almejas” area. Both non-linear autoregressive models and long and short term memory neural networks were used for prediction and comparison.

2.4 Related works of forecasting methods for time series data

A time series is generally defined as an ordered set formed by arranging the values of the same statistical indicators based on the order of their occurrence. Time series prediction has an important application value and is one of the popular research topics in recent years [27] Hydrological data is exactly the typical time series data. Existing time series forecasting techniques can be summarized into two main categories: statistical methods and machine learning methods [27], [28]. Statistical methods represented by AutoregressiveIntegrated Moving Average (ARIMA) models [28] are suitable for linear system forecasting, and machine learning methods represented by artificial neural networks are suitable for complex nonlinear time series.

However, there is often a mixture of short-term and long-term patterns in real data, especially for the data, which are highly cyclical, where long-term features reflect long-term changes between seasons, between years, etc.; short-term features reflect changes in the effects of weather, precipitation, etc., etc. Again, accurate time series forecasting is not possible without considering both cyclical patterns.

However, traditional methods (e.g., the extensive work in autoregressive methods) [28–30] are inadequate in this respect, since most of them do not distinguish between the two modes and do not model their interactions intuitively and dynamically. Therefore try to address this problem using a machine learning approach.

Hochreiter et al. proposed Long Short-Term Memory Network (LSTM), a machine learning method based on recurrent neural network RNN, with certain information mining capability for long-range temporal data, which is widely used in speech recognition, machine translation, fault prediction and load prediction [31] and other fields. Some scholars have even fused Convolution NeuralNetwork (CNN) and LSTM for behavior prediction and speech recognition [32, 33]. The common idea of these methods is to extract high-dimensional features with the short sequence feature abstraction ability of CNN, and then LSTM synthesizes the short sequence high-dimensional features for temporal prediction, which is suitable for processing temporal data with local relevance.

2.5 Problems to be solved in this doctor thesis and solutions

In the previous three sections, Discussed the existing methods for capturing long-term as well as short-term forecasts and common methods for time series data forecasting. And in these prior studies, captured two problems and tried to solve them.

First, there is the problem of prediction for long-run as well as short-run captures. In practice, long-term catch forecasts often represent overall catch trends and are often used by practitioners to determine future catch trends in an area or by management agencies such as fish associations as a reference for resource allocation. Short-term catch forecasts, on the other hand, provide a more detailed picture of catch variability and can be used as a guide for port or vessel scheduling and staffing levels. Of course, there are rich examples of existing studies discussing the problem of predicting capture volume, both in traditional capture volume prediction methods (based on environmental models, etc.) and in machine learning methods. However, in the existing studies, only long-term or short-term catch prediction has been discussed, and there is no case where researchers have considered long-term and short-term catch prediction together. Moreover, the motivation for using long-term or short-term capture data is not clearly stated in these studies, and the choice of data often depends only on the type of data available, without clarifying the relevance of long-term and short-term capture data. For example, in some studies, long-term capture data were selected for studies that required detailed variation, while short-term capture data were used for work that needed extensive data guidance. Therefore, in this study, not only discuss the problem of long-term and short-term capture prediction in an integrated manner but also match the long-term and short-term capture prediction problems with wide-range and limited-range data in the spatial dimension, respectively, for different usage scenarios.

Second, in terms of forecasting methods, various forecasting models have been used in existing studies to forecast the capture volume though. However, it is clear that most of the methods used are still focused on autoregressive or recurrent neural network type methods, and combined with the discussion in Chapter **2.4**, could be seen that the methods used do not perfectly solve the problems in the prediction of capture volume, and there are still limitations in their use. Moreover, there is no intentional selection of suitable prediction methods for data characteristics of long-term or short-term data. In this regard, a special deep learning framework is used in this doctor thesis to cope with the diverse and nonlinear forms of short-term data variation in the short-term capture prediction problem: it uses a convolutional layer to extract high-dimensional features, a recurrent layer to capture complex long-term dependency patterns, and a novel structure to jump the recurrent layer to capture ultra-long-term dependencies, allowing the network to better learn the periodicity of the input time series data. . Finally, the traditional autoregressive linear model is concatenated with a nonlinear neural network component to make the nonlinear deep learning model more robust to scale-violating time series. And for long-term capture volume data, for its sparser data and distinct linear characteristics. We propose a hybrid model based on autoregressive and LSTM to process the catch and water temperature data, so as to accomplish the long-term catch prediction with high accuracy.

2.6 Summary

In this chapter, explains previous research from the capture prediction studies relevant to this doctor thesis. In addition, described the problems of previous studies and clarify the issues to be addressed in this doctor thesis to enhance the accuracy and application implications of capture prediction.

Chapter 3

Predicting short-term Port-Catch Volume Using LSTM

3.1 Introduction

In this chapter, will describe one of the deep learning methods Long short-term memory (LSTM), and implement the work on predicting port captures in the presence of data limitations. For this purpose, constructs a data augmentation and data filtering method. LSTM is a special kind of RNN, mainly to solve the gradient disappearance and gradient explosion problems during training long sequences. Compared with other networks, LSTM can handle good time series data and capture long-term dependencies, so it can be expected to improve the accuracy of estimated captures by introducing it into this study.

Thus, in this doctor thesis, described a capture prediction method using an LSTM-based model. Specifically, first, the available data are expanded using a Gaussian distribution in a way that simulates fish coming to swim in nature. Next, the catch information of the rest days was removed according to the work schedule habits in Hokkaido region, and the interference of other factors was reduced. As mentioned above, the present method allows high precision estimation of harbor catches using a small amount of data.

3.2 LSTM model

The recurrent neural networks (RNNs) have been used for training the time series data. During the gradient back-propagation phase, the gradient signal can end up being multiplied a large number of times (as many as the number of time steps) by the weight matrix associated with the connections between the neurons of the recurrent hidden layer. This means that, the magnitude of weights in the transition matrix can have a strong impact on the learning process. They are networks with loops in them, allowing information to persist. The structure had be shown in Fig. 3.1.

According toFig. 3.1, X is the input vector, and S is the hidden layer vector. U, V and W are weight matrices. O is the output vector. The input and output are represented by the following equations, respectively.

$$S_t = g(V * X_t), \tag{3.1}$$

$$O_t = f(U * S_t + W * S_{(t-1)}), \tag{3.2}$$

where g and f are activation functions. As concerning the general RNN, the gradient after many stages of propagation tends to disappear (most of the time) or explode (rarely, but it has a great impact on the optimization process). This is called Problem of Long-Term Dependencies.

These issues are the main motivation behind the LSTM (Long Short-Term Memory) model which introduces a new structure called a memory cell (see Fig. 5.3 below). A memory cell is composed of three main elements: an input gate, a forget gate and

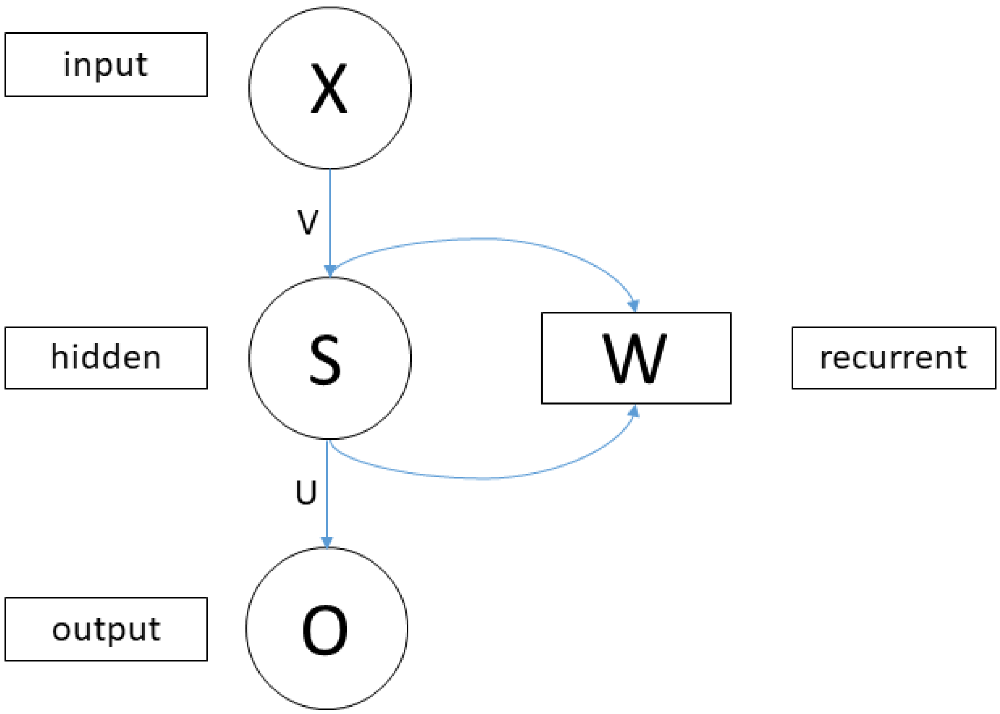


Figure 3.1: Structure of the recurrent neural networks

an output gate. The gates serve to modulate the interactions between the memory cell itself and its environment. Input gate controls the extent to which a new value flows into the memory. Forget gate controls the extent to which a value remains in memory. And output gate controls the extent to which the value in memory is used to compute the output activation of the block.

The equations below describe how a layer of memory cells is updated at every time step t. In these equations: x_t is the input to the memory cell layer at time t; W_i, W_f, W_c, W_o, U_i, U_f, U_c, U_o and V_o are weight matrices; b_i, b_f, b_c and b_o are bias vectors.

First, computes the values for i_t, the input gate, and $\widetilde{C}_t$ the candidate value for the

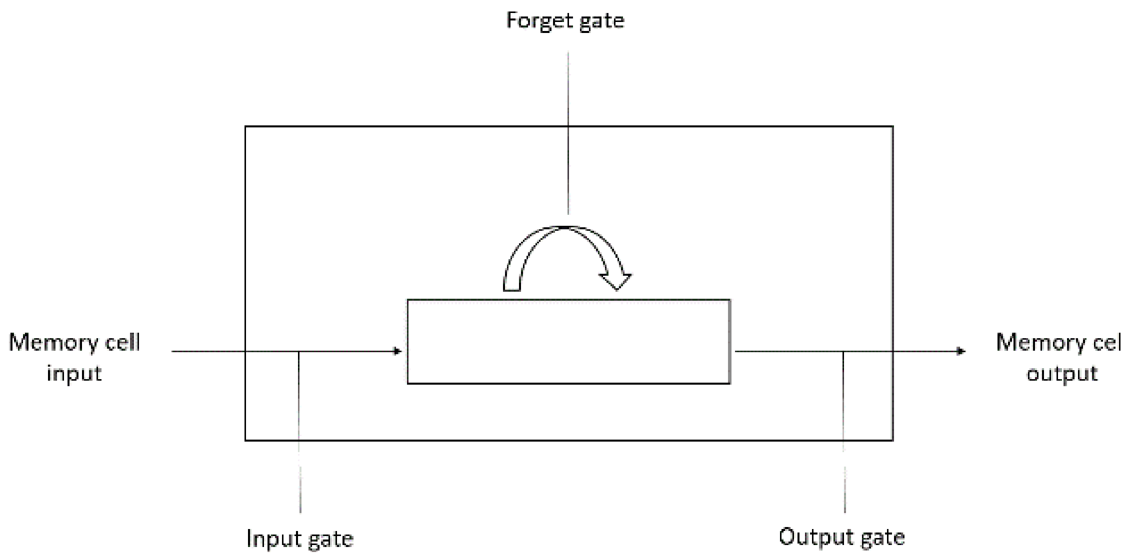


Figure 3.2: Structure of the Long Short-Term Memory networks.

states of the memory cells at time t, in Eqs. (3.3) and Eqs. (3.4):

$$i_t = \sigma(W_I x_t + U_i h_{(t-1)} + b_i), \tag{3.3}$$

$$\widetilde{C}_t = tanh(W_c x_t + U_c h_{(t-1)} + b_c), \tag{3.4}$$

Second, computes the value for f_t, the activation of the memory cells' forget gates at time t, in Eqs. (3.5):

$$f_t = \sigma(W_f x_t + U_f h_{(t-1)} + b_f), \tag{3.5}$$

Given the value of the input gate activation i_t, the forget gate activation f_t and the candidate state value $\widetilde{C}_t$, obtains C_t by the memory cells' new state at time t, in Eqs. (3.6):

$$C_t = i_t * \widetilde{C}_t + f_t * C_{(t-1)}, \tag{3.6}$$

With the new state of the memory cells, obtains the value of their output gates and, subsequently, their outputs as shown in Eqs. (3.7) and Eqs. (3.8):

$$o_t = \sigma(W_o x_t + U_o h_{(t-1)} + V_o C_t + b_o), \tag{3.7}$$

$$h_t = o_t \odot \tanh(c_t), \tag{3.8}$$

3.3 Data augmentation and filtering

3.3.1 Extended data based on Poisson distribution

In terms of the amount of catch, since the fish stocks come to the fishing port in line with the queuing model, it can be seen as a queuing problem. So used the Poisson distribution model to simulate the number of fish caught, in order to increase the amount of data as noise data.

First of all, added Poisson noise data as dummy data in the training set to expand the training set. In this doctor thesis, used Poisson model to generate extended data in the training set to expand the training set. In the training set, we extract the data of port-catch volume, next suppose a fish has a weight of $w = 3kg$ and treat k as the number of fishes in one group of fish. First, the port-catch volume (tons) is converted into the number of the shoals of fish, and the data of the shoals of fish is brought into the Poisson distribution to generate extended data. Finally, restore the data to tons. The reason for changing the tonnage of the fish to the shoals of fish is: First, increasing the λ of the Poisson distribution will make the fluctuation of the Poisson distribution smaller, thereby reducing the error and making the artificial data closer to the real data. Secondly, this way of data processing also simulates the form of motion of fish in nature. Regarding the setting of the coefficient n (the number of fish in each group), since the data scale of each fishing port is different, the size of the fish school is also different. Therefore, calculated the average value of the catch data of each fishing port as a benchmark for k value, and observed the effect of the change in k value on the experimental error to find the most suitable value of k. (likes Fig. 3.3 and Fig. 3.4).

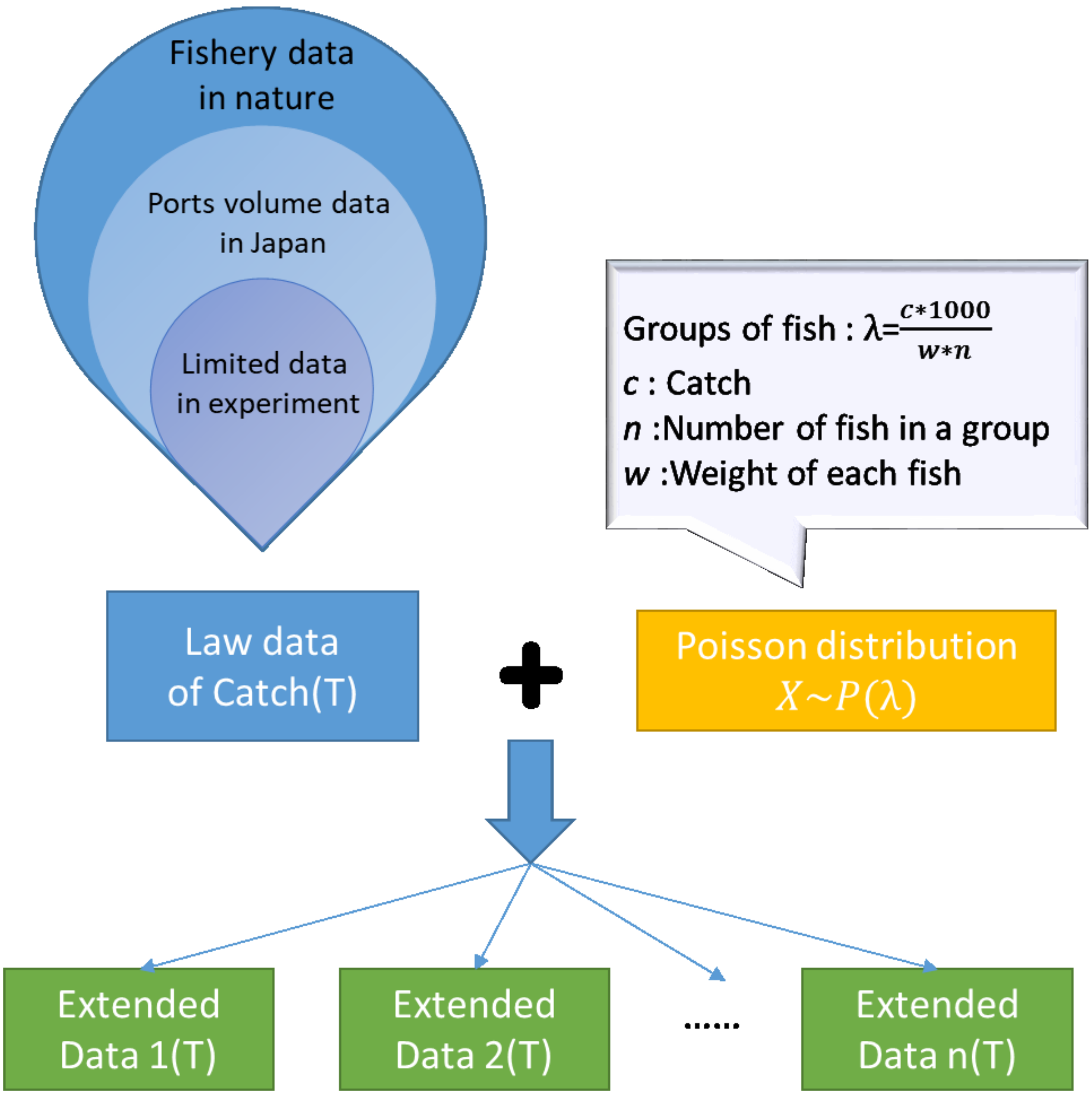


Figure 3.3: Extended data based on Poisson distribution.

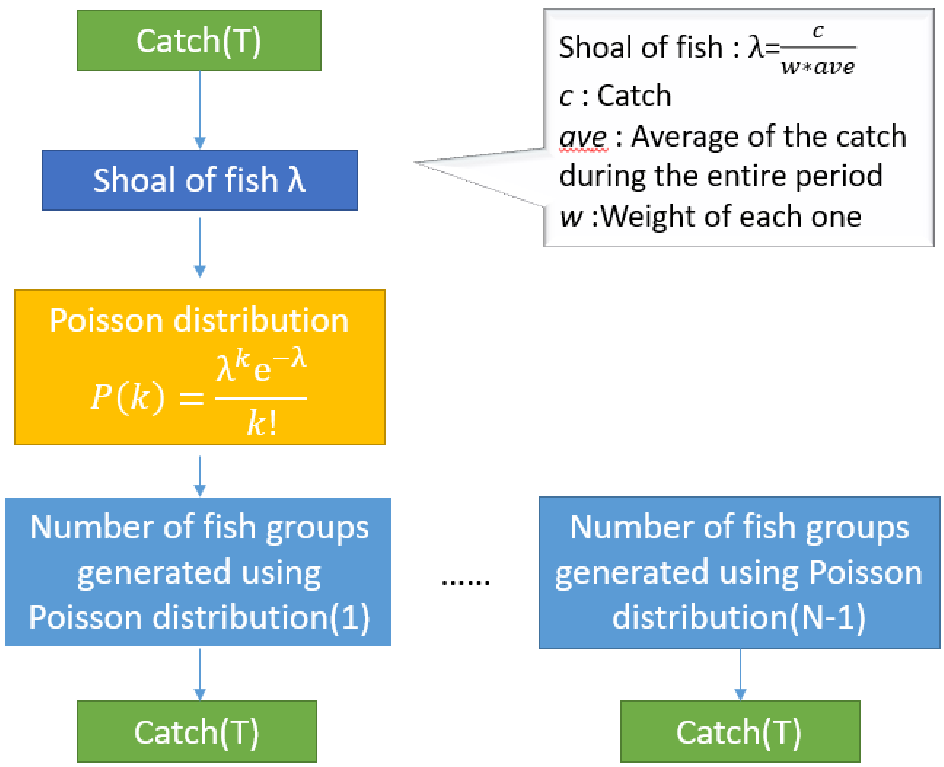


Figure 3.4: Generate Noise Data of Catch.

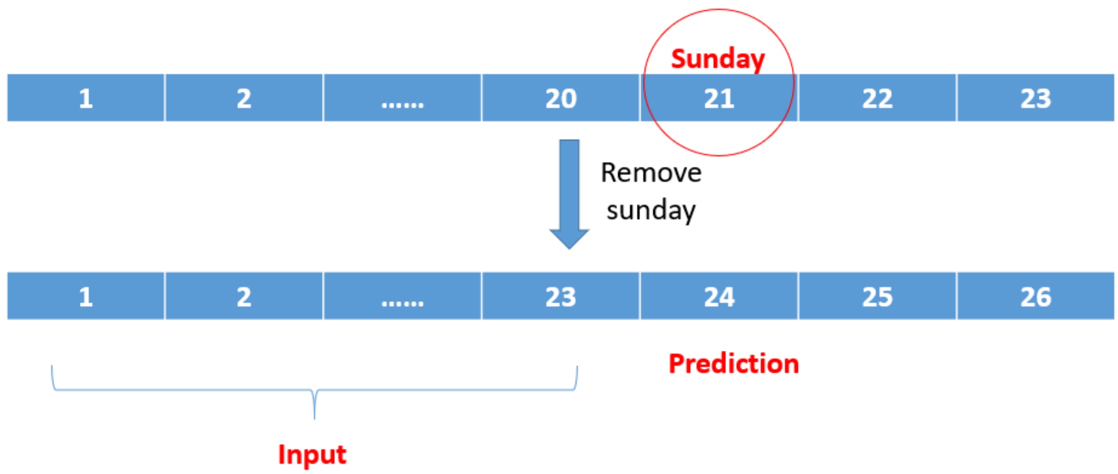


Figure 3.5: Remove Sunday to Develop the Prediction Performance.

3.3.2 Remove rest days (Sunday)

In the actual situation of the port operation, found that it is not only the weather, water temperature and other objective factors that affect the catch. Human subjective factors still have a great influence on the variation of catch. It is obvious that in the case of a port break, even if a certain amount of catch could have been obtained, the catch of the day would have been zero. It is this subjective variation that brings a negative impact on the prediction of catch.

In the catch data of the fishing ports, since Sunday is the day of absolute rest, the catch on Sunday is 0. So, likes the Fig. 3.5 treats these 0 data as meaningless data and remove them for learning and prediction.

3.4 Experiment

In this section, to confirm the validity of the proposed catch estimation method, conducted experiments on catch estimation using data from four ports in the eastern region of Hokkaido. First, section **3.4.1** describes the data set, the evaluation method, and the comparison method as the experimental conditions. Next, in **3.4.2**, presents and analyze the results on the experiments.

3.4.1 Experimental conditions

This experiment used the fishery catch in Nemuro, Ochiishi, Habomai and Rausu 4 ports from 2005 to 2015, June 1st to November 30th. For water temperature data, used satellite data from the NEAR-GOOS website and the same time interval (50m temperature). The relationship of Fishing port and Temperature had been shown in Table 1.The data of the first 20 days from June 1st were used to predict the catch value for the next day that likes Fig. 3.6.

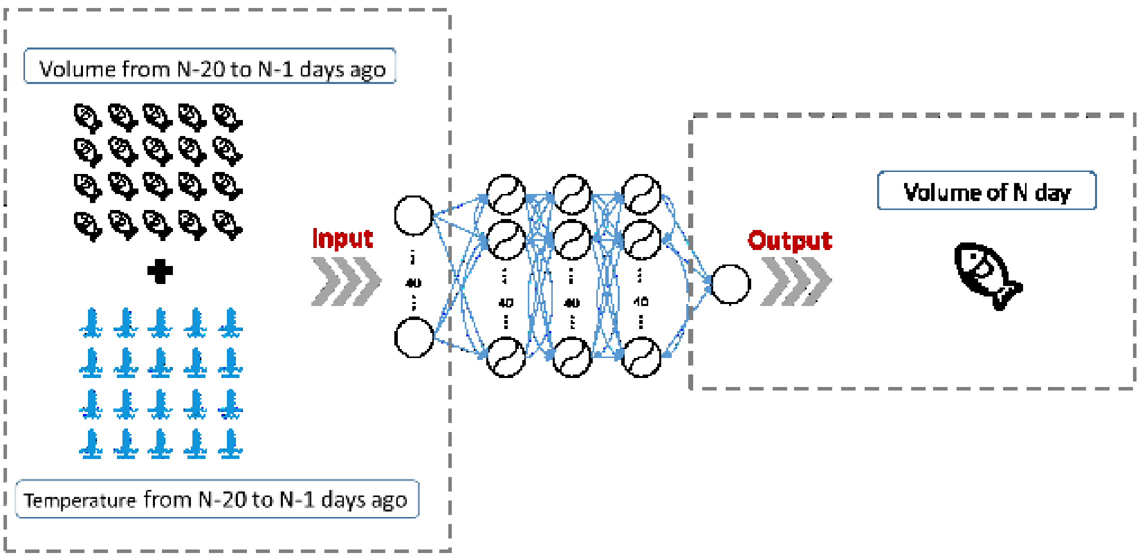


Figure 3.6: Prediction of fishery volume using LSTM.

Table 3.1: Details of parameters for the proposed method.

Parameters	Value
Epoch	111
Optimizer	6
Training rate	0.01
Activation function	Sigmoid, tanh
Loss	RMSE

In this experiment, used the data of the four fishery for the period of 2005-2015. First, the 2005-2014 data is used as a training set, and the 2015 data is used as a test set. First, let's use the first method that said before to process the raw data: Adding noise data to increase the data set of 2005-2010 by 9 times, then make 10 times the training set together with the original data, and the test set is unchanged. Experiment and see the results. Then use the second method of removing Sunday to process the raw data. Experiment and view the results. Finally, compare the experimental results of the original data with the experimental results of processing the data by the above two methods.

Table 3.1 shows the list of parameters used in the proposed methods. However, the parameters of the proposed and compared methods are determined in order to maximize the performance of each method.

To show the effectiveness of the proposed method, a set of ablation experiments was designed to compare the variability of the results. Specifically, used a simple LSTM model and an LSTM model incorporating data augmentation methods and data filtering methods, respectively, to make predictions of port captures. And their prediction results were compared by RMSE coefficients.

3.4.2 Experimental results

According to the experimental results, used the Root Mean Square Error (RMSE) to evaluate the experimental results. Comparing the RMSE in different situations and different training sets in Fig. 3.7, could be seen that:

1) Increases the training set by the method of simulating real data through Poisson distribution. It does reduce the over-fitting problem of LSTM to a certain extent and reduces the test error.

2) Sunday ' s fishing data can be seen as a human interference behavior or a law. The error generated by this part of the data accounts for a large part of the total test error. After removing the data on Sunday, the learning difficulty for the neural network is reduced, and the test error is also greatly reduced.

The experimental result of Nemuro had been shown in Fig. 3.8. The experimental result of Ochiishi had been shown in Fig. 3.9. The experimental result of Habomai had been shown in Fig. 3.10. The experimental result of Rausu had been shown in Fig. 3.11.

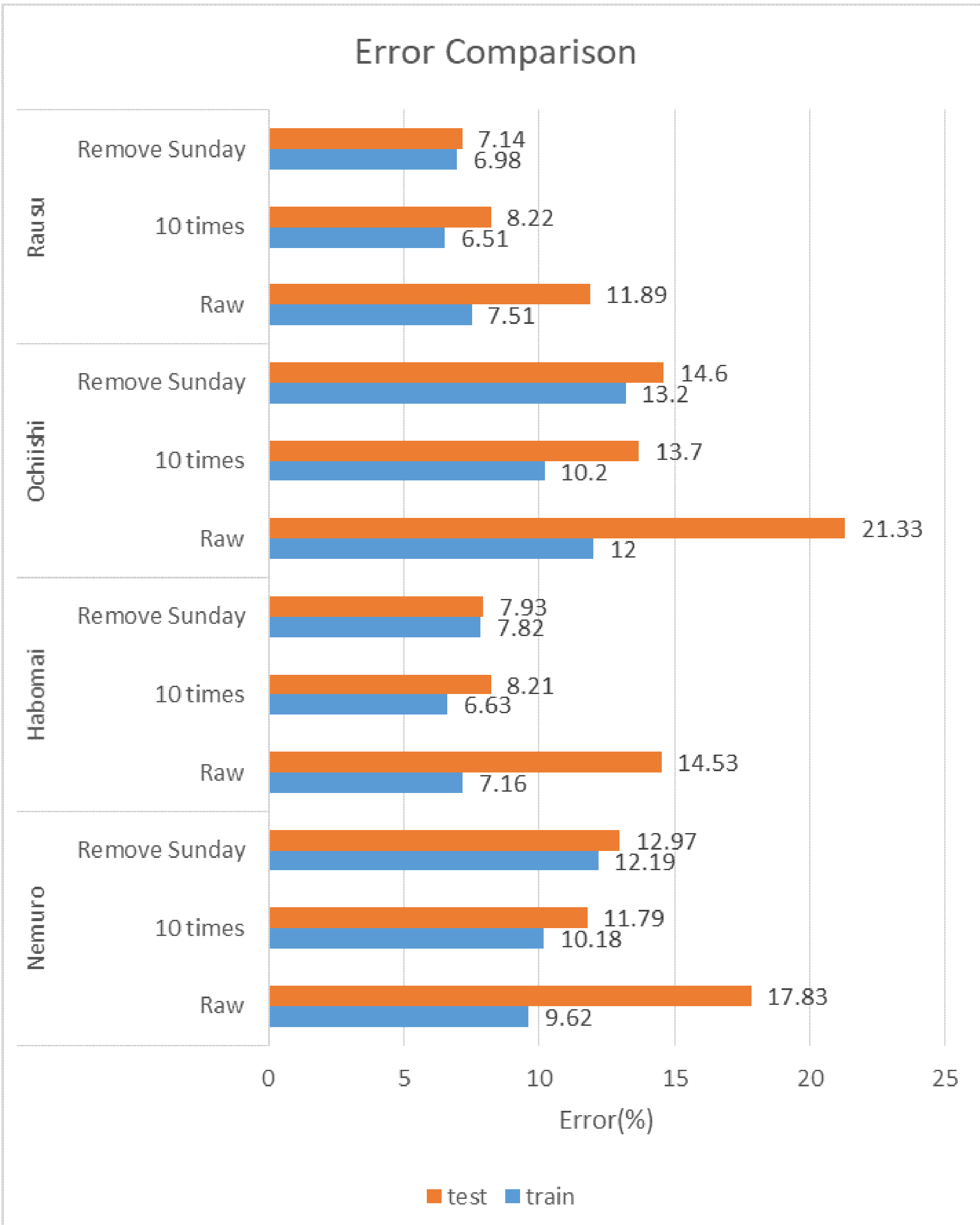


Figure 3.7: Error of Three Situations.

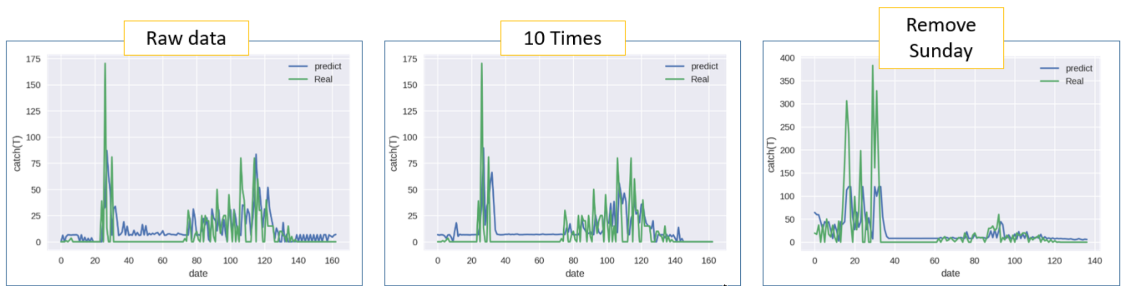


Figure 3.8: Prediction result of Nemuro.

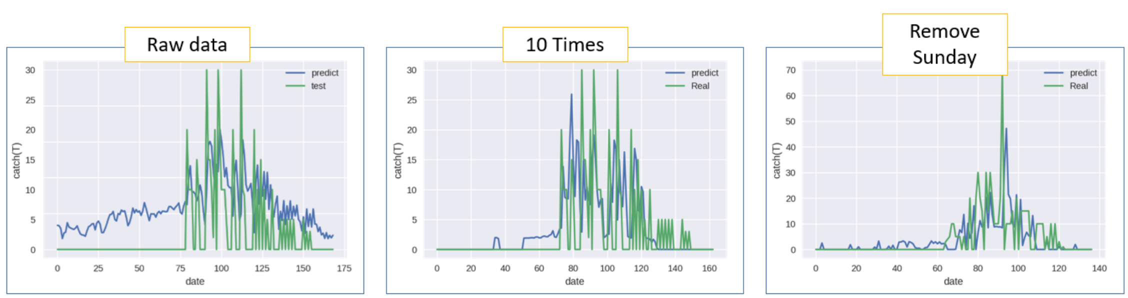


Figure 3.9: Prediction result of Ochiishi.

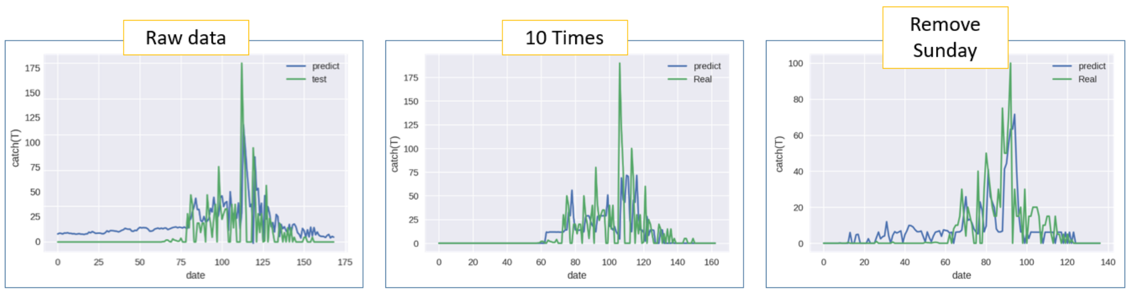


Figure 3.10: Prediction result of Habomai.

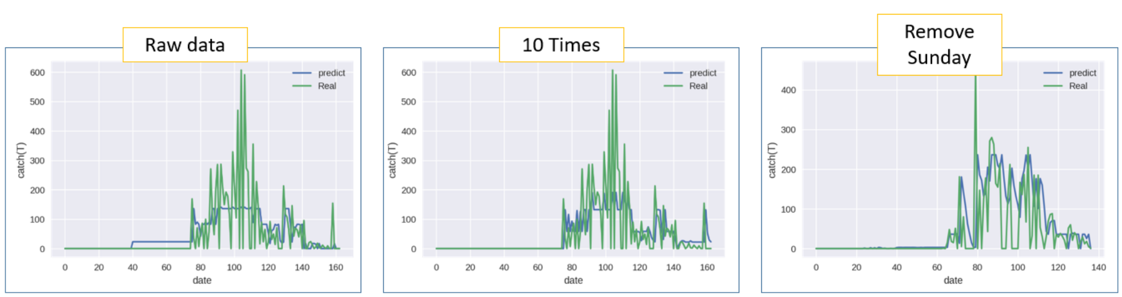


Figure 3.11: Prediction result of Rausu.

3.5 Summary

For the experimental results, can see that it ' s feasible to use the LSTM model in RNN to predict the fishery catch data. And the method for the original data also have a certain effect. Especially in the case of the sparse data, our data processing procedure can improve the prediction accuracy. However, the more effective ways to reduce the prediction error are continuously required. Through the prediction of the fishery port-volume data and the efforts made to reduce the prediction error, the forecast of fishing port capture to support the fishery-logistics transportation will be developed in our future work.

Chapter 4

Predicting long-term Port-Catch Volume Using Using ARIMA

4.1 Introduction

In the wide range of catch forecasting, the data volume limitation owing to the large time unit does not satisfy the most favorable conditions for neural network models, which characteristically require large amounts of data to perform training. However, these wide-range, long-time unit data, which can represent the trend changes in the catch on a macroscopic scale, are more linear and thus more suitable for processing by traditional statistical methods [34]. Therefore, used the ARIMA model to forecast the monthly catch for the entire Hokkaido area.

The specific process of the proposed method is described in Chapter **4.2**. In Chapter **4.3**, described the relevant settings and results of the experiments. The experimental results show that our proposed method can effectively predict the capture at long-term scales.

4.2 ARIMA and LSTM based prediction model

In this section, discussed in detail the ARIMA algorithm in **4.2.1** And in **4.2.2**, implemented the consideration of water temperature features based on the captured volume data in a way that combines LSTM and ARIMA to achieve higher accuracy prediction.

4.2.1 ARIMA model

Box and Jenkins introduced the ARIMA model in 1970. Also referred to as the Box —Jenkins methodology, ARIMA is composed of a set of activities for identifying, estimating, and diagnosing ARIMA models with time-series data.

The ARIMA model has three parameters, as indicated in Eqs. (4.1): p, d, and q. p represents the number of lags used in the forecasting model itself and is also known as the autoregressive (AR) term [35]. d refers to the time-series data that require several orders of differential differentiation to be stable and is also known as the integrated term. q represents the number of lags of the forecasting errors used in the forecasting model and is also known as the moving average (MA) term [36]. It is expressed as follows.

$$\hat{y}_t = \mu + \varphi_1 y_{t-1} + \cdots + \varphi_p y_{t-p} + \theta_1 e_{t-1} + \cdots + \theta_p e_{t-p}, \tag{4.1}$$

where $\varphi_\bullet$ denotes the coefficient of AR and $\theta_\bullet$ denotes the coefficient of MA. In contrast to the daily catch forecast, used the daily sea temperature over the month to forecast the monthly catch in the wide-area catch forecasting.

4.2.2 ARIMA and LSTM based model

Seawater temperature has an unavoidable influence on the capture forecasting problem, but the nature of the ARIMA-based autoregression model leads to an inability to resolve other characteristic variables. Therefore, to enable the model to learn the characteristic information resulting from the water temperature data, included an LSTM layer in ARIMA to resolve the monthly water temperature data and to enhance the forecasting accuracy, with the structure depicted in Fig. 4.1.

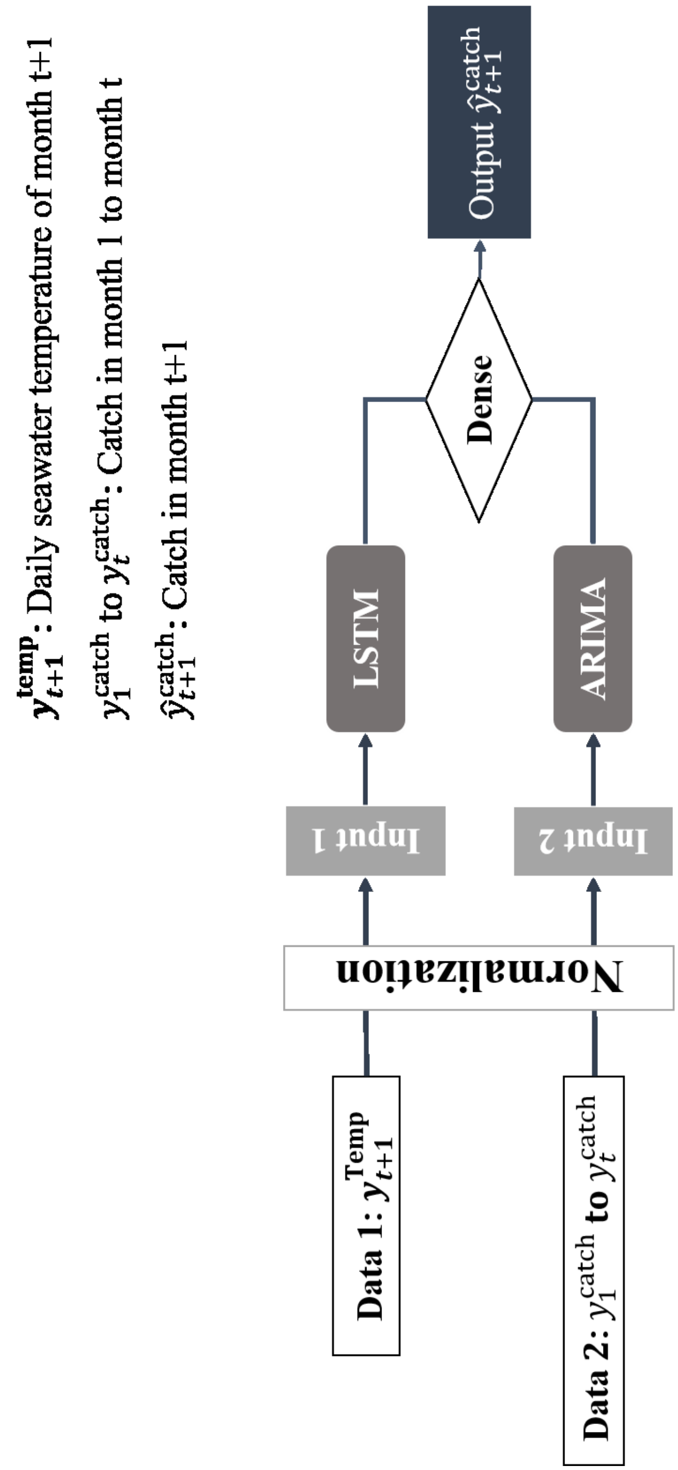


Figure 4.1: Model structure of ARIMA.

First, extracts the water temperature and fish catch data as two inputs. For example, to forecast the catch data of month $t+1$, used the catch data of the previous month 1 to month t and the daily seawater temperature data of month $t+1$.

The data normalization process has often been required for the use of the ARIMA model. Then, substitutes $(y_t^{\text{catch}} - y_{\text{min}}^{\text{catch}})/(y_{\text{max}}^{\text{catch}} - y_{\text{min}}^{\text{catch}})$ with y_t^{catch} as the data normalization. Also, substitutes $(y_t^{\text{temp}} - y_{\text{min}}^{\text{temp}})/(y_{\text{max}}^{\text{temp}} - y_{\text{min}}^{\text{temp}})$ with y_t^{temp}. Subsequently, inputs the water temperature and fish catch data into the LSTM and ARIMA, respectively, and use two algorithms to process the two data separately. Finally, used a fully connected layer to integrate the outputs of the two networks as the final output $\hat{y}_{t+1}^{\text{catch}}$.

4.3 Experiments

In this section, to confirm the validity of the proposed long-term catch prediction method, conducted a prediction experiment using monthly catches in the Hokkaido region. **4.3.1** describes the experimental conditions, and **??** describes the experimental results and discussion.

4.3.1 Experimental conditions

Used data from Fisheries Research Institute and NEAR-GOOS. Specifically, the monthly Hokkaido catch data from the Fisheries Research Institute and the corresponding surface water temperature data (from NEAR-GOOS) are used as wide-area data. Selected the monthly total catch data for Hokkaido from September to February of each year for the period 2000 to 2016, as well as data on the surface water temperature at 50 m for each area and time.

Data covering the period from September 2000 to February 2015 are used as the training set, and data from September 2015 to February 2016 are used as the testing set. The catch data from month 1 to month n are fed into the ARIMA network as inputs; the daily surface water temperature data from month $n + 1$ are fed into the neural network, whereas the catch data from month $n + 1$ are used as the output.

Notably, for the month $n + 1$ water temperature data, divided the water temperature data into those of the Sea of Japan and the Pacific Ocean according to the actual sea area of the Hokkaido area (Fig. 4.2) and input these data into the neural network. Because the distribution of currents in the Sea of Japan differs from that in the Pacific Ocean, and the water temperature is affected by these currents [37], different states occur, which have different effects on the catch. Therefore, separating the water temperature data of these two bodies of water is more favorable for catch forecasting.

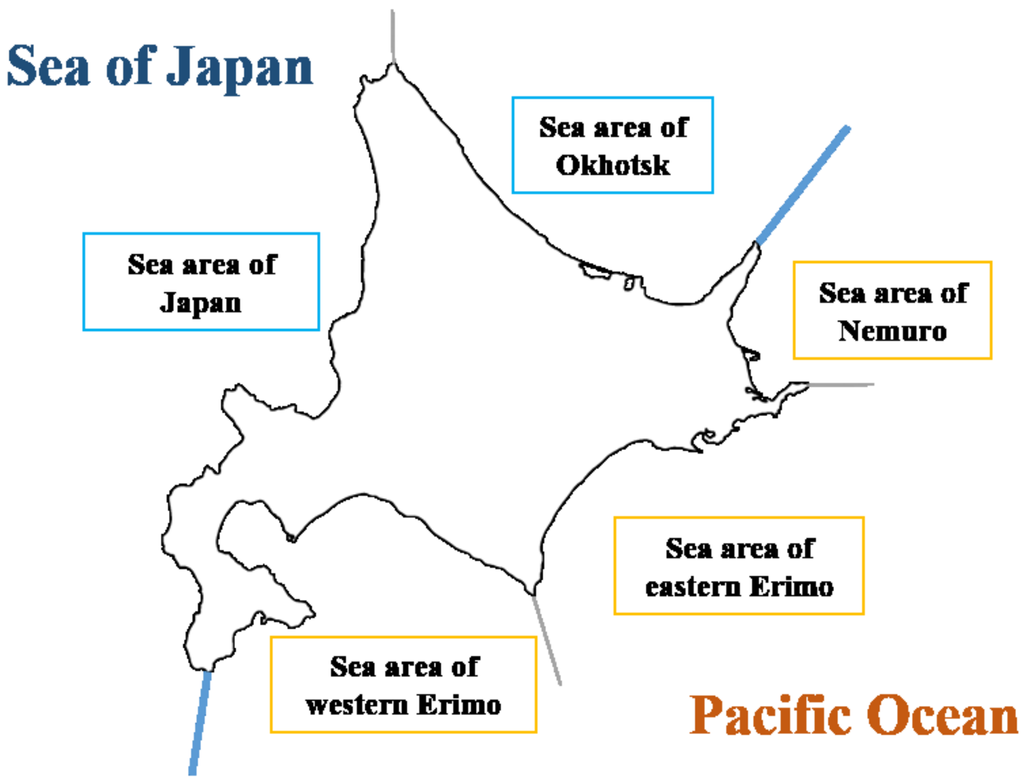


Figure 4.2: Hokkaido sea area distribution.

The water temperature data of the Sea of Japan were calculated by averaging the water temperature data from the sea area of Japan and those of Okhostk, whereas the water temperature data of the Pacific Ocean were calculated by averaging the water temperature data from the sea areas of Nemuro, Eastern Erimo, and Western Erimo.

The methods used in our comparative evaluation are as follows.

- **XGboost** [38] is a boosting algorithm.
- **ARIMA** indicates the autoregressive integrated moving average model.
- **LightGBM** [39] refers to a light gradient boosting machine, a distributed gradient boosting framework based on a decision tree algorithm.
- **LSTM** refers to a long short-term memory network model.
- **Hybrid methods** including Additive-ARIMA-LSTM, Multiplicative-ARIMA-LSTM, Additive-ETS-LSTM, and Multiplicative-ETS-LSTM four models.
- **L-ARIMA** is the proposed time-series data forecasting model based on LSTM and ARIMA.

Used one metric, namely the root mean squared error (RMSE), to measure the magnitude of the error in the forecasting results as a quantitative criterion. The metrics are defined as follows in Eqs. (5.9).

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{m} (\hat{y}_t^{\mathrm{catch}} - y_t^{\mathrm{catch}})^2}{m}}. \tag{4.2}$$

Table 5.2 shows the list of parameters used in the proposed methods. However, the parameters of the proposed and compared methods are determined in order to maximize the performance of each method.

Table 4.1: Details of parameters for the proposed method in long-term catch volume forecasting.

Parameters	Value
Order of the autoregressive model: p	3
Degree of differencing: d	1
Order of the moving-average model: q	2
Units in LSTM layer: L	32

4.3.2 Experimental results

The errors of the experiments are shown in Table 5.7. The upper two tables are comparisons of the RMSE from 2015/09 to 2016/02. The optimal estimation was not obtained by using LSTM. A plausible result was obtained by L-ARIMA. The averaged error for 10 estimations by using L-ARIMA is presented for each month in the lower table.

Table 4.2: Comparison of long-time forecasting errors.

Methods	ARIMA	XGBoost	LightGBM	LSTM	L$-$ARIMA
RMSE($\times 10^7$)	4.17	4.91	5.11	non	**2.25**

Methods	Additive ARIMA$-$LSTM	Multiplicative ARIMA$-$LSTM	Additive ETS$-$LSTM	Multiplicative ETS$-$LSTM	L$-$ARIMA
RMSE($\times 10^7$)	3.41	3.84	4.22	4.04	**2.25**

Year/ Month	2015/09	2015/10	2015/11	2015/12	2016/01	2016/02
RMSE($\times 10^7$) for each month by L-ARIMA	1.56	1.69	4.29	1.43	2.12	2.96

4.4 Summary

The experimental results indicate that the forecasting error is reduced after the LSTM layer, which is capable of handling the water temperature data, is added to the ARIMA model. The variation characteristics of water temperature help the network model to learn the variation patterns of the catch data. Although certain individual data are far from the real data, the forecast data still fit the real data effectively in general and could fit the trend of the real data appropriately. Moreover, compared with pure ARIMA and other forecasting methods, the L-ARIMA method achieve higher forecasting accuracy despite data volume limitations. Notably, in the case of the LSTM model used only for monthly catch forecasting, the neural network model is not able to perform learning and forecasting effectively, owing to the small amount of data.

Chapter 5

Analysis of Long-Term and Short-Term Fish Catch Data

5.1 Introduction

In this chapter, combined a method for predicting short-term captures with a method for predicting long-term captures. By predicting the capture volume in the Hokkaido region, explored the linkage of capture volume prediction results at multiple scales, as shown in Fig. 5.1. Specifically, based on the LSTM prediction of catch in Chapter **3**, proposed and implement a neural network model based on convolutional neural networks and long- and short-term memory neural networks with multiple time-scale features for predicting short-term measures of daily catch in the eastern ports of Hokkaido, Japan. Although the model proposed in Chapter **3** has achieved some results in short-term catch forecasting. But just the LSTM-based model also still has limitations, and there is still room for improvement. Therefore, in this chapter, proposed a new model based on the original one. The purpose is to get higher forecasting accuracy.

And a long-term yield prediction and analysis model combining the autoregressive integrated moving average (ARIMA) method and neural network proposed in Chapter **4** was used to explore short-term water temperature and long-term yield dependence in the presence of sparse data, and this method was used to investigate the total monthly catch in Hokkaido.

The data used in this case included not only temporal multi-scale but also spatial multi-scale was considered. Therefore, in the end, even performed correlation analysis and the intrinsic linkage of change trends for the data under both scales as well as the predicted results. In summary, this chapter builds on the content of Chapters **3** and **3**. It goes further to make an improved and deeper analysis based on the integration of their studies. As mentioned above, our proposed model can achieve high accuracy in predicting captures at both different temporal and spatial scales to meet different usage scenarios.

After that, **5.2** describes a catch prediction method based on CNN and LSTM for predicting the daily catch at ports in Hokkaido area. **5.3**, on the other hand, predicts the monthly catch of Hokkaido as a whole using the method prediction of Chapter **4**. After that, **5.4** describes the experimental conditions, the results and the correlation analysis of the long and short term capture volume data, and finally **5.5** gives the summary.

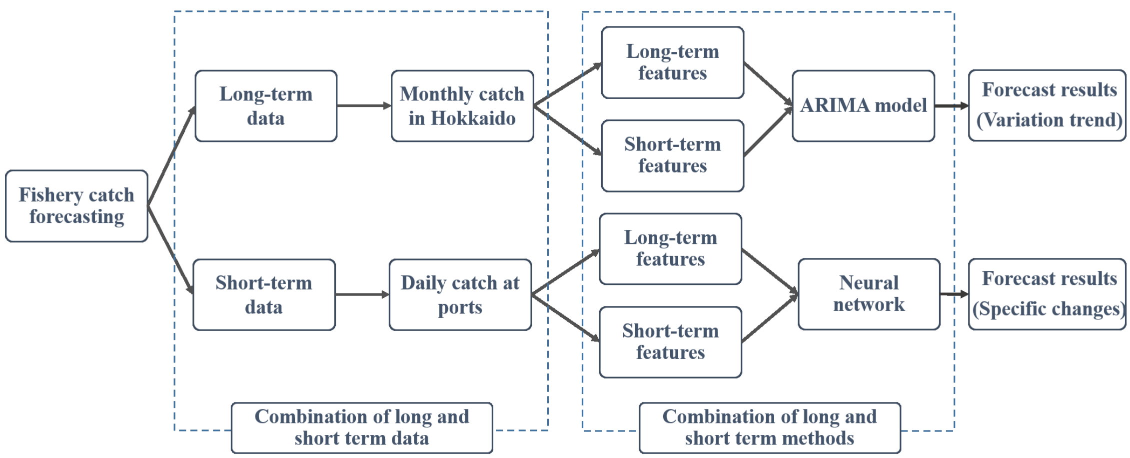


Figure 5.1: Process summary in Chapter 5.

5.2 Prediction of Daily Catch of Port Using Neural Network

In this chapter, proposed a capture volume prediction method combining convolutional neural networks and recurrent neural networks and enhancing the ability of the model to capture the characteristics of long- and short-term changes in the data with the designed data splitter structure.

In the process of applying fish catch forecasting, real time-series data often include a mixture of short- and long-term patterns. This is particularly true for port-catch data, which are highly cyclical. Long-term characteristics reflect the labor cycle, seasonal climate, and other changes, whereas short-term features reflect the effects of weather and human activities, among other influences. Accurate time-series forecasting is not possible without considering both types of cyclical patterns. In this study, aims to solve this problem using a machine learning approach.

Hochreiter et al. proposed the long short-term memory (LSTM) architecture as a machine learning method based on recurrent neural networks (RNNs) [40], with certain information mining capability for long-range temporal data [41]. LSTM models are widely used in the recognition of speech, emotion, and human activity, as well as in load forecasting [31, 42–44]. Behavior forecasting and the recognition of activity, speech, and emotion have also been performed by fusing convolutional neural network (CNN) [45] and LSTM [32, 33, 46, 47] models. The common bases of these methods are the extraction of high-dimensional features with the short sequence feature abstraction ability of CNN models and the subsequent synthesis of high-dimensional short-sequence features for temporal forecasting, which is suitable for processing temporal data with local relevance.

On the basis of these methods, used a specialized neural network model with a structure consisting of a convolutional layer, a data splitter, two recurrent LSTM layers, and a fully connected layer. Input layer is consisted with $n \times T$ nodes. Used T length sequence data $\mathcal{X}_s = \{\boldsymbol{x}_{s+1}, \boldsymbol{x}_{s+2}, \cdots, \boldsymbol{x}_{s+T}\}$ $(s = 0, 1, 2, \cdots)$ for the case$n = 2$. The whole data set generated by the given time-series data is presented by $\{\mathcal{X}_s\}_{s>0}$. Concretely, x_s^1 and x_s^2 are the variables of the input nodes for the temperature and catch volume data, respectively.

The model uses a convolutional layer to extract high-dimensional features, a data splitter to divide the data into different time scales, and a recurrent layer to capture the complex long-term dependency patterns. The splitter divides the data into two time scales; one is input to the LSTM layers and the other is processed by skip-LSTM layers. Finally, a fully connected layer is used to integrate the outputs of the two LSTM layers and generate the final prediction results. This network architecture enables better learning of the periodicity of the input time-series data (Fig. 5.2). The technical details of each section are described in detail later.

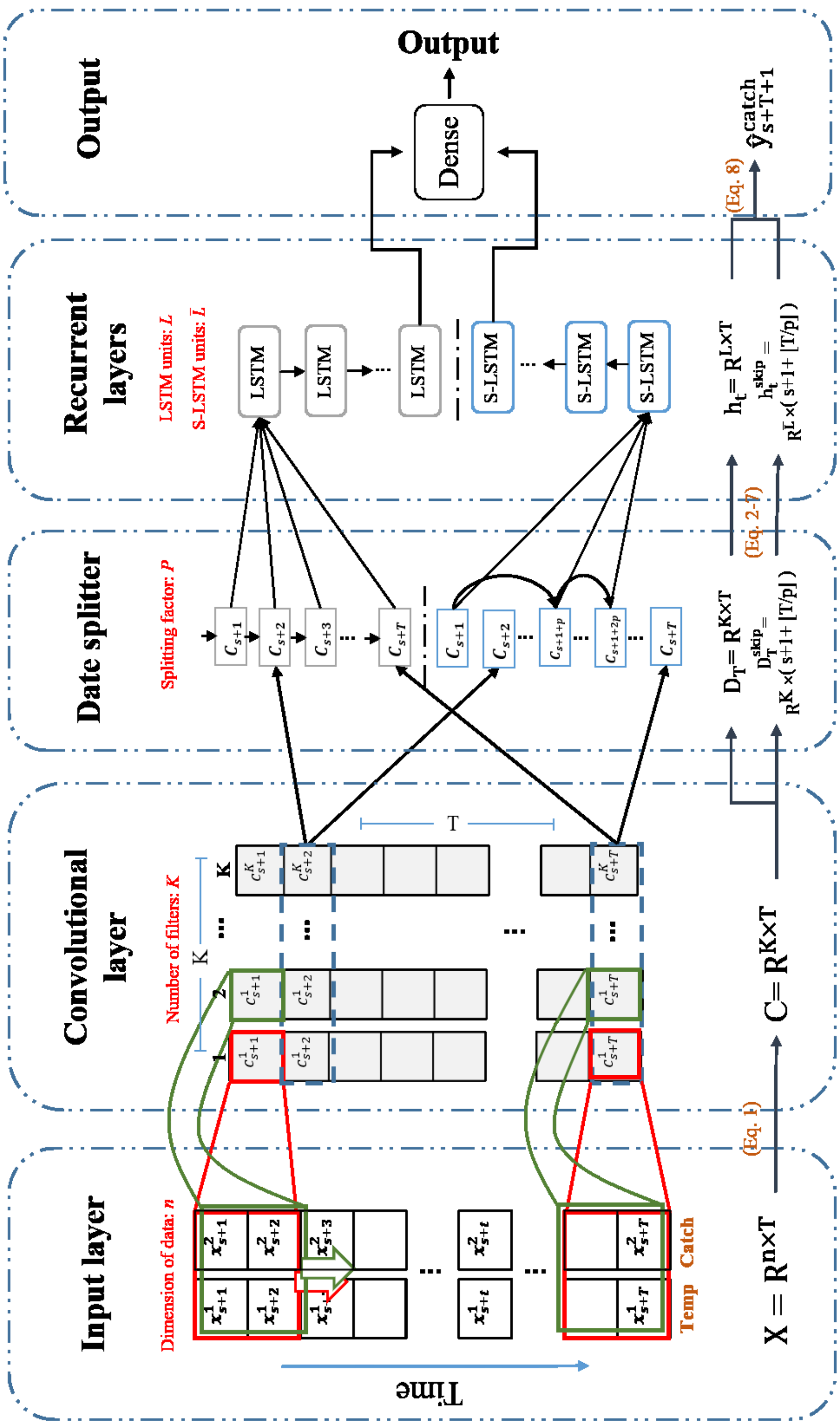


Figure 5.2: The model structure of a neural network for forecasting fish catch volume is presented.

5.2.1 Problem Formulation

Given time-series data $\mathcal{D} = \{\boldsymbol{y}_1, \boldsymbol{y}_2, \boldsymbol{y}_3, \cdots\}$, each $\boldsymbol{y}_s$ is the n-dimensional real vector. Our objective is to establish the estimation of $\boldsymbol{y}_{s+T+1}$ using T-length sequence data $\{\boldsymbol{y}_{s+1}, \boldsymbol{y}_{s+2}, \cdots, \boldsymbol{y}_{s+T}\}$, where $s = 0, 1, 2, \cdots$. That is, our goal is to generate a rolling forecast of a series of future signals. Concerning the dimension of each datum, used two cases for $n = 2$ and $n = 1$. In the case of $n = 2$, each datum is presented by $\boldsymbol{y}_s = (y_s^{\mathrm{temp}}, y_s^{\mathrm{catch}})$ for daily catch forecasting, In the case of $n = 1$, each datum is presented by $\boldsymbol{y}_s = y_s^{\mathrm{catch}}$ in monthly catch forecasting. Where y_s^{temp} and y_s^{catch}) present the sea temperature data and salmon catch volume data, respectively.

5.2.2 Convolutional Layer

The CNN model performs a convolution operation by generating convolutional check information in a receptive field of an appropriate size, which can express the original data at a higher, more abstract level. Therefore, Used the 1D CNN, which is widely used to process sequence data [48], as the convolutional layer (without a pooling layer) to extract the short-term features of the time series, where the input matrix is $\mathcal{X} \in \mathbb{R}^{n \times T}$, and the k-th filter sweeps through the input and produces $C^k = \left\{c_{s+1}^k, c_{s+2}^k, \cdots, c_{s+T}^k\right\}$ $(k = 1, 2, 3, \cdots, K)$, where K is the number of filters, as shown in Eq. (5.1).

$$C^k = \mathrm{ReLU}(\mathcal{X} * W_k + b_k), \tag{5.1}$$

where $*$ represents the convolution operation, $W_k \in \mathbb{R}^n$ is the weight vector of the convolution kernel, and b_k represents the offset. ReLU is used as the activation function, and ReLU (x) = Max (0, x). The structure of a 1D CNN is shown in the convolutional layer section of Fig. 5.2, with the input time-series data from the input layer, which is a multidimensional matrix $\mathcal{X} \in \mathbb{R}^{n \times T}$. It is convolved from top to bottom as shown by the arrows in the figure, with red representing one filter and green representing another, which can be followed by other filters. Making each vector C^k of length T by zero-padding for the input matrix $\mathcal{X}$ and the output matrix $\mathcal{C}$ of the convolutional layer of size $K \times T$.

5.2.3 Data Splitter

LSTM models can remember historical information and understand long-term dependencies. However, longer input results in more information being contained within the system, and when the sequence is excessively long, instability and gradient disappearance occur during the training of a single LSTM. Thus, long-term interdependence cannot be captured. The proposed model mitigates this problem by using a data splitter and bypassing the loop layer, which leverages the real-world cyclical pattern.

The data splitter intervals $(p \geq 1)$ longer data from the input data. In the output matrix $\mathcal{C} \in \mathbb{R}^{K \times T}$ of the convolutional layer, extracts $D_T^{\text{skip}} = \left\{C_{s+1}, C_{s+1+p}, \cdots, C_{s+1+\lfloor T/p \rfloor * p}\right\}$, where $D_T^{\text{skip}} \in \mathbb{R}^{K \times (s+1+\lfloor T/p \rfloor)}$. And the original data are used as the $D_T = \{C_{s+1}, C_{s+2}, \cdots, C_{s+T}\} \in \mathbb{R}^{K \times T}$, where $C_t = \{c_t^1, c_t^2, \cdots, c_t^K\}(t = s+1, s+2, \cdots, s+T)$. D_T is input into the LSTM layer, and the split data D_T^{skip} are then input into a new LSTM layer, S-LSTM.

5.2.4 Recurrent Layer

In the convolutional layer, the output enters the loop layer at the same time as the jump loop layer. The loop layer is an LSTM network that refers to the gate function, which is used to mine the time-series change rules of longer intervals in the time series. The structure is shown in Fig. 5.3.

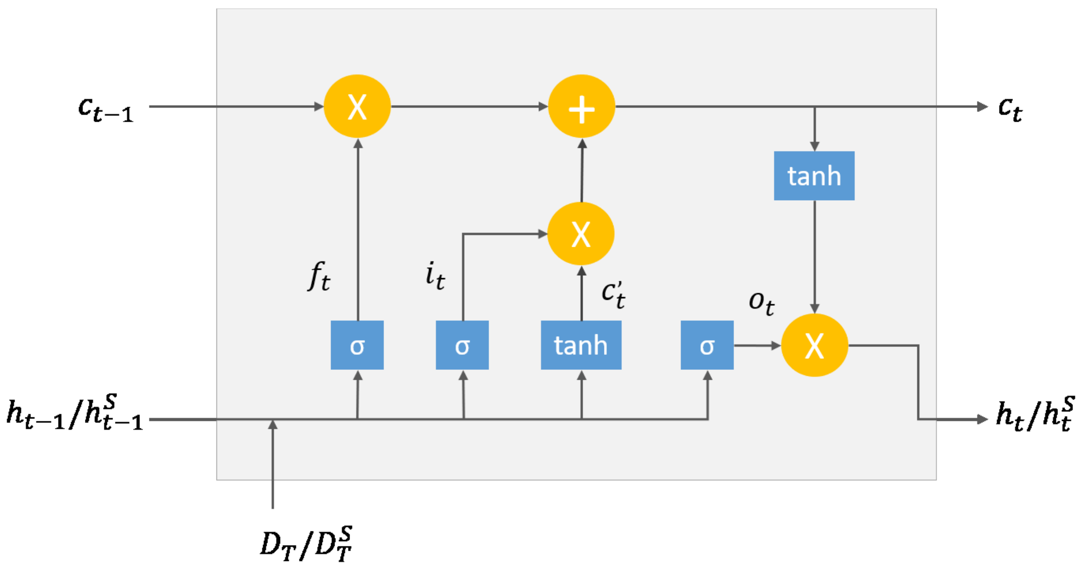


Figure 5.3: Network structure of LSTM.

D_T refers to the data value of the time t input sequence. c_t refers to a memory cell or cell state, which is the core of the network and controls the transmission of information. i_t refers to an input gate, which determines the amount of information that the current D_T retains in c_t. f_t refers to a forget gate, which determines the amount of c_{t-1} of the cell state from the previous moment is saved to the current c_t. o refers to the output gate, which determines the amount of output h_t transmitted by c_t to the current state. h_{t-1} refers to the state of the hidden layer at time $t-1$. The corresponding formulae for the aforementioned process are provided in Eqs. (5.2) to (5.7).

$$i_t = \sigma(W^{Di} D_T + W^{h-i} h_{t-1} + b_i), \tag{5.2}$$

$$f_t = \sigma(W^{Df} D_T + W^{h-f} h_{t-1} + b_f), \tag{5.3}$$

$$o_t = \sigma(W^{Do} D_T + W^{h-o} h_{t-1} + b_o), \tag{5.4}$$

$$\dot{c_t} = \tanh(W^{Dc}D_T + W^{h-c}h_{t-1} + b_c), \tag{5.5}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \dot{c_t}, \tag{5.6}$$

$$h_t = o_t \odot \tanh(c_t), \tag{5.7}$$

where W^{Di}, W^{Df}, W^{Do}, and W^{Dc} refer to the weight matrix of the input gate, forget gate, output gate, and cell state, respectively. W^{h-i}, W^{h-f}, W^{h-o}, and W^{h-c} refer to the weight matrix of the hidden layer to the input gate, forget gate, output gate, and cell state, respectively. b_i, b_f, b_o, and b_c refer to the input gate, forget gate, output gate, and cell state offset, respectively. $\sigma(\cdot)$ refers to the $sigmoid$ activation function $S(x) = 1/(1+e^{-x})$. $\tanh$ refers to the hyperbolic tangent activation function $\tanh x = \sinh x/\cosh x$. $\odot$ denotes the multiplying operation of the elements of a vector. The output of this layer is the hidden state of each timestamp, denoted by h_t. Here sets the dimensionality “LSTM units” of the output space to L, which means that all weight matrices W in them are in $\mathbb{R}^{L\times T}$.

The computational process of the S-LSTM layer is represented by the LSTM layer; simply replace the input data with the matrix D_T^{skip} and its output is denoted as h_t^{skip}. Setting the output dimensionality of the S-LSTM layer to $\bar{L}$, so that the range of all weight matrices W is $\mathbb{R}^{\bar{L}\times(s+1+\lfloor T/p\rfloor)}$.

Finally, a fully connected layer is used to combine the output of the LSTM and S-LSTM layers, as shown in Eq. (5.8).

$$\hat{y}_{s+T+1}^{\text{catch}} = W^{\text{LSTM}}h_t + W^{\text{S-LSTM}}h_t^{\text{skip}} + b, \tag{5.8}$$

W^{LSTM} and $W^{\text{S-LSTM}}$ refer to the weight matrix of the LSTM and S-LSTM, respectively. b refers to the offset. The output $\hat{y}_{s+T+1}^{\text{catch}}$ of the fully connected layer is the forecasting result, which represents the port catch volume data.

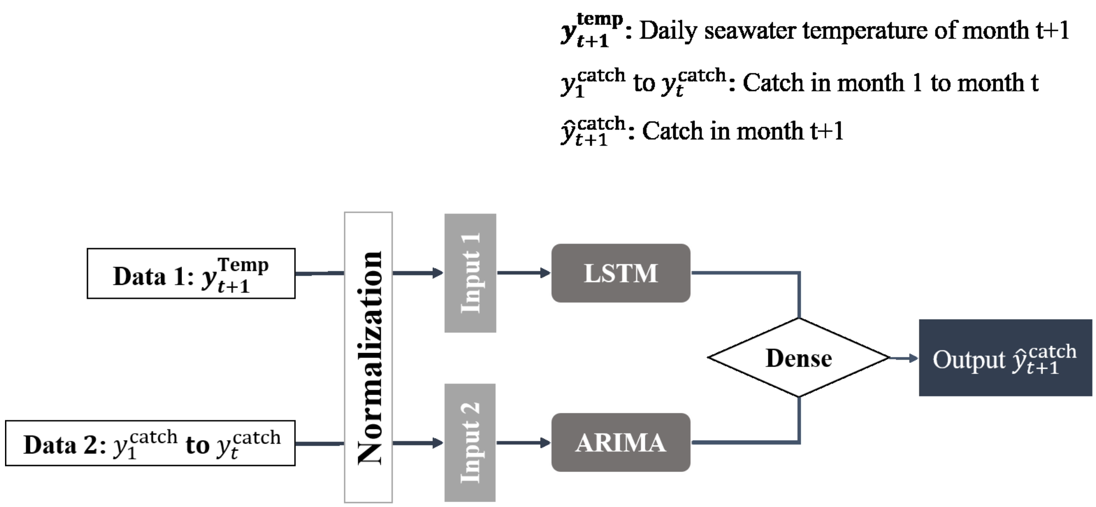


Figure 5.4: The structure of ARIMA based model.

5.3 Forecasting of Monthly Catch in Hokkaido Using ARIMA

In this section, still used the proposed, ARIMA-based forecasting method in Chapter **4** to forecast the monthly captures in Hokkaido region. In large scale catch forecasting, the data volume limitation due to large time units does not satisfy the most favorable conditions for neural network models, which are characterized by the need for large amounts of data for training. However, these large scale, long time unit data, which can represent the trend change in catch at macroscopic scale, are more linear and more suitable for traditional statistical methods. Therefore, used the ARIMA model to predict monthly catches for the entire Hokkaido region. And to address the problem that ARIMA cannot solve other characteristic variables, in order to consider the effect of water temperature on catch variation, introduced the LSTM layer to enhance the model's ability to handle multiple variables. The structure depicted in Fig. 5.4.

5.4 Experiments

In this section, to confirm the validity of the proposed long- and short-term catch prediction method, conducted prediction experiments for the port and the entire catch in the Hokkaido region. In the following, 5.4.1 shows the experimental conditions, and 5.4.2 shows the experimental results and discussion. The 5.4.3 section discusses the intrinsic correlation between the long- and short-term catches and the predicted results.

5.4.1 Experimental conditions

Used data from JIJI Fishery News, Fisheries Research Institute, and NEAR-GOOS. Specifically, used the daily port catch data from JIJI Fishery News with the surface seawater temperature data from NEAR-GOOS as local-area data for the forecasting studies, whereas the monthly Hokkaido catch data from the Fisheries Research Institute and the corresponding surface water temperature data (also from NEAR-GOOS) are used as wide-area data. The details of the data are listed in Table 5.2.

Table 5.1: Details of the data used for the study.

	Data Length	Data Type	Time Unit	URL
JIJI Fishery News	2013	Set-net Fishing (Non-public data)	Daily	http://suisan.jiji.com/apps/
National Fisheries Research Institute	96	Salmon Catch	Monthly	http://salmon.fra.affrc.go.jp/zousyoku/salmon/salmon.html
NEAR-GOOS	2013/96	Surface Seawater Temperature(50m)	Daily/Monthly	https://ds.data.jma.go.jp/gmd/goos/data/database.html

- Short-term catch volume forecasting

Used catch data from June to November of each year from the set-net fishing periods from 2005 to 2015. The location selected is the eastern part of Hokkaido, where the catch volume is relatively high, and selected four representative ports (as indicated in Fig. 5.5): Nemuro, Habomai, Ochiishi, and Rausu, as well as 50 m surface water temperature data at the same locations and at the same times.

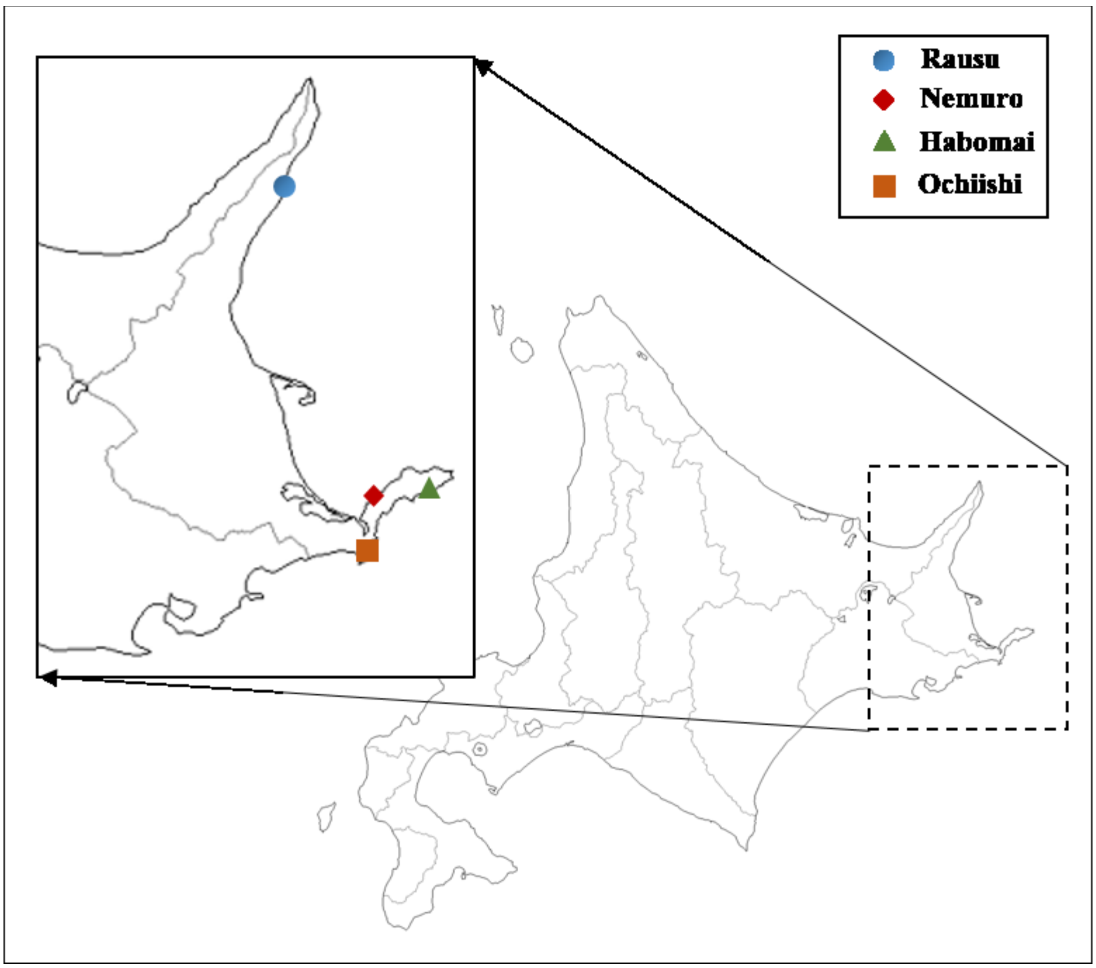


Figure 5.5: Location distribution of the four ports selected for the study.

Table 5.2: Details of parameters for the proposed method in short-term catch volume forecasting.

Parameters	Value
Dimension of data: n	2
Time steps: T	14
Number of filters in convolutional layer: K	40
Splitting factor in data splitter: P	4
LSTM units in recurrent layer: L	40
S-LSTM units in recurrent layer: $\bar{L}$	10

For this part of the experiment, as the total amount of data spanned the period 2005 to 2015, used the port catch and water temperature data for the 2005 to 2014 period from each of the four ports as the training set for the neural network input. Moreover, used the data for the remaining year, 2015, as the testing set to verify the experimental results. The inputs to the network are the catch and water temperature data from day 1 to day n, whereas the output datum is the catch on day $n + 1$.

Table 5.3: Details of parameters for the proposed method in long-term catch volume forecasting.

Parameters	Value
Order of the autoregressive model: p	3
Degree of differencing: d	1
Order of the moving-average model: q	2
Units in LSTM layer: L	32

- Long-term catch volume forecasting

Selected the monthly total catch data for Hokkaido from September to February of each year for the period 2000 to 2016, as well as data on the surface water temperature at 50 m for each area and time.

Data covering the period from September 2000 to February 2015 are used as the training set, and data from September 2015 to February 2016 are used as the testing set. The catch data from month 1 to month n are fed into the ARIMA network as inputs; the daily surface water temperature data from month $n+1$ are fed into the neural network, whereas the catch data from month $n+1$ are used as the output.

- Methods for Comparison

The methods used in our comparative evaluation are as follows.

- **XGboost** [38] is a boosting algorithm.
- **AR** refers to the autoregressive mode.
- **ARIMA** indicates the autoregressive integrated moving average model.
- **TCN** [49] is the temporal convolutional network, which combines dilated convolution and residual block.

- **CNN-LSTM** is the network model produced by combining the cell of CNN and that of LSTM.
- **LightGBM** [39] refers to a light gradient boosting machine, a distributed gradient boosting framework based on a decision tree algorithm.
- **LSTM** refers to a long short-term memory network model.
- **S-LSTM** is the proposed neural network model with a data splitter.
- **Hybrid methods** including Additive-ARIMA-LSTM, Multiplicative-ARIMA-LSTM, Additive-ETS-LSTM, and Multiplicative-ETS-LSTM four models.
- **L-ARIMA** is the proposed time-series data forecasting model based on LSTM and ARIMA.

All these methods are widely used for forecasting time-series data, covering the scope of statistics and machine learning. Among them, AR and ARIMA belong to traditional statistical methods, XGBoost and LightGBM are both gradient boosting decision trees (GBDTs) in traditional machine learning methods, and TCN belongs to deep learning. Specifically, AR and ARIMA models, as traditional statistical models, are used even more widely in forecasting efforts such as price forecasting [50], wind speed forecasting [51], and even COVID- 19 situations [52]. On the other hand, XGBoost applies to both classification and regression and is used in prediction work in industries such as electricity [53] and health care [54], as well as in web text classification, malware classification [55], etc. Similarly, LightGBM has performed well in financial forecasting and cancer patient classification [56] [57]. Although TCN is a recently proposed model, it is also widely used in the fields of the weather forecasting [58], runoff forecasting [59], etc.

In addition, to validate the model's performance combining ARIMA and LSTM (L-ARIMA) is proposed in this doctor thesis. Two additive hybrid methods(Additive-ARIMA-LSTM, Additive-ETS-LSTM) and two multiplicative hybrid methods (Multiplicative-ARIMA-LSTM, Multiplicative-ETS-LSTM), which directly combine linear and non-linear models, are used for comparison.

The experiments were conducted using the TensorFlow [60] machine learning framework for the Python programming language.

- Metrics

In this study, used one metric, namely the root mean squared error (RMSE), to measure the magnitude of the error in the forecasting results as a quantitative criterion. The metrics are defined as follows in Eqs. (5.9).

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{m} (\hat{y}_t^{\mathrm{catch}} - y_t^{\mathrm{catch}})^2}{m}}. \tag{5.9}$$

where $\hat{y}_t^{\mathrm{catch}}$ represents the forecast value, y_t^{catch} represents the true value, and m represents the total amount of test data. The RMSE is the square root of the ratio of the sum of the squares of the deviations of the observations from the true value to the number of observations m. It is generally used to measure the deviation of the observations from the true values.

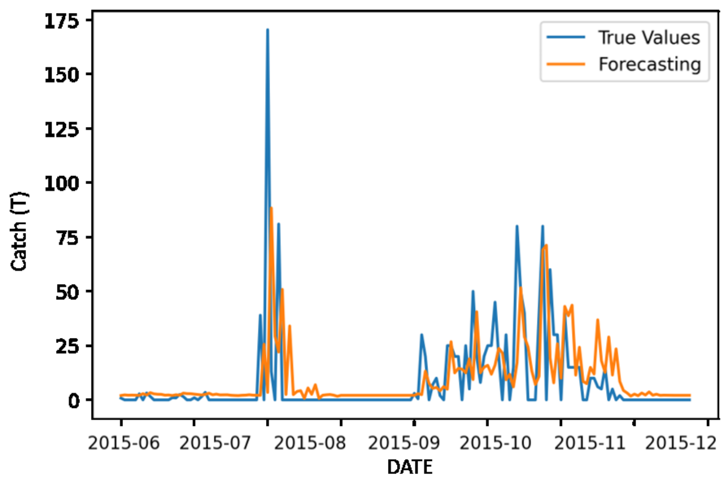
5.4.2 Experimental results

5.4.2.1 Results of daily catch volume

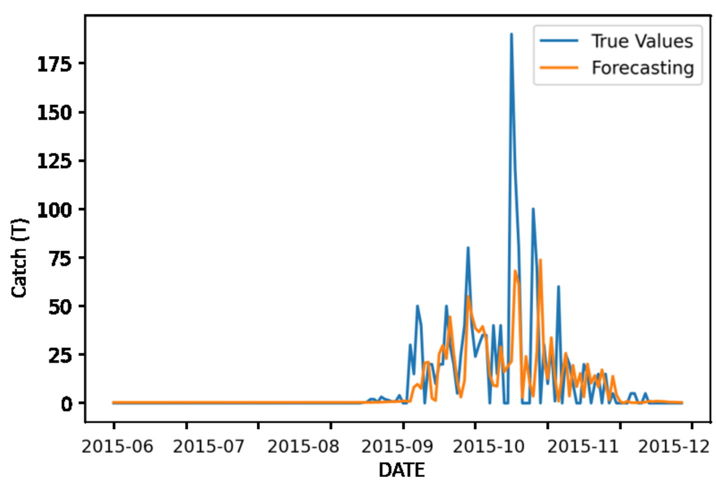
Table 5.4 presents the forecasting errors for the catch at each of the four ports in eastern Hokkaido as predicted by the neural network. Also calculated the error percentage for comparison purposes because of the different scales of the catch volume data from port to port. This corresponds to the error after normalization of the data from the four ports and is more indicative of the model's predictive capability at each port. From the table, the S-LSTM model reduced forecasting error compared to other models commonly used for time-series data forecasting. This model greatly alleviates the unavoidable gradient disappearance and explosion problems in recurrent neural network optimization and improves forecasting accuracy by simultaneously increasing the perceptual range of the model to the data. It can be observed that the forecasting of the network model for the catch at each of the four ports fluctuated to an extent but generally remained within a specific range. The model exhibited slightly better forecasting for Nemuro compared to the other three ports, which was also reflected in the visualization results.

Table 5.4: Comparison of short-time forecasting errors.

Ports	Errors	XGboost	AR	ARIMA	TCN	CNN-LSTM	S-LSTM
Nemuro	RMSE	21.49	21.48	20.63	23.53	24.39	**19.43**
	RMSE[%]	3.32	3.32	3.19	3.94	3.77	**3.00**
Habomai	RMSE	24.11	22.07	26.12	30.79	23.71	**20.14**
	RMSE[%]	8.04	7.36	8.71	10.25	7.90	**6.71**
Ochiishi	RMSE	5.35	**5.30**	7.95	6.58	7.82	5.89
	RMSE[%]	4.46	**4.42**	6.63	5.49	6.52	4.91
Rausu	RMSE	90.99	82.89	87.82	84.17	98.01	**78.75**
	RMSE[%]	10.11	9.21	9.76	9.35	10.89	**8.75**

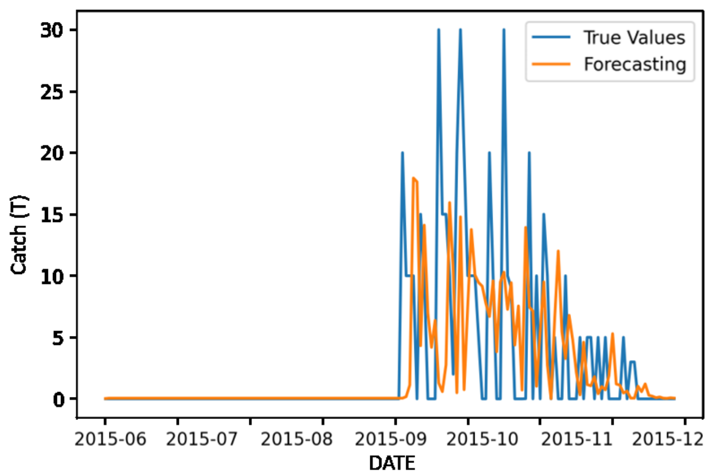


(a) Nemuro

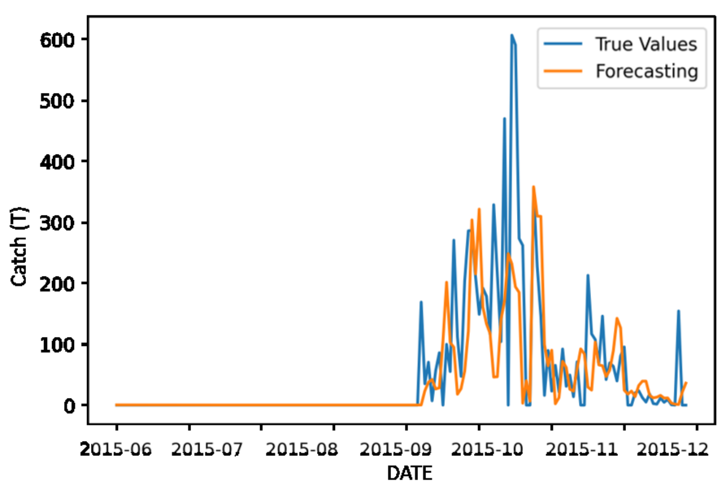


(b) Habomai

Figure 5.6: Forecasting results of daily catch per port using the proposed method(1).



(a) Ochiishi



(b) Rausu

Figure 5.7: Forecasting results of daily catch per port using the proposed method(2).

A visualization of the forecasting is shown in Fig. 5.6 and Fig. 5.7, with a comparison of the forecast and true values in 2015 for the four ports of Nemuro, Habomai, Ochiishi, and Rausu. The orange curve represents the forecast results and the blue curve represents the true changes in catch. The x-axis represents the time and the y-axis represents the catch (in tons). It can be seen that the network fit the real data well in the part where the capture was zero. In the period from October to November, when the catch volume changed dramatically, the forecasting of the capture volume data from Nemuro exhibited the best fit.

Overall, although the network showed good forecasting of the capture volume, it did not exhibit a good forecasting ability for the peak portion of the catch volume, particularly for Habomai and Rausu. Owing to the high sensitivity of the RMSE error to outliers, the RMSE [%] values for Habomai and Rausu in Table 5.4 were also higher than those for the other two ports. This may be attributed to insufficient feature volume data. Although the water temperature data were also included as a feature quantity in this experiment, it may be possible to capture the variation patterns of the catch quantity more effectively by incorporating richer feature quantities, such as wind speed and weather.

Table 5.5: p-values of t-test for S-LSTM concerning four ports.

	Nemuro	Habomai	Ochiishi	Rausu
p-value	0.233	0.405	0.334	0.657

Also performed statistical tests of the forecast results. Concretely, t-test and likelihood-ratio test were applied for the forecast results by S-LSTM and the other methods. The p-values of the t-test results are presented in Table 5.5, while the p-values of the

Table 5.6: p-values of the likelihood-ratio test for S-LSTM against other methods (CNN-LSTM, TCN, ARIMA, AR and XGBoost).

	Nemuro	Habomai	Ochiishi	Rausu
CNN-LSTM: S-LSTM ($\times 10^{-3}$)	0.432	6.84	4.11	2.42
TCN: S-LSTM ($\times 10^{-3}$)	1.10	5.94	0.343	4.81
ARIMA: S-LSTM ($\times 10^{-3}$)	0.179	0.275	1.22	5.39
AR: S-LSTM ($\times 10^{-3}$)	0.028	1.81	46.2	8.83
XGBoost: S-LSTM($\times 10^{-3}$)	4.70	5.83	23.0	4.17

likelihood-ratio test are shown in Table 5.6. As a result, the p-values were greater than 0.05. Therefore, the forecast results by S-LSTM were not significantly different from the target data. Similarly, the p-values of the likelihood-ratio test are less than 0.05, and the goodness of fit to the target data of the proposed method S-LSTM is better than the other methods.

5.4.2.2 Results of Monthly catch volume

The errors of the experiments are shown in Table 5.7. The upper two tables are comparisons of the RMSE from 2015/09 to 2016/02. The optimal estimation was not obtained by using LSTM. A plausible result was obtained by L-ARIMA. The averaged error for 10 estimations by using L-ARIMA is presented for each month in the lower table.

Table 5.7: Comparison of long-time forecasting errors.

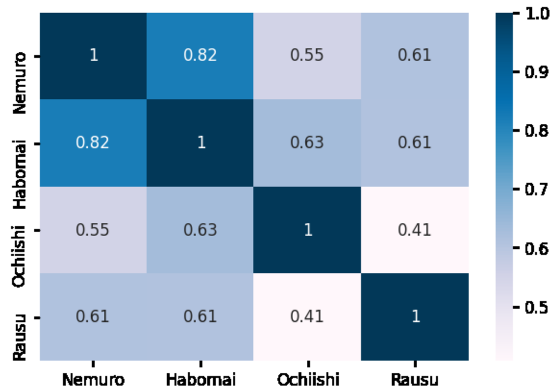
Methods	ARIMA	XGBoost	LightGBM	LSTM	L−ARIMA
RMSE($\times 10^7$)	4.17	4.91	5.11	non	**2.25**

Methods	Additive ARIMA−LSTM	Multiplicative ARIMA−LSTM	Additive ETS−LSTM	Multiplicative ETS−LSTM	L−ARIMA
RMSE($\times 10^7$)	3.41	3.84	4.22	4.04	**2.25**

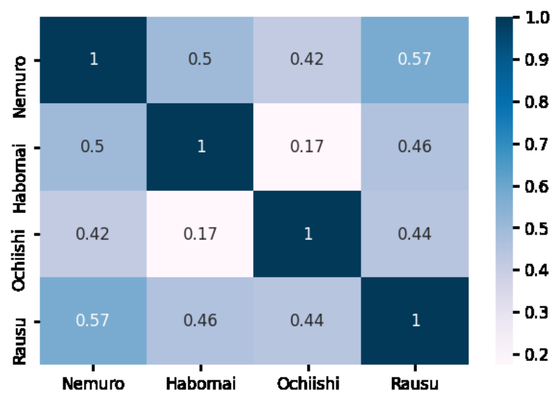
Year/ Month	2015/09	2015/10	2015/11	2015/12	2016/01	2016/02
RMSE($\times 10^7$) for each month by L-ARIMA	1.56	1.69	4.29	1.43	2.12	2.96

5.4.3 Analysis of Long-Term and Short-Term Fish Catch Data

Used monthly catch data from September to early December 2015 for Hokkaido and the port catch data to analyze the correlation between the long-term and short-term catches, as shown in Fig. 5.9. First, although the catches of the four ports are not quantitatively the same, the trends in the timeline were very similar, and the increasing and decreasing changes in the catches of each port were consistent. This is also reflected in the correlation distribution of catches between ports, as shown in Fig. 5.8. Analyzed and compare the correlation of catch data for the four ports from September 1, 2015, to November 30, 2015. Fig. 5.8 (a) is the correlation plot generated from the real values of port captures, and Fig. 5.8 (b) is the correlation plot generated from the results obtained by our proposed network structure with two kinds of LSTM. In Fig. 5.8 (b), the correlations are reduced due to the errors in forecasting (especially between Habomai and Ochiishi). However, a correlation among the catches of the four ports is evident nonetheless. The variations in the catches among the four ports in the same region are influenced by geographical characteristics and are consistent to a certain extent. For this reason, the correlations can contribute to the prediction of catches in different ports in the same region. And as additional information, the reference data between a related pair of two ports in Fig. 5.8 might be available for the total forecasting of port catch in eastern Hokkaido.



(a) Correlation for true data



(b) Correlation for forecast results

Figure 5.8: Correlation of ports catch volume data, September–November 2015.

Second, the monthly catch in Hokkaido is a cumulative value, and the magnitude of its change reflected the amount of real-time catch by combining the monthly catch of the entire island of Hokkaido and the changes in the catch data of its eastern ports. It can be observed that at the beginning of October, when the rapid increase in the monthly catch in Hokkaido approached its peak, the catch of each port also increased. Moreover, when the catch of each port decreased after the second half of October, the change in the monthly catch in Hokkaido also tended to level off. Although the port catch data are slightly inconsistent between the forecast results and the real values, it is nonetheless evident that the real-time changes in port catches are in line with the long-term trend shown by the monthly catches.

According to the aforementioned analysis, there are correlations in both the spatial and time scales between the monthly Hokkaido catch as long-term data and the port catch as short-term data. The analysis and forecasting of the two data types also play a positive role in corroborating one another and help in the analysis of the changes in the catches from different perspectives.

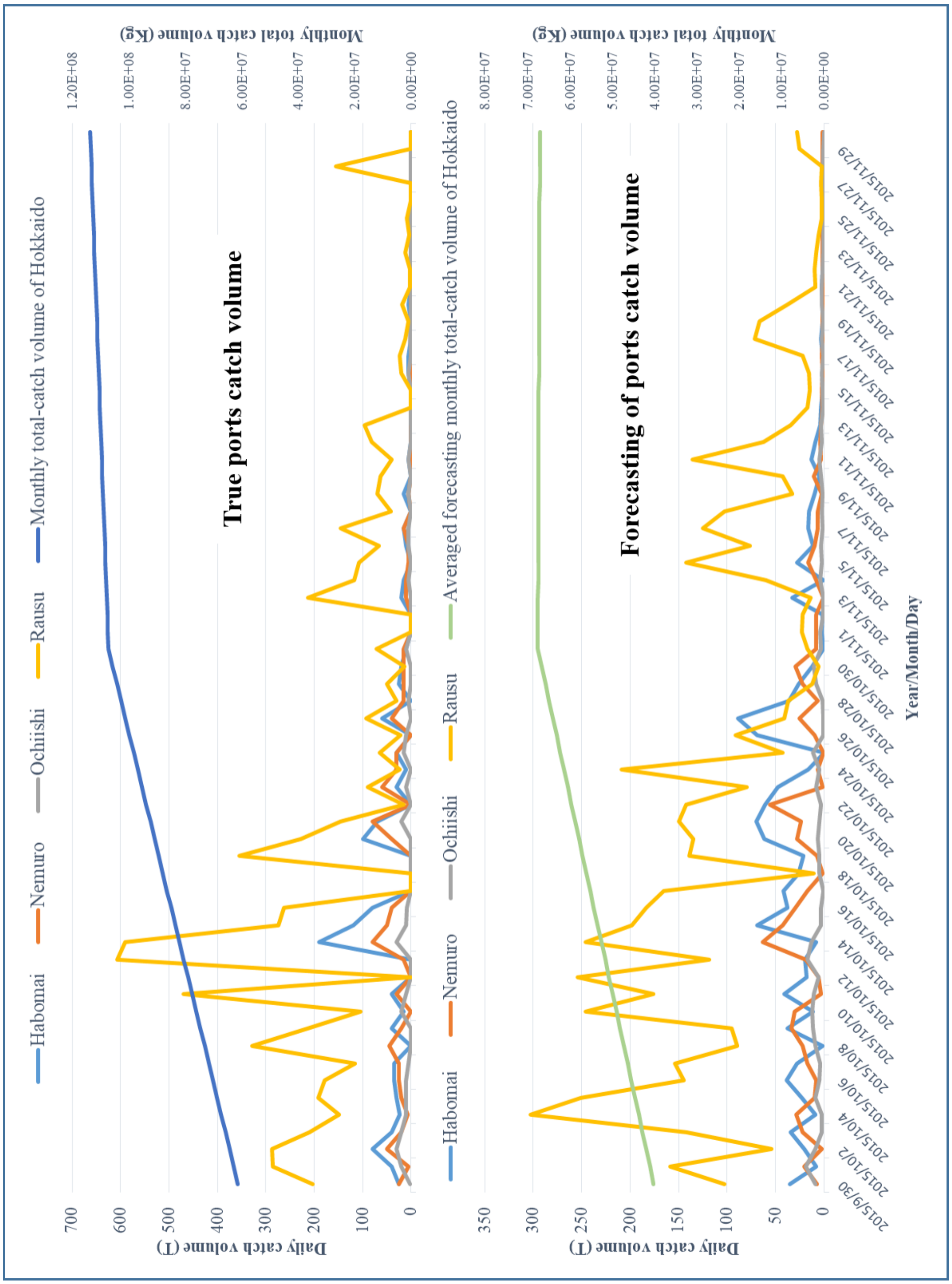


Figure 5.9: Comparison of fish catch from 2015–9 to 2015–12.

5.5 Summary

In this chapter, Two models were used to study the catch of Hokkaido. A neural network model was used to forecast the catch from the local area, short-term data, represented by the daily catch of the port, and the ARIMA model was used to study the catch forecasting from the larger regional area, long-term data, represented by the monthly catch of the entire island of Hokkaido. Finally, the two different data and their prediction results are analyzed and discussed for different time and space scales. The results demonstrated the feasibility of the proposed approach adopting two models to forecast the data on different scales; the models were able to perform forecasting of catches with a certain accuracy. Furthermore, the forecasting results can reflect the changing patterns of the catch on different scales by forecasting and analyzing the catch from different angles, which can aid practitioners in understanding the overall data trends and reflect detailed short-term changes to guide daily fishing work.

Chapter 6

Conclusions

6.1 Summary of this research

This doctor thesis proposed a new method that can handle capturing volume data at different time scales by building a deep learning method applicable to multi-timescale capture volume prediction for various scenarios in practical applications. In the following, will outline each chapter of this doctor thesis.

In chapter **2**, explained the conventional research on capture volume prediction as a related study in this research and clarify the problem to be addressed in this doctor thesis. In Chapter **3**, constructed a short-term catch prediction method using a long short-term memory network (LSTM) combined with a Gaussian distribution-based data augmentation method suitable for catch prediction with sparse data. Specifically, simulated the migratory movement of fish and use the available data to generate catch data that are closer to the real ones by Gaussian distribution. The data are then filtered to remove influencing factors such as rest days to obtain higher prediction results. In **4** chapter, constructed an ARIMA-based prediction method for long-term captures. Specifically, the properties of the autoregressive method are used to process the long-term catch data, while an LSTM layer is added to process other feature data such as water temperature. Thus, high-precision prediction of long-term catch data with sparse data is achieved. In chapter **5**, the methods of short-term catch prediction and long-term catch prediction are combined. For the short-term capture prediction method, based on the LSTM model in Chapter **3**, a capture prediction method based on CNN and LSTM is proposed and applied to the prediction of daily capture. And a prediction experiment is conducted on monthly capture data using the long-term capture prediction method in **4** chapter. The prediction results of the two experiments were also analyzed in combination.

Among the many capture prediction techniques, a deep learning approach was chosen to perform capture prediction. From the results, the deep learning method does achieve better accuracy than other methods. Moreover, based on this, that was able to achieve better prediction accuracy by combining the comprehensive prediction techniques of deep learning methods and traditional statistical methods based on data characteristics.

As mentioned above, high accuracy processing of captures in different scenarios was achieved by building deep learning models capable of handling different scales of capture predictions.

6.2 Future issues of this research

In this thesis, by proposing a deep learning approach that introduces knowledge of catch forecasting, implemented methods that enable high-precision prediction of captures at multiple scales. In the actual prediction work, in order to cater to the characteristics and limitations of the data on each scale, designed different deep learning models according to different usage scenarios. Although the prediction accuracy is guaranteed, this undoubtedly increases the difficulty and resource consumption of the work. Based on the correlation of long- and short-term data discussed in the thesis and in future work, it can be seen that the design aspects of the models can be optimized and improved. able to design more general models that also meet the prediction requirements of data at different scales by mining the commonality of the data.

In future work, plans to start from the correlation of long- and short-term data and explore the possibility of using long- and short-term data to complement each other in a capture forecasting problem to achieve scaling of the data over time scales and thus improve forecasting accuracy.

The above two points summarize the future issues of this study.

Acknowledgment

It has been 7 years since I came to Japan, and the confusion I felt when I was running around for my future is still fresh in my mind, but I have already reached the end of my doctorate.

I would like to thank my advisor, Prof. Shioya, for his guidance during the five years of master's to doctoral studies. I am grateful to my advisor, Shioya, for his guidance during the five years from the M.S. to the Ph.D. He not only led me through the research path but also gave me deep care in my life. He has also devoted a lot of effort to choose the topic, and writing and revising my thesis. It is your careful guidance that has brought me to my present achievements.

I would also like to thank Dr. Suziki for your great help in my lab work and thesis writing. Your careful revision of my writing materials has been a great help to me on my way.

Thank you to my family for your tolerance and support. Thanks to my uncle's help, I was able to come to Japan to continue my studies. Thank you, my parents, for your endless support. More importantly, I would like to thank my wife for being with me all the way across the ocean. Finally, I would also like to thank my daughter, your arrival has made me motivated and fearless.

As my doctoral career is coming to an end, a thousand words are not enough to express my feelings. I only hope that my future life will also live up to this time.

References

[1] R. Saville, K. Hatanaka, M. Sano, and M. Wada, "Application of information and communication technology and data sharing management scheme for the coastal fishery using real-time fishery information," Ocean & Coastal Management, vol.106, pp.77–86, 2015.

[2] Fisheries Agency, "Fy2019 trends in fisheries fy2020 fisheries policy summary," 2019. Japan.

[3] K. Sakaguchi and N. Yamashita, "Method to forecast the catches of japanese common squid todarodes pacificus in the sea of okhotsk off hokkaido," Bulletin of the Japanese Society of Fisheries Oceanography, vol.79, no.2, pp.43–45, 2015.

[4] S.E. Matson, I.G. Taylor, V.V. Gertseva, and M.W. Dorn, "Novel catch projection model for a commercial groundfish catch shares fishery," Ecological modelling, vol.349, no.C, pp.51–61, 2017.

[5] T. Komiya, "Relationship of amount of puerulus postlarvae and first-stage fry shrimp collected and japanese spiny lobster catch in chiba prefecture," Bulletin of the Chiba Prefectural Fisheries Research Center, vol.12, pp.27–34, 2018.

[6] J.R. Leathwick, J. Elith, M.P. Francis, T. Hastie, and P. Taylor, “Variation in demersal fish species richness in the oceans surrounding new zealand: an analysis using boosted regression trees,” Marine Ecology Progress Series, vol.312, pp.267–281, 2006.

[7] Y. Kokaki, N. Tawara, *et al.*, “Daily fish catch forecasting for fixed shore net fishing using state space model describing probabilistic behavior of fish inside net,” IEICE Tech. Rep., vol.119, no.64, pp.13–18, 2019.

[8] G.E. Box, G.M. Jenkins, G.C. Reinsel, and G.M. Ljung, Time series analysis: forecasting and control, John Wiley & Sons, 2015.

[9] J. Caddy, Some considerations underlying definitions of catchability and fishing effort in shellfish fisheries, and their relevance for stock assessment purposes, Department of Fisheries and Oceans, Resource Branch, Invertebrates & Marine ⋯, 1979.

[10] J.A. Gulland, Fish stock assessment: a manual of basic methods, Wiley, 1983.

[11] G. Romagnoni, S. Mackinson, J. Hong, and A.M. Eikeset, “The ecospace model applied to the north sea: Evaluating spatial predictions with fish biomass and fishing effort data,” Ecological Modelling, vol.300, pp.50–60, 2015.

[12] V. Christensen and D. Pauly, “Ecopath ii—a software for balancing steady-state ecosystem models and calculating network characteristics,” Ecological modelling, vol.61, no.3-4, pp.169–185, 1992.

[13] C. Walters, V. Christensen, and D. Pauly, "Structuring dynamic models of exploited ecosystems from trophic mass-balance assessments," Reviews in fish biology and fisheries, vol.7, no.2, pp.139–172, 1997.

[14] C. Walters, D. Pauly, and V. Christensen, "Ecospace: prediction of mesoscale spatial patterns in trophic relationships of exploited ecosystems, with emphasis on the impacts of marine protected areas," Ecosystems, vol.2, no.6, pp.539–554, 1999.

[15] 曹聡, 宇都宮栄二, and 吉原貴仁, "スマートブイデータを用いた漁種別漁獲量予測の検討," 電子情報通信学会技術研究報告; 信学技報, vol.121, no.266, pp.52–54, 2021.

[16] 荒井頼子 and 出原真理子, "機械学習を用いた衛星画像による漁場予測の研究," 日本水産工学会誌, vol.56, no.1, pp.57–60, 2019.

[17] M. Navarro, A. Hailu, T. Langlois, K.L. Ryan, and M.E. Kragt, "Determining spatial patterns in recreational catch data: a comparison of generalized additive mixed models and boosted regression trees," ICES journal of marine science, vol.77, no.6, pp.2216–2225, 2020.

[18] T. Chen, L. Zhong, N. Zhou, and D. Hoppe, "Catch weight prediction for multi-species fishing using artificial neural networks," 2021 20th IEEE International Conference on Machine Learning and Applications (ICMLA), pp.1545–1552, IEEE, 2021.

[19] R. Nian, Q. Yuan, H. He, X. Geng, C.W. Su, B. He, and A. Lendasse, "The identification and prediction in abundance variation of atlantic cod via long short-term memory with periodicity, time–frequency co-movement, and lead-lag effect

across sea surface temperature, sea surface salinity, catches, and prey biomass from 1919 to 2016," Frontiers in Marine Science, p.629, 2021.

[20] T.J. Quinn and R.B. Deriso, Quantitative fish dynamics, oxford university Press, 1999.

[21] O. Sakai and Y. Tsukahara, "Japanese coastal longline cpue and catch-at-length for pacific bluefin tuna: Update up to 2016 fishing year," tech. rep., ISC/18/PBFWG-1/01, 2018.

[22] 岩田繁英, 中原尚知, and 婁小波, "社会環境影響評価のための簡易な漁獲量推定方法の検討: 漁獲確率を用いた漁獲量推定方法の可能性," 国際漁業研究, vol.17, no.1, pp.31–41, 2019.

[23] S. Giri, S. Purkayastha, S. Hazra, A. Chanda, I. Das, and S. Das, "Prediction of monthly hilsa (tenualosa ilisha) catch in the northern bay of bengal using bayesian structural time series model," Regional Studies in Marine Science, vol.39, p.101456, 2020.

[24] 万田敦昌, 小川翔大, 久野正博, 藤田弘一, 武田保幸, 御所豊穂, 海野幸雄, and 山田二久次, "機械学習を用いた三重県におけるブリ類漁獲量の実用的予測モデルの構築," 国際漁業研究, vol.18, no.1, pp.1–19, 2020.

[25] 山下慶太郎, "高知県周辺海域で漁獲されるカツオの漁況予測," 黒潮の資源海洋研究, 2020.

[26] R.A. Cavieses Núñez, M.Á. Ojeda Ruiz de la Peña, A. Flores Irigollen, and M. Rodríguez Rodríguez, "Deep learning models for the prediction of small-

scale fisheries catches: finfish fishery in the region of "bahía magadalena-almejas " ," ICES Journal of Marine Science, vol.75, no.6, pp.2088–2096, 2018.

[27] W.A.G. NA, "Forecasting the future and understanding the past," 1994.

[28] M. Längkvist, L. Karlsson, and A. Loutfi, "A review of unsupervised feature learning and deep learning for time-series modeling," Pattern Recognition Letters, vol.42, pp.11–24, 2014.

[29] G.E. Box, G.M. Jenkins, G.C. Reinsel, and G.M. Ljung, Time series analysis: forecasting and control, John Wiley & Sons, 2015.

[30] H.F. Yu, N. Rao, and I.S. Dhillon, "Temporal regularized matrix factorization for high-dimensional time series prediction," Advances in neural information processing systems, vol.29, 2016.

[31] C. Fan, F. Xiao, and Y. Zhao, "A short-term building cooling load prediction method using deep learning algorithms," Applied energy, vol.195, pp.222–233, 2017.

[32] F.J. Ordóñez and D. Roggen, "Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition," Sensors, vol.16, no.1, p.115, 2016.

[33] G. Trigeorgis, F. Ringeval, R. Brueckner, E. Marchi, M.A. Nicolaou, B. Schuller, and S. Zafeiriou, "Adieu features? end-to-end speech emotion recognition using a deep convolutional recurrent network," Proc. 2016 IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP), pp.5200–5204, 2016.

[34] M. Längkvist, L. Karlsson, and A. Loutfi, "A review of unsupervised feature learning and deep learning for time-series modeling," Pattern Recognition Letters, vol.42, pp.11–24, 2014.

[35] H. Akaike, "Autoregressive model fitting for control," in Selected Papers of Hirotugu Akaike, pp.153–170, Springer, New York, 1998.

[36] J. Durbin, "Efficient estimation of parameters in moving-average models," Biometrika, vol.46, no.3/4, pp.306–316, 1959.

[37] O. Leeuwenburgh and D. Stammer, "The effect of ocean currents on sea surface temperature anomalies," Journal of physical oceanography, vol.31, no.8, pp.2340–2358, 2001.

[38] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," Proc. of the 22nd acm sigkdd international conference on knowledge discovery and data mining, pp.785–794, 2016.

[39] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.Y. Liu, "Lightgbm: A highly efficient gradient boosting decision tree," Advances in Neural Information Processing Systems, vol.30, pp.3146–3154, 2017.

[40] A. Cleeremans, D. Servan-Schreiber, and J.L. McClelland, "Finite state automata and simple recurrent networks," Neural computation, vol.1, no.3, pp.372–381, 1989.

[41] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural computation, vol.9, no.8, pp.1735–1780, 1997.

[42] A.S. Weigend, Time series prediction: forecasting the future and understanding the past, Routledge, 2018.

[43] A. Baird, S. Amiriparian, M. Milling, and B.W. Schuller, “Emotion recognition in public speaking scenarios utilising an lstm-rnn approach with attention,” Proc. 2021 IEEE Spoken Language Technology Workshop (SLT), pp.397–402, 2021.

[44] A. Saeed, C. Li, M. Danish, S. Rubaiee, G. Tang, Z. Gan, and A. Ahmed, “Hybrid bidirectional lstm model for short-term wind speed interval prediction,” IEEE Access, vol.8, pp.182283–182294, 2020.

[45] A. Krizhevsky, I. Sutskever, and G.E. Hinton, “Imagenet classification with deep convolutional neural networks,” Communications of the ACM, vol.60, no.6, pp.84–90, 2017.

[46] K. Xia, J. Huang, and H. Wang, “Lstm-cnn architecture for human activity recognition,” IEEE Access, vol.8, pp.56855–56866, 2020.

[47] S. Zhang, X. Zhao, and Q. Tian, “Spontaneous speech emotion recognition using multiscale deep convolutional lstm,” IEEE Trans on Affective Computing, 2019.

[48] O. Abdeljaber, O. Avci, S. Kiranyaz, M. Gabbouj, and D.J. Inman, “Real-time vibration-based structural damage detection using one-dimensional convolutional neural networks,” Journal of Sound and Vibration, vol.388, pp.154–170, 2017.

[49] S. Bai, J.Z. Kolter, and V. Koltun, “An empirical evaluation of generic convolutional and recurrent networks for sequence modeling,” arXiv preprint arXiv:1803.01271, 2018.

[50] T. Kristiansen, “Forecasting nord pool day-ahead prices with an autoregressive model,” Energy Policy, vol.49, pp.328–332, 2012.

[51] P. Poggi, M. Muselli, G. Notton, C. Cristofari, and A. Louche, “Forecasting and simulating wind speed in corsica by using an autoregressive model,” Energy conversion and management, vol.44, no.20, pp.3177–3196, 2003.

[52] F. Khan, A. Saeed, and S. Ali, “Modelling and forecasting of new cases, deaths and recover cases of covid-19 by using vector autoregressive model in pakistan,” Chaos, Solitons & Fractals, vol.140, p.110189, 2020.

[53] W. Wang, Y. Shi, G. Lyu, and W. Deng, “Electricity consumption prediction using xgboost based on discrete wavelet transform,” DEStech Transactions on Computer Science and Engineering, no.aiea, 2017.

[54] M. Alim, G.H. Ye, P. Guan, D.S. Huang, B.S. Zhou, and W. Wu, “Comparison of arima model and xgboost model for prediction of human brucellosis in mainland china: a time-series study,” BMJ open, vol.10, no.12, p.e039676, 2020.

[55] T. Chen and C. Guestrin, “Xgboost: A scalable tree boosting system in proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining,(pp. 785–794),” New York, NY, USA: ACM, vol.10, no.2939672.2939785, 2016.

[56] X. Sun, M. Liu, and Z. Sima, “A novel cryptocurrency price trend forecasting model based on lightgbm,” Finance Research Letters, vol.32, p.101084, 2020.

[57] D. Wang, Y. Zhang, and Y. Zhao, "Lightgbm: an effective mirna classification method in breast cancer patients," Proceedings of the 2017 International Conference on Computational Biology and Bioinformatics, pp.7–11, 2017.

[58] P. Hewage, A. Behera, M. Trovati, E. Pereira, M. Ghahremani, F. Palmieri, and Y. Liu, "Temporal convolutional neural (tcn) network for an effective weather forecasting using time-series data from the local weather station," Soft Computing, vol.24, no.21, pp.16453–16482, 2020.

[59] K. Lin, S. Sheng, Y. Zhou, F. Liu, Z. Li, H. Chen, C.Y. Xu, J. Chen, and S. Guo, "The exploration of a temporal convolutional network combined with encoder-decoder framework for runoff forecasting," Hydrology Research, vol.51, no.5, pp.1136–1149, 2020.

[60] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G.S. Corrado, A. Davis, J. Dean, M. Devin, *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," arXiv preprint arXiv:1603.04467, 2016.

Author's research achievements

Paper

(A) **Academic journal**

[**A-1**] Y. ZHANG, M. Yamamoto, G. Suzuki, and H. Shioya, "Collaborative Forecasting and Analysis of Fish Catch in Hokkaido From Multiple Scales by Using Neural Network and ARIMA Model," *IEEE Access*, vol. 10, pp. 7823–7833, 2022. (2022 IF:3.476).

[**A-2**] Y. ZHANG, G. Suzuki, and H. Shioya, "Prediction and Detection of Sewage Treatment Process Using N-BEATS Autoencoder Network," *IEEE Access*, vol. 10 , pp. 112594–112608, 2022. (2022 IF:3.476).

(B) **International conference**

[**B-1**] Y. ZHANG, H. Shioya, and M. Wada, "Predicting the Port-Catch Volume of Salmon at Eastern Hokkaido," in *The Third NPAFC-IYS Virtual Workshop*, 2021, pp. 142–143.

(C) **Domestic conference**

[**C-1**] Y. ZHANG, H. Shioya, and M. Wada, "A Study on Predicting the Port Fishing Volume by LSTM," 情報処理学会北海道支部, 情報処理北海道シンポシウム, 2018.

[**C-2**] Y. ZHANG, H. Shioya, and M. Wada, "ニューラルネットワークを使用した北海道東部の港湾漁獲量の予測," 人工知能学会, 人工知能学会全国大会 JSAI2020, 2020.

[**C-3**] M. Yamamoto, Y. ZHANG, G. Suziki, M. Izumi, and H. Shioya, "サケの漁獲量の分析および 3D 可視化に関する検討," 第 20 回情報科学技術フォーラム（FIT2021）, 2021.

[**C-4**] Y. ZHANG, M. Yamamoto, G. Suzuki, and H. Shioya, "A Study on Collaborative Prediction and Analysis of Fish Catch in Hokkaido from Multiple Scales by Using Neural Network and ARIMA Model," インテリジェント・システム・シンポジウム 2021, 2021.

[**C-5**] Y. ZHANG, G. Suzuki, and H. Shioya, "Application of N-Beats-based Models for Forecasting and Early Warning in Sewage Treatment process," インテリジェント・システム・シンポジウム 2022, 2022.

(D) **Award**

[**D-1**] 第 20 回情報科学技術フォーラム（FIT2021） FIT 奨励賞 (2021 年 8 月)