CHAPTER 4
ANALYSIS AND DESIGN

4.1 Analysis

The problem in Luwak Coffee company is that they cannot predict the actual stock everyday and if they calculated manually, it will take a long time. Therefore, a program to predict item stock made to solve this problem. This program can do calculated faster than the manual.

Table below is data table Luwak Coffee Sales Stock from January 2017 until June 2018. This data table will be used in this project.

Table 4.1: Data Table Luwak Coffee Sales Stock

Date	Item Code	Item Name	Coffee Type	Coffee Packaging	Stock
2017-01-03	KLKH0001	Kopi Hitam (PCS)	Kopi Hitam	PCS	702
2017-01-03	KLWK0001	White Koffie (PCS)	Coffeemix	PCS	2136
2017-01-04	KLHC0001	High Class Dark Coffee (PCS)	Kopi High Class	PCS	2
2017-01-04	KLWK0002	White Koffie (BOX)	Coffeemix	BOX	52498
2017-01-04	KLKH0004	Kopi Hitam (RTG)	Kopi Hitam	RTG	16
2017-12-30	KLWK0002	White Koffie (BOX)	Coffeemix	BOX	170
2018-01-02	KLKH0001	Kopi Hitam (PCS)	Kopi Hitam	PCS	344
2018-03-08	KLCR0001	Creamer (KG)	Bahan	KG	50
...	...	...	...	...	...
2018-06-30	KLWK0003	White Koffie (BAG)	Coffeemix	BAG	213

Illustration 4.1: Data Chart

Can be seen from table and chart above, the stock everyday has different number and then it has high fluctuative rate. Therefore, this situation makes the company hard to provide stock everyday.

The calculated method use the Backpropagation algorithm. This algorithm can do calculated with the historical data and can correcting the error. The historical data is taken from the company in excel form and then converted to csv format. After that, the file csv will be inputed in java using this program.

Step by step will be done by researchers:

1. Read CSV File

   In this step will read the CSV file and then divided into two parts: training data and testing data. Data from January 2017 until Desember 2017 used as training data and Januari 2018 until June 2018 used as testing data. The data will used Linked List for data structure.

2. Data Normalization

   The data that already stored in Linked List will do normalization. Before doing normalization, the data that has word form will be changed into number. In order to change into number, the data will distinct to make

easier data normalization. In this step use Min-Max Normalization. The purpose of this step is to convert all number or some words into number with the range from 0 until 1 for every data every parameter.

Formula for Min-Max Normalization

Illustration 4.2: Min-Max Normalization

z : result of normalization

x : data that will be normalized

min(x) : minimum value in every parameter

max(x) : maximum value in every parameter

3. Backpropagation Algorithm process

Illustration 4.3: Backpropagation Structure

Step 1: Initialize all weight with random numbers

Input weight (Vij) : random numbers between -0,5 to 0,5

Hidden weight (Wjk) : random numbers between -0,5 to 0,5

Input bias weight (V0j) : random numbers between -0,5 to 0,5

Hidden bias weight (W0k) : random numbers between -0,5 to 0,5

Step 2: For each training data do step 3 until 8

Step 3: Calculate all output hidden unit

Illustration 4.4: Calculation Hidden Unit Formula

Step 4: Calculate Activation Function of all hidden unit

Illustration 4.5: Activation Function Hidden Unit Formula

Step 5: Calculate output unit

Illustration 4.6: Calculation Output Unit Formula

Step 6: Calculate Activation Function of output unit

Illustration 4.7: Activation Function Output Unit Formula

Step 7: Calculate δ output unit based output

Illustration 4.8:Calculation δ Output Formula

Calculate the rate hidden weights with learning rate α

Illustration 4.9: Δ Hidden Weight Formula

Step 8: Do the same calculate δ for hidden layer

Illustration 4.10: Calculation δ Hidden Unit Formula

Illustration 4.11: Δ Input Weight Formula

Calculate rate weight input and weight bias

Step 9: Calculate all weight changes

$$w_{kj}\ (baru) = w_{kj}\ (lama) + \Delta w_{kj}$$

Illustration 4.12: Weight Update Formula

Step 10: Check the loss with the square loss function

Illustration 4.13: Square Loss Function

Step 11: Repeated the step 3-10 until loss < targetError

Step 12: Calculate using another hidden layer (3,5 and 7)

4.2 Desain

At this stage, the researcher will explain the design of the program. This design will discuss the program process from beginning to the end. The program is created using Java Programming Language.

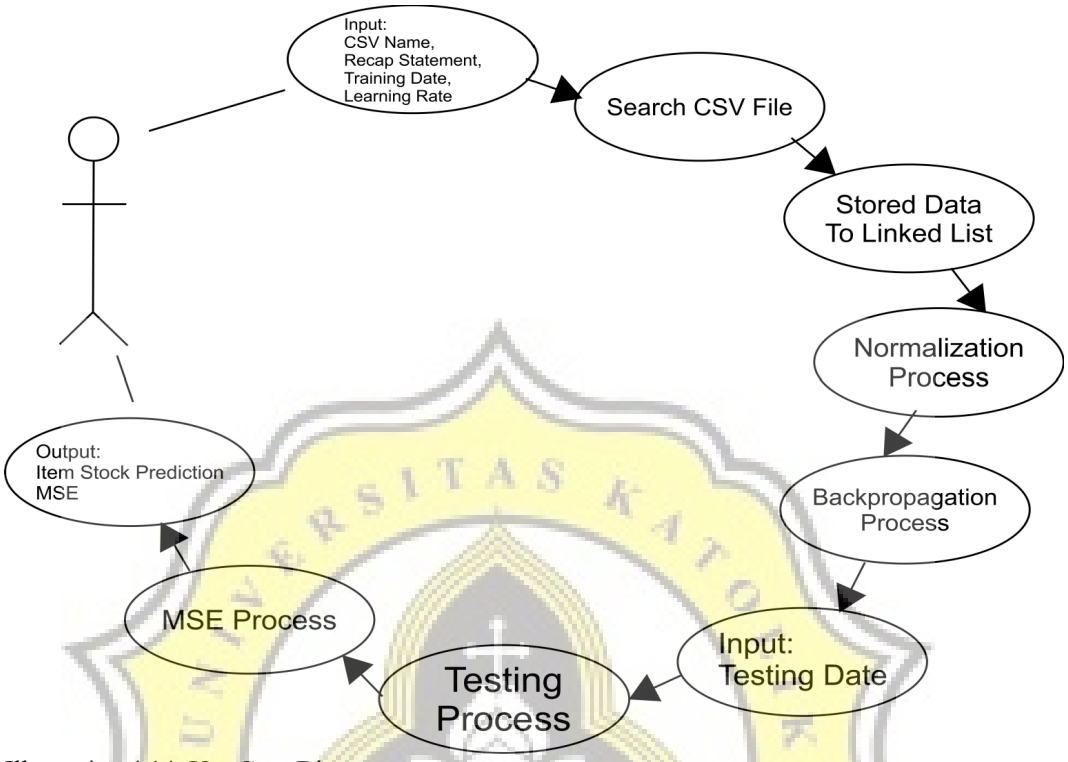

Illustration 4.14: Use Case Diagram

From Illustration above, the program will ask the user to input CSV Filename, Recap Statement (Per Day or Per Month or Per Year), Training Date (Date From and Date To) and Learning Rate. After that, the program will search CSV File based on user input and then the data will be stored in Linked List. The stored data will be normalized and then the program will do Backpropagation process. The final result is the item stock prediction and RMSE every hidden layer. Illustration below is the flow chart of Backpropagation Algorithm.

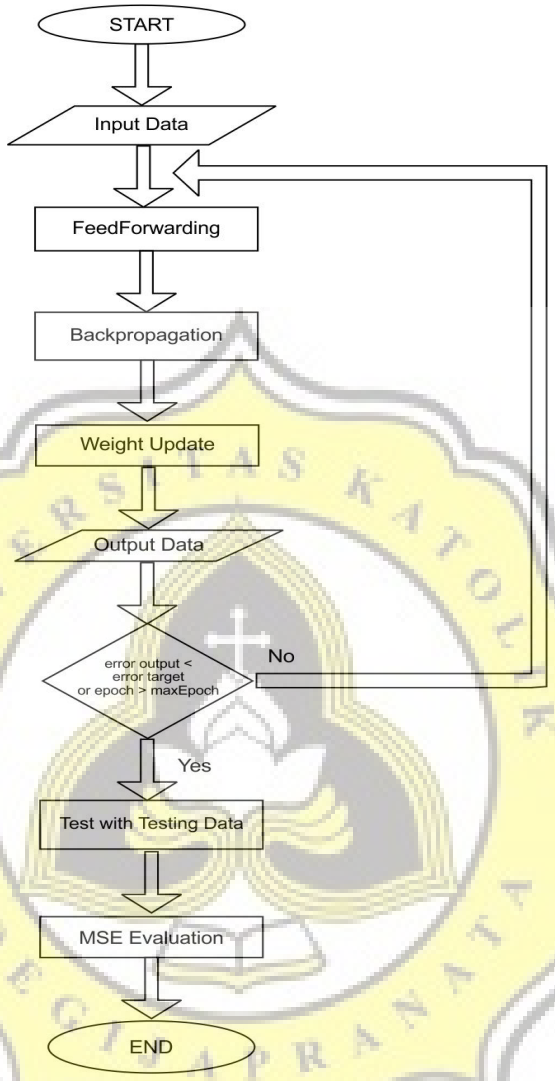

Illustration 4.15: Backpropagation Flow Chart

The first step from flow chart above is Feedforward. In Feedforward, the program will generate some random weight numbers from -0,5 to 0,5. The training data will be inputed as X1 until X4. Then, the inputed data will be proceed output through Feedforward phase.

After Feedforward phase, the output will have some loss compared to actual data. Therefore, the algorithm will learn from the loss and counting the delta weight for each output in hidden layer. The algorithm will correct the weight

by adding the old weight with the delta weight. This calculation will be repeated until reach the target error or reach the maximum number of iteration.

After training the data, the program will test the testing data only on Feedforward phase. Next, the output prediction will be count using RMSE evaluation to get the average of error.