

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1. Saham**

Saham adalah bukti dari kepemilikan suatu bagian perusahaan, yang berarti ketika membeli suatu emiten saham maka sudah memiliki sebagian dari perusahaan yang dibeli tersebut. Dengan zaman yang sudah serba digital, tidak ada lagi yang bernama sertifikat saham sebagai bukti kepemilikan saham. Ketika membeli saham, maka pembeli tersebut tidak mendapat bukti sertifikat secara fisik (Wira, 2021). Saham memiliki aset yang ada di sekitar, banyak perusahaan publik yang menghasilkan suatu produk yang digunakan sehari-hari seperti pada sektor pangan, keuangan, dan juga infrastruktur (Putra, 2018). Saham dapat menghasilkan suatu keuntungan dengan dua cara, yaitu memanfaatkan kenaikan harga saham atau yang biasa disebut *capital gain* dan mendapatkan bagi hasil dari perusahaan yang sahamnya dimiliki oleh pembeli yang disebut *dividen* (Wira, 2021).

##### **2.1.1. Indeks Pasar Saham**

Pada dunia pasar saham terdapat sebuah indikator yang menilai performa dari saham-saham yang terdaftar pada Bursa Efek Indonesia. (Handini & Astawinetu, 2020)

###### **2.1.1.1. Indeks LQ45**

Indeks LQ45 ini memiliki likuiditas yang cukup tinggi dan umumnya memiliki *value* yang cukup besar dan isi dari indeks tersebut merupakan 45 saham yang telah terdaftar di Bursa Efek Indonesia. Saham-saham pada indeks LQ45 ini dilakukan pergantian setiap enam bulan sekali. Untuk masuk ke dalam indeks LQ45 terdapat seleksi yang harus dilalui oleh saham-saham tersebut supaya masuk ke dalam kriteria pemilihan saham indeks LQ45. Kriteria untuk lolos pada indeks LQ45 sebagai berikut.

1. Berada di urutan 60 terbesar dalam total transaksi saham di pasar dengan melihat transaksi selama 12 bulan terakhir,
2. Terurut berdasarkan kapitalisasi pasar,

3. Terdaftar di Bursa Efek Indonesia setidaknya paling sedikit selama 3 bulan,
4. Emiten saham mempunyai laporan keuangan, frekuensi dan jumlah hari transaksi yang cukup baik di pasar saham.

Jika terdapat emiten saham yang sudah tidak memenuhi kriteria tersebut maka akan digantikan oleh saham perusahaan lain yang sudah memenuhi kriteria indeks LQ45 (Handini & Astawinetu, 2020).

#### **2.1.1.2. Jakarta Islamic Indeks ( JII )**

Indeks saham ini terdiri dari 30 saham yang sudah sesuai syariah islam dan menjadi sebuah tolak ukur untuk kinerja investasi saham syariah. Untuk kriteria pemilihan hampir sama seperti indeks LQ45, hanya saja emiten saham yang terdaftar lebih disesuaikan dengan syariah islam, yang dimana perusahaan tersebut tidak melakukan kegiatan judi, dan juga bukan sebuah usaha yang memperjual belikan atau memproduksi makanan dan minuman yang haram.

#### **2.1.2. Analisis Fundamental dan Analisis Teknikal**

Sebuah analisis perlu dilakukan dalam dunia saham, supaya tidak menjadi judi karena tidak menguasai ilmu bisnis tersebut tetapi langsung terjun tanpa adanya persiapan apapun.

##### **2.1.2.1. Analisis Fundamental**

Analisis fundamental merupakan suatu strategi analisis yang dilakukan terhadap suatu perusahaan dengan berdasarkan fundamental ekonomi perusahaan atau emiten saham tersebut. Umumnya teknik analisis itu digunakan oleh investor karena cocok untuk membuat suatu keputusan pemilihan saham perusahaan yang akan dibeli untuk investasi jangka panjang. Teknik analisis ini menggunakan laporan keuangan yang harus dilihat oleh calon investor dan juga harus bersabar untuk investasi jangka panjang pada perusahaan yang dipilih (Putra, 2018).

### **2.1.2.2. Analisis Teknikal**

Analisis teknikal merupakan teknik yang sering digunakan oleh para *trader* saham. Teknik analisis ini mempelajari data pasar di masa lampau dengan melihat grafik pergerakan harga dan volume transaksi saham perusahaan. Informasi yang bersifat baru dan mengikuti suatu tren akan berpengaruh terhadap harga saham itu sendiri. Sehingga, teknik analisis ini digunakan oleh para *trader* untuk mengetahui pada waktu kapan mereka harus *take profit*, *cut loss*, *hold* untuk mengurangi resiko kerugian yang banyak (Putra, 2018).

## **2.2. Peramalan ( *forecasting* )**

Peramalan merupakan ilmu yang dimanfaatkan untuk melakukan perkiraan kejadian - kejadian yang akan terjadi di masa mendatang, data di masa lalu sangat berperan terhadap ilmu peramalan ini karena data tersebut akan digunakan untuk peramalan di masa mendatang dengan bentuk model matematis. Peramalan juga didefinisikan sebagai suatu dugaan terhadap kondisi yang akan datang, data yang sering digunakan umumnya adalah data deret waktu historis. Hal yang harus diperhatikan dalam proses peramalan yaitu data harus relevan dengan apa yang akan diramal dan pemilihan teknik atau model peramalan yang tepat agar hasil peramalan tersebut dapat bermanfaat (Kadim, 2017).

### **2.2.1. Langkah Sistem Peramalan**

Proses peramalan memerlukan sebuah langkah dasar yang harus diikuti supaya mendapatkan hasil yang memuaskan, yaitu sebagai berikut (Kadim, 2017).

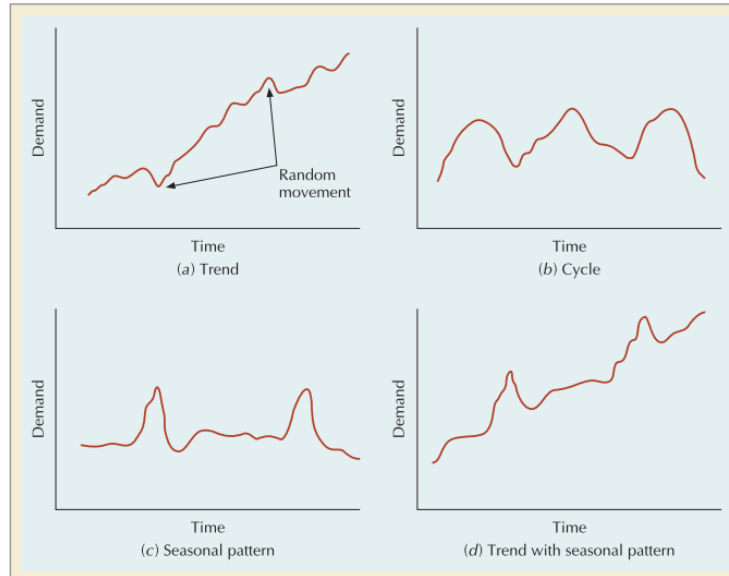
1. Menetapkan tujuan
2. Memilih unsur yang diramal
3. Menentukan jangka waktu peramalan
4. Memiliki tipe model untuk peramalan
5. Mengumpulkan data untuk peramalan
6. Membuat model peramalan

## 7. Mengimplementasi hasil peramalan

### 2.2.2. Peramalan Serial Waktu

Analisis pada data serial waktu harus diawali oleh memplot data dari jangka atau skala waktu tertentu dan dipelajari sehingga dapat melihat bentuk pola data waktu. Pola data waktu tersebut dikelompokkan sebagai berikut (Kadim, 2017).

1. Konstan yang berarti data tersebut secara rata-rata dikatakan stabil membentuk suatu garis lurus horizontal yang biasanya berada di jangka pendek atau menengah.
2. *Trend* merupakan data jangka panjang yang datanya tidak stabil, terdapat kenaikan dan penurunan dalam data tersebut dari waktu ke waktu.
3. *Seasonal* terdapat pola yang berulang-ulang secara teratur dalam setiap waktu atau periode tertentu, umumnya dipengaruhi oleh faktor iklim atau faktor manusia.
4. Siklus hampir sama dengan *seasonal* tetapi siklus ini memiliki durasi waktu yang lebih panjang dan dipengaruhi oleh fluktuasi ekonomi jangka panjang.
5. Residu merupakan suatu titik data yang disebabkan oleh situasi yang tidak lazim dan data tersebut tidak terdapat suatu pola yang dapat diprediksi oleh model-model karena bersifat acak.



**Gambar 2.1. Grafik Pola Data Sumber : (Russel & Taylor, 2011)**

### 2.3. Metode Autoregressive Integrated Moving Average (ARIMA)

ARIMA atau metode Box-Jenkins yang dikembangkan oleh George Box dan Gwilym Jenkins untuk melakukan pemodelan analisis pada data *time series* (Whitten & Bentley, 2007). ARIMA melakukan analisis menggunakan data lampau atau masa lalu ( $Z_{t-1}$ ) dan juga data masa kini ( $Z_t$ ).

ARIMA terdiri dari tiga unsur yaitu  $AR(p)$ ,  $I(d)$ , dan  $MA(q)$ , yang dimana  $p, d, q$  tersebut adalah bilangan bulat tidak negatif.  $ARIMA(p, d, q)$  mempunyai tiga klasifikasi model lainnya, yaitu  $AR(p)$ ,  $MA(q)$ , dan  $ARMA(p, q)$  (Wei, 2006).

#### 2.3.1. Autoregressiv ( $AR(p)$ )

$AR(p)$  adalah model dimana  $Z_t$  merupakan fungsi dari data di masa yang lalu, yakni  $t-1, t-2, \dots, t-p$ . Persamaan tersebut sebagai berikut.

$$Z_t = \phi_0 + \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \dots + \phi_p Z_{t-p} + \alpha_t \quad (1)$$

Keterangan :

$Z_t$  = data pada waktu t

$Z_{t-p}$  = data pada waktu t-p

$\alpha_t$  = error pada periode t-p

$\phi_0$  = konstanta model AR

$\phi_p$  = koefisien dari  $Z_{t-p}$  pada model AR

Persamaan AR( $p$ ) juga memiliki bentuk khusus, sebagai berikut.

- a. Model AR(1) :  $Z_t = \phi_0 + \phi_1 Z_{t-1} + \alpha_t$
- b. Model AR(2) :  $Z_t = \phi_0 + \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \alpha_t$
- c. Model AR(3) :  $Z_t = \phi_0 + \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \phi_3 Z_{t-3} + \alpha_t$

### 2.3.2. Moving Average ( MA( $q$ ) )

MA( $q$ ) untuk memprediksi  $Z_t$  sebagai fungsi dari kesalahan prediksi di masa lalu dalam memprediksi  $Z_t$ . Persamaan tersebut sebagai berikut.

$$Z_t = \theta_0 + \alpha_t - \theta_1 \alpha_{t-1} - \theta_2 \alpha_{t-2} - \dots - \theta_p \alpha_{t-p} \quad (2)$$

Keterangan :

$Z_t$  = data pada waktu  $t$

$\alpha_{t-p}$  = error pada periode  $t-p$

$\alpha_t$  = error pada periode  $t$

$\theta_0$  = konstanta model MA

$\theta_p$  = koefisien dari  $Z_{t-p}$  pada model MA

Persamaan MA( $q$ ) juga memiliki bentuk khusus, sebagai berikut.

- a. Model MA(1) :  $Z_t = \theta_0 + \alpha_t - \theta_1 \alpha_{t-1}$
- b. Model MA(2) :  $Z_t = \theta_0 + \alpha_t - \theta_1 \alpha_{t-1} - \theta_2 \alpha_{t-2}$
- c. Model MA(3) :  $Z_t = \theta_0 + \alpha_t - \theta_1 \alpha_{t-1} - \theta_2 \alpha_{t-2} - \theta_3 \alpha_{t-3}$

### 2.3.3. Autoregressive Moving Average ( ARMA ( $p,q$ ) )

$$Z_t = \phi_0 + \phi_1 Z_{t-1} + \dots + \phi_p Z_{t-p} + \alpha_t - \theta_1 \alpha_{t-1} - \dots - \theta_p \alpha_{t-p} \quad (3)$$

Dapat juga ditulis :

$$\phi_p(B)Z_t = \theta_q(B)\alpha_t$$

Dimana :

$$\phi_p(B) = 1 - \phi_1 B + \dots + \phi_p B^p \text{ dan,}$$

$$\theta_q(B) = 1 - \theta_1 B + \dots + \theta_q B^q$$

B = Operator geser mundur

$Z_t$  = data pada waktu t

$Z_{t-p}$  = data pada waktu t-p

$\alpha_t$  = *error* pada periode t-p

$\phi_0$  = konstanta model AR

$\phi_p$  = koefisien dari  $Z_{t-p}$  pada model AR

$\theta_p$  = koefisien dari  $Z_{t-p}$  pada model MA

Persamaan ARMA(p,q) juga memiliki bentuk khusus, sebagai berikut.

$$\text{Model ARMA}(1,1) : Z_t = \phi_0 + \phi_1 Z_{t-1} + \alpha_t - \theta_1 \alpha_{t-1}$$

#### 2.3.4. Autoregressive Integrated Moving Average ( ARIMA (p,d,q) )

ARIMA dilakukan pada data stasioner atau telah dilakukan *differencing* sehingga data telah stasioner. Gabungan dari model ARMA(p,q) dan juga proses *differencing* membentuk ARIMA ini.

$$\phi_p(B)(1 - B)^d Z_t = \theta_0 + \theta_q(B) \alpha_t \quad (4)$$

Keterangan :

B = Operator geser mundur

$\theta_0$  = konstanta model MA

$\theta_p$  = koefisien dari  $Z_{t-p}$  pada model MA

$\phi_p$  = koefisien dari  $Z_{t-p}$  pada model AR

$Z_t$  = data pada waktu t

$\alpha_t$  = *error* pada periode t-p

Persamaan ARIMA(p,d,q) juga memiliki bentuk khusus, sebagai berikut.

Model ARIMA(1,1,1) :

$$\phi_1(B)(1 - B)^1 Z_t = \theta_0 + \theta_1(B) \alpha_t$$

$$(1 - \phi_1 B)(1 - B)Z_t = \theta_0 + (1 - \theta_1 B) \alpha_t$$

$$(1 - B - \phi_1 B + \phi_1 B^2)Z_t = \theta_0 + (1 - \theta_1 B) \alpha_t$$

$$(1 - (1 + \phi_1)B + \phi_1 B^2)Z_t = \theta_0 + (1 - \theta_1 B) \alpha_t$$

$$Z_t = (1 + \phi_1) Z_{t-1} - \phi_1 Z_{t-2} + \theta_0 + (1 - \theta_1 \alpha_{t-1}) \alpha_t$$

### 2.3.5. Auto Correlation Factor (ACF)

Dalam pencarian model *time series*, *auto correlation factor* (ACF) berperan penting karena ACF ini menentukan nilai  $\rho$ . ACF berfungsi menunjukkan besarnya korelasi atau suatu hubungan antara variabel pengamatan awal dan pengamatan pada waktu  $t$ . Persamaan untuk ACF dapat ditulis dalam persamaan (5).

$$\hat{\rho}_k = \frac{\sum_{t=1}^{n-k} (Z_t - \bar{Z})(Z_{t+k} - \bar{Z})}{\sum_{t=1}^n (Z_t - \bar{Z})^2} \quad (5)$$

Keterangan :

$\rho_k$  = Koefisien autokorelasi lag ke- $k$

$Z_t$  = data pada waktu  $t$

$\bar{Z}$  = rata – rata  $Z_t$

$n$  = jumlah data

### 2.3.6. Partial Auto Correlation Factor (PACF)

PACF berfungsi sebagai himpunan penunjuk yang menandakan besarnya korelasi parsial antara pengamatan awal dengan pengamatan pada waktu  $t$  yang nantinya akan menentukan nilai  $\rho$ . Persamaan untuk PACF dapat ditulis dalam persamaan (6).

$$\hat{\phi}_{k+1,k+1} = \frac{\hat{\rho}_{k+1} - \sum_{j=1}^k \hat{\phi}_{kj} \hat{\rho}_{k+1-j}}{1 - \sum_{j=1}^k \hat{\phi}_{kj} \hat{\rho}_j} \quad (6)$$

### 2.3.7. Uji Augmented Dickey – Fuller (ADF)

Dalam melakukan analisis pada data *time – series*, data yang akan digunakan harus stasioner terlebih dahulu. Stasioneritas tersebut dapat dilihat melalui



plot ACF yang grafiknya mendekati nol secara cepat, selain itu dapat diketahui dengan uji *Augmented Dickey – Fuller* (ADF). Uji ADF tersebut dilakukan untuk menentukan apakah data *time – series* perlu dilakukan tahapan yang dinamakan *differencing* atau tidak. Uji untuk ADF menggunakan persamaan (7). (Rosadi, 2012)

$$\Delta Y_t = \beta_1 + \beta_2 + \delta Y_{t-1} + \phi_i \sum_{i=1}^k \Delta Y_{t-1} + \varepsilon_t \quad (7)$$

Keterangan :

$\Delta Y_t$  = *different* pertama dari Y

$\beta_1$  = nilai konstan

$\beta_2$  = koefisien *trend*

$\delta$  = koefisien lag Y

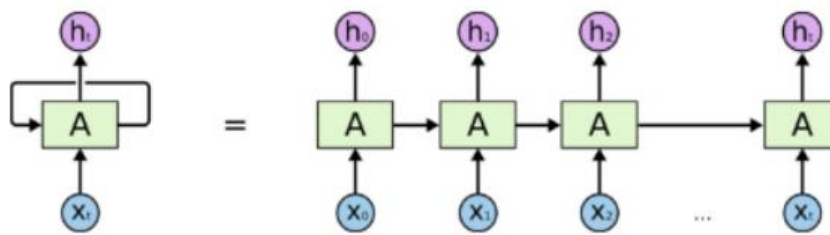
$\phi$  = koefisien *difference* lag Y

$\varepsilon$  = error

#### 2.4. Recurrent Neural Network

RNN merupakan jenis dari *neural network* yang memang secara alami sudah dibuat untuk mengolah data *time-series* dan data sekuensial (DiPietro & Hager, 2020). RNN telah terbukti memiliki kemampuan yang kuat untuk menangkap korelasi tersembunyi yang terjadi pada data dalam pengenalan suara, pemrosesan bahasa alami, dan aplikasi prediksi deret waktu. Hal tersebut sangat berguna untuk masalah pengurutan dengan mengoperasikan informasi input serta jejak informasi yang diperoleh sebelumnya karena koneksi yang berulang (Tian, Ma, Zhang , & Zhan, 2018).

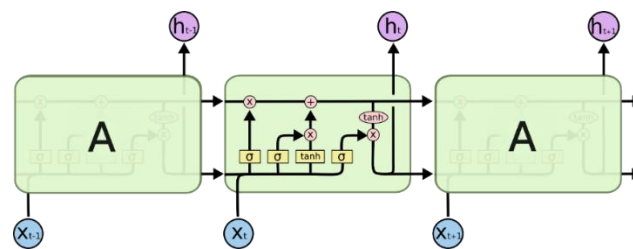
Model RNN ini tentu saja memiliki *input*, *output*, dan *hidden layer*. Aliran informasi yang berada dalam model RNN mengalir secara satu arah dari *input* menuju *hidden layer*. Proses pada *hidden layer* menangkap informasi pada waktu titik sebelumnya, dan *output* diturunkan dari waktu sekarang dan ingatan sebelumnya. RNN bekerja dengan baik ketika *output* dekat dengan *input* terkait karena informasi dari *node* sebelumnya diteruskan ke *node* berikutnya. Untuk gambar proses perulangan tersebut dapat dilihat pada Gambar 2.2 dibawah ini.



**Gambar 2.2. Looping dalam RNN Sumber : (Olah, 2015)**

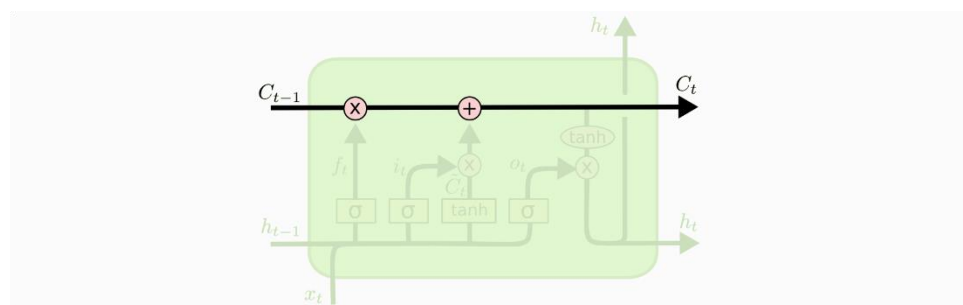
### 2.5. Long Short-Term Memory

Jaringan ini adalah sebuah jenis tersendiri dari RNN yang memang mengatasi masalah yang terdapat dalam RNN. Model LSTM ini dibuat dengan tujuan lebih praktis untuk mengingat informasi-informasi yang berada di jangka waktu yang lama. Untuk gambaran struktur LSTM dapat dilihat pada Gambar 2.3 dibawah ini.



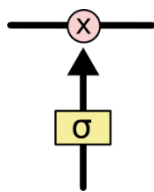
**Gambar 3.3. Arsitektur Looping LSTM Sumber : (Olah, 2015)**

Dalam LSTM memiliki *cell state* yang terletak pada bagian atas diagram yang berbentuk garis horizontal. Informasi akan mengalir secara mudah pada bagian *cell state*. Bentuk *cell state* dapat dilihat pada Gambar 2.4 dibawah ini.



**Gambar 4.4. Bagian Cell State Sumber : (Olah, 2015)**

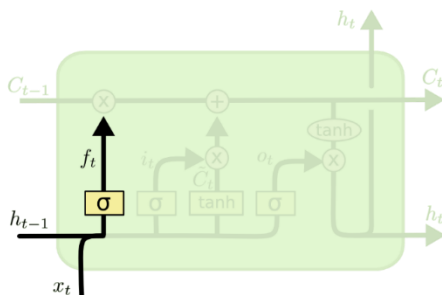
Pada bagian *cell state* memiliki struktur lain yang disebut *gates* seperti pada Gambar 2.5, yang dimana pada bagian tersebut bersifat opsional yang membiarkan informasi yang berkaitan lewat atau memberhentikan informasi tersebut.



**Gambar 5.5. Lapisan Sigmoid Sumber : (Olah, 2015)**

Lapisan sigmoid memberikan *output* berupa 0 dan 1 yang masing-masing bentuk tersebut memiliki arti jika bernilai 0 maka informasi tidak ada yang lewat dan jika nilai 1 maka informasi akan dibiarkan lewat atau diteruskan. LSTM memiliki *gates* sebanyak 3 jenis, yaitu *forget gate*, *input gate*, *output gate*.

Pada *forget gate* seperti namanya yaitu *forget* atau melupakan yang berarti pada layer ini akan ada beberapa informasi yang dihilangkan dari *cell state*. Jika dilihat pada Gambar 2.6, terdapat  $h_{t-1}$  dan  $x_t$  yang akan menampilkan 0 dan 1 untuk setiap *cell state*.



**Gambar 6.6. Forget Gate Sumber : (Olah, 2015)**

Persamaan yang digunakan dalam tahapan ini :

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (8)$$

Keterangan :

- |  |   |
|--|---|
| $f_t$ : <i>forget gate</i>                   | $h_{t-1}$ : nilai <i>output</i> sebelum orde ke-t |
| $\sigma$ : fungsi sigmoid                    | $x_t$ : <u>input saat orde ke-t</u>               |
| $W_f$ : nilai bobot untuk <i>forget gate</i> | $b_f$ : nilai <i>bias</i> pada <i>forget gate</i> |

Nilai bobot ( $W$ ) dapat dilihat pada persamaan 8.

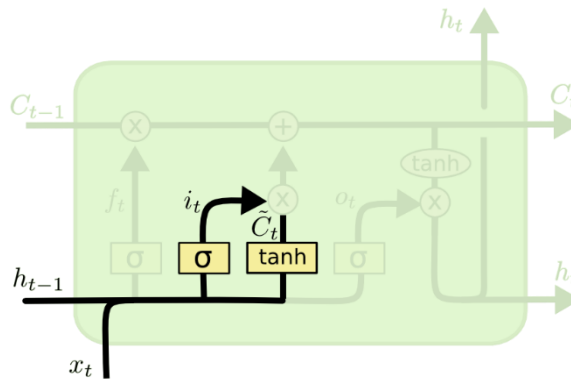
$$W = (-1\sqrt{d}, 1\sqrt{d}) \quad (9)$$

Keterangan :

W = bobot

d = Total Data

Saatnya untuk memutuskan informasi apa yang akan disimpan dalam status sel setelah menentukan informasi apa yang dikecualikan di gerbang lupa. Prosedur ini akan melalui dua tahap: gerbang input akan mengidentifikasi nilai input mana yang akan diubah, dan status sel akan menggunakan fungsi tanh untuk menambahkan nilai baru. Status sel kemudian diperbarui menggunakan nilai yang diperoleh dari gerbang gabungan dan input tanh. Gambar 2.7 menunjukkan ilustrasi langkah-langkah ini.



Gambar 7.7. Input gate dan tanh Sumber : (Olah, 2015)

Persamaan yang digunakan pada proses ini dijabarkan sebagai berikut

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (10)$$

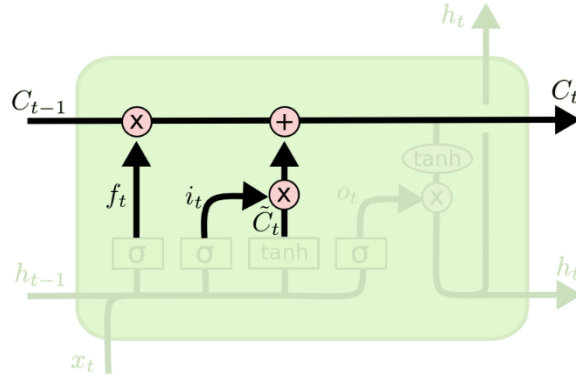
$$\bar{c}_t = \tanh(WC \cdot [h_{t-1}, x_t] + bC) \quad (11)$$

Keterangan :

- |   |                                     |
|---|-------------------------------------|
| $i_t$ : input gate                        | $h_{t-1}$ : nilai output sebelumnya |
| $\sigma$ : fungsi sigmoid                 | $x_t$ : input saat ini              |
| $W_i$ : nilai bobot untuk input gate      | $b_i$ : nilai bias pada input gate  |
| $WC$ : nilai bobot untuk cell state       | $\tanh$ : fungsi tanh               |
| $\bar{c}_t$ : nilai baru untuk cell state | $bC$ : nilai bias untuk cell state  |

Tahapan yang dilewati selanjutnya, yaitu melakukan *update* pada *cell state*. Terjadi sebuah proses perkalian *cell state* dengan *state* yang lawas melalui proses

penghilangan informasi yang dilakukan oleh *forget gate* pada tahapan sebelumnya, sehingga hasil yang diperoleh kemudian ditambahkan dengan nilai terbaru untuk *update state*. Tahapan ini dapat dilihat pada Gambar 2.8.



**Gambar 8.8. Update Cell State** Sumber : (Olah, 2015)

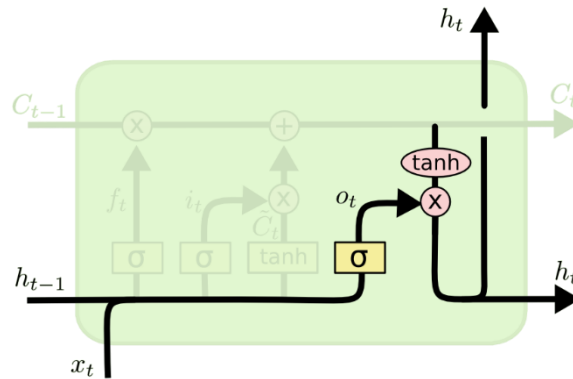
Persamaan yang digunakan pada tahap ini dijabarkan sebagai berikut

$$C_t = f_t \times C_{t-1} + i_t \times \bar{C}_t \quad (12)$$

Keterangan :

- |                                      |  |
|--------------------------------------|--|
| $C_t$ : nilai baru <i>cell state</i> | $C_{t-1}$ : nilai <i>cell state</i> sebelumnya |
| $f_t$ : nilai <i>forget gate</i>     | $i_t$ : nilai <i>input gate</i>                |

Proses terakhir yang dilewati pada model LSTM ini adalah menentukan *output*. Untuk menentukan *output*, lapisan sigmoid akan memutuskan bagian apa dari *cell state* yang akan dihasilkan. Selanjutnya, *cell state* tersebut dilakukan sebuah pengolahan melalui tanh dan dikalikan dengan output dari *sigmoid gate* supaya dapat menghasilkan *output* sesuai dengan yang sudah diputuskan. Proses ini dapat dilihat pada Gambar 2.9.



**Gambar 9.9.** *Output Layer* Sumber : (Olah, 2015)

Persamaan yang digunakan dalam tahap ini adalah :

$$ot = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (13)$$

Keterangan :

- |   |   |
|---|---|
| $ot$ : nilai <i>output gate</i>             | $h_{t-1}$ : nilai <i>output</i> sebelumnya        |
| $\sigma$ : fungsi sigmoid                   | $x_t$ : <i>input</i> saat ini                     |
| $W_o$ : nilai bobot untuk <i>ouput gate</i> | $b_o$ : nilai <i>bias</i> pada <i>output gate</i> |

Atau persamaan yang didapat untuk mendapatkan nilai output orde ke-t dapat diajabarkan pada persamaan berikut:

$$ht = ot \times \tanh (Ct) \quad (14)$$

Keterangan :

- |                                      |                       |
|--------------------------------------|-----------------------|
| $ht$ : nilai <i>output gate</i> ke-t | $\tanh$ : fungsi tanh |
| $ot$ : nilai <i>output gate</i>      |                       |

## 2.6. Normalisasi

Normalisasi disini bertujuan untuk meminimalisir tingkat *error* ketika digunakan untuk melakukan prediksi. Normalisasi *MinMax* merupakan salah satu teknik normalisasi yang sudah umum digunakan. Metode *Minmax* ini akan merubah data yang ada dan menghasilkan sebuah nilai dengan rentang 0 – 1. Rumus dari normalisasi *MinMax* dapat dilihat pada persamaan 14 (Aldi, Jondri, & Aditsania, 2018).

$$X_n = \frac{x - X_{min}}{X_{max} - X_{min}} \quad (15)$$

Keterangan :

$X_n$  = Hasil normalisasi data

$x$  = Data aktual

$X_{min}$  = Data terkecil

$X_{max}$  = Data terbesar

### 2.7. Mean Absolute Percentage Error (MAPE)

Kesalahan absolut untuk setiap periode dibagi dengan nilai pengamatan aktual untuk waktu itu untuk menentukan *mean absolute percentage error* (MAPE). Persentase kesalahan mutlak kemudian harus dirata-ratakan. Besarnya persentase variasi antara data aktual dan data prediksi dihitung dengan menggunakan pengukuran kesalahan yang dikenal sebagai MAPE. Untuk nilai MAPE diperoleh menggunakan persamaan (16). (S & W, 2015)

$$\frac{\sum_{t=1}^n \left| \left( \frac{A_t - F_t}{A_t} \right) 100 \right|}{n} \quad (16)$$

Keterangan :

$A_t$  = Permintaan aktual ke-t

$F_t$  = Hasil peramalan ke-t

$n$  = banyaknya data peramalan

### 2.8. Penelitian Terdahulu

Penulis mendapatkan referensi dari penelitian terkait yang sudah pernah dilakukan sebelumnya sebagai bahan acuan dalam pembuatan penelitian ini. Berikut beberapa penelitian terdahulu yang berkaitan dengan penelitian ini.

1. Wilda Yulia Rusyida dan Versiandika Yudha Pratama (2020) melakukan penelitian yang berjudul “Prediksi Harga Saham Garuda Indonesia di Tengah Pandemi *Covid-19* Menggunakan Metode ARIMA”. Penelitian tersebut bertujuan untuk melihat seberapa akurat metode ARIMA dalam

prediksi harga saham PT. Garuda Indonesia. Penelitian menggunakan data harga saham harian PT. Garuda Indonesia dari 22 April 2019 sampai 20 April 2020. Hasil penelitian menunjukkan bahwa model ARIMA terbaik adalah ARIMA (3,1,2) yang digunakan untuk memprediksi harga saham dari 21 April 2020 sampai 13 Juli 2020. Dari hasil analisis terlihat bahwa terdapat penurunan karena berhentinya sektor operasional saat masa pandemi *covid-19* (Rusyida & Pratama, 2020).

2. Dona Ayu Rezaldi dan Sugiman (2021) melakukan penelitian yang berjudul “Peramalan metode ARIMA Data Saham PT. Telekomunikasi Indonesia”. Penelitian tersebut bertujuan untuk mengetahui model ARIMA yang tepat untuk meramal harga saham penutup dari PT. Telekomunikasi Indonesia pada bulan Juni 2020 sampai Mei 2021. Data harga saham tersebut diperoleh dari *yahoo finance* sebanyak 84 data harga penutup. Hasil penelitian menunjukkan bahwa model ARIMA yang terbaik adalah ARIMA(0,2,1) untuk meramal harga penutup saham PT. Telekomunikasi Indonesia dengan nilai MSE sebesar 3.070 (Rezaldi & Sugiman, 2021).
3. Nur Fitriani Bintang Pradana dan Sri Lestanti (2020) melakukan penelitian yang berjudul “Aplikasi Prediksi Jangka Pendek Harga Bitcoin Menggunakan Metode ARIMA”. Tujuan dari penelitian ini untuk membantu memprediksi harga bitcoin yang tidak menentu kedepannya. Penelitian ini menggunakan metode ARIMA karena kemampuan metode tersebut dalam menghasilkan akurasi yang tinggi untuk prediksi dalam jangka waktu yang pendek. Hasil penelitian menunjukkan bahwa model ARIMA yang terbaik dalam penelitian tersebut adalah ARIMA(3,1,3) dengan evaluasi menggunakan *Mean Absolute Percentage Error* (MAPE) sehingga diperoleh nilai rata-rata MAPE sebesar 0,84 dan rentang nilai 1,34 pada prediksi hari pertama dan nilai 0.98 pada hari ketujuh (Pradana & Lestanti, 2020).
4. Efrike Sofyani Putri, dan Mujiono Sadikin (2021) melakukan penelitian yang berjudul “Prediksi Penjualan Produk Untuk Mengestimasi Kebutuhan Bahan Baku Menggunakan Perbandingan Algoritma LSTM dan ARIMA”.



Tujuan dari penelitian ini adalah untuk mengestimasi rasa yang paling laku di tahun yang akan datang. Pada penelitian ini membandingkan model ARIMA dengan LSTM. Hasil penelitian menunjukkan metode LSTM memiliki kinerja yang terbaik dengan range interval  $[-1,1]$  dan juga epoch = 100. Metode ini menghasilkan nilai RMSE rata-rata 0,22 selama uji coba untuk model LSTM dibandingkan dengan 60,21 untuk model ARIMA. Hasil rerata persentase error model antara nilai harian minimum 29,57% untuk metode LSTM dan 73% untuk model ARIMA menggunakan MAPE. Dari performansi yang diperoleh untuk masing-masing model dapat disimpulkan bahwa penggunaan model LSTM mengungguli ARIMA dalam hal ini (Putri & Sadikin, 2021)

5. Ardiyan Agusta, Iin Ernawati, dan Anita Muliawati (2021) melakukan penelitian yang berjudul “Prediksi Pergerakan Harga Saham Pada Sektor Farmasi Menggunakan Algoritma Long Short-Term Memory”. Penelitian ini bertujuan untuk memprediksi harga saham untuk perusahaan farmasi, yaitu PT. Kalbe Farma. Data yang digunakan dalam penelitian ini adalah data harga *closing* KLBF dari Juli 2018 sampai Juli 2020. Hasil penelitian menyatakan bahwa kombinasi parameter optimal yang digunakan oleh model LSTM didasarkan pada memvariasikan jumlah ukuran batch per epoch menggunakan jumlah epoch 200, ukuran batch 10, dan jumlah sel yang dioptimalkan oleh adam. Kombinasi ini menghasilkan nilai RMSE rata-rata 27.310 dan waktu komputasi rata-rata 322,28 detik (Agusta, Ernawati, & Muliawati, 2021)