

**PROPOSAL SKRIPSI**  
**PENERAPAN ALGORITMA LSTM ( *Long-Short Term Memory* )**  
**UNTUK PREDIKSI HARGA SAHAM**



**Disusun oleh :**  
**Rachmad Agung Pambudi**  
**160411100032**

**Dosen Pembimbing 1. Eka Mala Sari Rochman, S.Kom., M.Kom 19841104 200812 2003**  
**Dosen Pembimbing 2. Sri Herawati, S.Kom., M.Kom., 19830828 200812 2002**

**PROGRAM STUDI INFORMATIKA**  
**JURUSAN TEKNIK INFORMATIKA**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS TRUNOJOYO MADURA**  
**2020**

## LEMBAR PERSETUJUAN PROPOSAL

Nama : Rachmad Agung Pambudi  
NRP : 16.04.1.1.1.00032  
Bidang Minat : Kecerdasan Komputasional  
Program Studi : Informatika  
Jurusan : Teknik Informatika  
Dosen Pembimbing : 1. Eka Mala Sari Rochman, S.Kom., M.Kom.  
2. Sri Herawati, S.Kom., M.Kom.,  
Judul Tugas Akhir : Penerapan Algoritma LSTM (*Long-Short Term Memory*) Untuk Prediksi Harga Saham

Proposal ini telah disetujui di seminar pada Tanggal, ..... 2020

Dosen Pembimbing I

Dosen Pembimbing II

Eka Mala Sari Rochman, S.Kom., M.Kom.

NIP. 19840716 200812 2 001

Sri Herawati, S.Kom., M.Kom.,

NIP. 19830828 200812 2 002

Mengetahui,

Ketua Program Studi  
Teknik Informatika,

Menyetujui,

Koordinator Lab Riset  
Jurusan Teknik Informatika

Yoga Dwitya Pramudita, S.Kom., M.Cs.

NIP. 19840413 200812 1 002

Husni, S.Kom., M.Kom

NIP. 19790722 200312 1 001

## ABSTRAK

Saham merupakan salah satu pilihan dalam investasi yang menarik karena dapat menghasilkan keuntungan yang besar, jika dapat membaca pergerakan perubahan harga saham. Bagi pemula untuk memulai berinvestasi saham memiliki resiko kerugian yang sangat besar dikarenakan belum bisa membaca pergerakan harga saham. Untuk meminimalkan resiko kerugian diperlukan belajar membaca pergerakan harga saham dan perkembangan pasar modal yang merupakan indikator dalam masuk didunia investasi saham. Dengan perkembangan teknologi kecerdasan buatan dan pemrosesan prediksi saat ini, pergerakan harga saham dapat diidentifikasi dengan cara otomatis secara perhitungan matematis yang sangat kompleks. *Deep Learning* merupakan salah satu teknologi kecerdasan buatan yang memiliki akurasi pengenalan yang tinggi dengan data yang sangat banyak. Penelitian ini menggunakan analisis harga saham sebelumnya dalam suatu perusahaan, dengan teknik *Deep learning*, *Recurrent Neural Networks (RNN)* modul pemrosesan *Long-Short Term Memory (LSTM)* untuk melakukan prediksi harga saham. Fitur yang digunakan dalam pemrosesan prediksi yaitu harga tutup. Diharapkan dengan metode *Long-Short Term Memory (LSTM)* bisa memberikan performa terbaik.

Kata kunci: *Deep Learning*, *Long-Short Term Memory (LSTM)*, *Recurrent Neural Networks (RNN)*.

## DAFTAR ISI

ABSTRAK .....	ii
DAFTAR ISI .....	iii
DAFTAR GAMBAR .....	v
DAFTAR TABEL .....	vi
BAB I PENDAHULUAN .....	1
1.1 Latar Belakang .....	1
1.2 Perumusan Masalah .....	2
1.2.1 Permasalahan .....	2
1.2.2 Metode Usulan .....	2
1.2.3 Pertanyaan Penelitian .....	2
1.3 Tujuan dan Manfaat .....	2
1.3.1 Tujuan Penelitian .....	2
1.3.2 Manfaat Penelitian .....	2
1.4 Batasan-batasan .....	2
1.5 Sistematika Proposal .....	3
BAB II KAJIAN PUSTAKA .....	4
2.1 Saham .....	4
2.1.1 Pengertian Saham .....	4
2.1.2 Saham PT Semen Indonesia (Persero) Tbk (SMGR) .....	4
2.2 <i>Recurrent neural network</i> (RNN) .....	5
2.3 <i>Long Short Term Memory</i> (LSTM) .....	8
2.3.1 Algoritma pelatihan <i>Long Short Term Memory Networks</i> .....	12
2.3.2 Feedforward .....	14
2.3.3 Backpropagation Through Time (BPTT) .....	15
2.3.4 Mencari Bobot dari proses backpropagation urutan (sequence) .....	18
2.3.5 Optimasi untuk update bobot Gradient Descent .....	19
2.4 Penelitian Terkait .....	21
BAB III METODE USULAN .....	23
3.1 Metode .....	23
3.1.1 Rancangan Arsitektur LSTM .....	23
3.1.2 Rancangan Arsitektur Sistem .....	25
3.2 Analisa Kebutuhan .....	26
3.2.1 Kebutuhan <i>Hardware</i> .....	26
3.2.2 Kebutuhan <i>Software</i> .....	26

3.3 Analisa Input .....	26
3.4 Analisa Proses .....	27
3.5 Analisa Output .....	27
3.6 Desain Rancangan.....	28
3.6.1 Flowchart Diagram .....	28
3.6.1 Flowchart Multiple Train-Test Split .....	30
3.6.2 Flowchart Tahap Pelatihan LSTM.....	30
3.6.3 Flowchart Tahap Pengujian LSTM.....	31
3.7 Skenario Uji Coba dan Evaluasi.....	32
3.8 Pembuatan Laporan Tugas Akhir.....	32
DAFTAR PUSTAKA .....	34

## DAFTAR GAMBAR

Gambar 1 Arsitektur pada RNN.....	5
Gambar 2 Memori pada RNN .....	5
Gambar 3 Memori pada LSTM.....	6
Gambar 4 Arsitektur Hidden Layer .....	9
Gambar 5 Status sel LSTM.....	9
Gambar 6 Gerbang LSTM .....	10
Gambar 7 Penjagaan gerbang (Keep gate) $ft$ LSTM .....	10
Gambar 8 Menambahkan informasi baru (write gate) LSTM .....	11
Gambar 9 Update cell state LSTM .....	11
Gambar 10 Membaca keputusan (Output) LSTM .....	12
Gambar 11 Arsitektur LSTM forward .....	23
Gambar 12 Penjabaran dari Hidden Layer LSTM .....	23
Gambar 13 Arsitektur LSTM Backward.....	24
Gambar 14 Penjabaran dari Hidden Layer LSTM Backward.....	24
Gambar 15 Diagram IPO Arsitektur Sistem .....	25
Gambar 16 Flowchart Sistem Keseluruhan .....	28
Gambar 17 Flowchart Multiple Train-Test Split .....	30
Gambar 18 Flowchart pelatihan LSTM .....	31
Gambar 19 Flowchart pengujian LSTM .....	32

## DAFTAR TABEL

Table 1. Tabel Penelitian Terkait .....	21
Tabel 41 Uji coba dengan pembagian data .....	32
Tabel 42. Tabel Perkiraan Jadwal .....	33

# BAB I PENDAHULUAN

## 1.1 Latar Belakang

Pada era digital saat ini, Investasi saham di pasar modal pada setiap negara merupakan aset yang sangat penting bagi setiap perusahaan didunia. Sebab secara langsung maupun tidak, investor dari seluruh dunia dapat memberikan dampak terhadap ekonomi pada negara tempat berinvestasi. Saham adalah surat berharga yang dikeluarkan oleh perusahaan BUMN ataupun Swasta yang berbentuk Perseroan Terbatas (PT). Pemilik saham juga memiliki hak untuk memiliki sebagian dari perusahaan. Banyak masyarakat dari yang remaja, dewasa, hingga lansia dapat membeli atau menanamkan modalnya kedalam pasar modal untuk mengharapkan keuntungan. Hal ini senada dengan pendapat Widoatmodjo yaitu ada beberapa keuntungan dalam membeli saham adalah Capital Gain, deviden, dan saham juga dapat dijaminkan ke bank sebagai agunan untuk memperoleh kredit [1].

Prediksi (peramalan) adalah proses dalam menduga atau memperkirakan kejadian yang terjadi diwaktu mendatang dengan memanfaatkan informasi atau data yang relevan pada waktu sebelumnya (historis) melalui metode ilmiah [2].

Kegiatan untuk memprediksi harga saham banyak dilakukan dengan berbagai teknik pemodelan dan penerapan metode tertentu. Pada penelitian sebelumnya telah dilakukan perbandingan metode untuk memprediksi harga saham sektor teknologi Amerika dengan menggunakan metode Backpropagation, SVM, dan LSTM (*Long-Short Term Memory*) untuk backpropagation menghasilkan ketepatan 68.649% dengan epochs 10 dan 67,434% dengan epochs 100, untuk SVM menghasilkan ketepatan 66.9823% dengan epochs 10 dan 67.1212% dengan epochs 100, LSTM menghasilkan ketepatan 68.51635% dengan epochs 10 dan 69.04171% dengan epochs 100 [3].

*Long Short Term Memory networks* (LSTM) merupakan sebuah evolusi dari arsitektur RNN, dimana pertama kali diperkenalkan oleh Hochreiter dan Schmidhuber (1997) [4]. RNN memiliki kekurangan yaitu dapat dilihat pada inputan yang memiliki rentang informasi yang sangat besar sehingga ketika block output ( $h$ ) memerlukan informasi yang relevan maka RNN tidak dapat menghubungkan informasi karena memori lama yang tersimpan akan tidak berguna dengan seiringnya berjalannya waktu karena akan tergantikan dengan memori baru, permasalahan ini ditemukan oleh *Bengio, et al.* (1994). Berbeda dengan RNN, LSTM tidak memiliki kekurangan



tersebut karena LSTM (*Long-Short Term Memory*) dapat mengatur memori di setiap masukannya dengan menggunakan memory cells dan gate units [5].

Penelitian ini akan menerapkan metode LSTM (*Long-Short Term Memory*) karena pada penelitian sebelumnya mendapatkan hasil yang baik dalam proses prediksi. Pada penelitian ini menggunakan metode LSTM dengan objek harga penutupan PT Semen Indonesia (Persero) Tbk, dengan tujuan memprediksi harga penutupan dalam jangka waktu 5 hari berikutnya.

## **1.2 Perumusan Masalah**

### **1.2.1 Permasalahan**

Permasalahan yang sering terjadi dalam sistem prediksi ialah keterbatasan data faktor yang mempengaruhi kinerja saham saat ini dan juga keakuratan dalam memprediksi suatu saham tertentu.

### **1.2.2 Metode Usulan**

Penelitian ini mengusulkan Penerapan metode LSTM (*Long-Short Term Memory*) untuk prediksi harga saham.

### **1.2.3 Pertanyaan Penelitian**

1. Berapa akurasi tingkat error yang dihasilkan dalam memprediksi harga saham PT Semen Indonesia (Persero) Tbk?

## **1.3 Tujuan dan Manfaat**

### **1.3.1 Tujuan Penelitian**

1. Mengetahui tingkat error yang dihasilkan dalam memprediksi Harga saham PT Semen Indonesia (Persero) Tbk dengan metode LSTM (*Long-Short Term memory*)

### **1.3.2 Manfaat Penelitian**

1. Peneliti mengetahui tingkat error yang dihasilkan dari metode *LSTM (Long-Short Term memory)*. untuk sistem prediksi.

## **1.4 Batasan-batasan**

Batasan masalah pada penelitian ini adalah sebagai berikut :

1. Data yang akan digunakan pada penelitian ini adalah data saham disitus Alphavantage(<https://www.alphavantage.co/>) disertakannya API ( Application Programming Interface) yang ada di situs tersebut dan situs pendukungnya yahoo finance.
2. Data yang akan digunakan pada penelitian ini adalah harga saham PT Semen Indonesia (Persero) Tbk saat harga penutupan.
3. Metode yang digunakan LSTM (*Long-Short Term Memory*)
4. Data saham tidak mempertimbangkan hari libur.
5. Data yang dipakai 1500 data, dari tanggal (09/01/2014) sampai (09/01/2020)

## **1.5 Sistematika Proposal**

Sistematika penulisan Proposal ini terdiri dari beberapa bab, yaitu :

### **BAB I PENDAHULUAN**

Pada bab ini membahas tentang latar belakang, rumusan masalah, tujuan, manfaat, batasan masalah, serta sistematika penulisan laporan.

### **BAB II KAJIAN PUSTAKA**

Pada bab ini membahas tentang landasan teori dan penelitian sebelumnya.

### **BAB III METODE USULAN**

Pada bab ini membahas tentang metode yang diusulkan (algoritma), arsitektur (Input, Proses, Ouput), Data penelitian, tahapan penelitian, serta rencana pengujian.

### **REFERENSI**

Berisi tentang daftar jurnal-jurnal atau referensi yang dijadikan acuan dalam penelitian ini.

## **BAB II KAJIAN PUSTAKA**

### **2.1 Saham**

#### **2.1.1 Pengertian Saham**

Saham merupakan sebuah surat berharga yang dikeluarkan oleh sebuah perusahaan, baik Swasta maupun BUMN yang mencari modal dari bursa efek dengan cara menerbitkan efek atau bisa disebut emiten. Pemilikan saham adalah juga pemilik sebagian dari perusahaan [6].

#### **2.1.2 Saham PT Semen Indonesia (Persero) Tbk (SMGR)**

Pada tanggal 17 Mei 1991 PT Semen Gresik Tbk mendapatkan persetujuan Badan Pengawas Pasar Modal (Bapepam) lewat surat bernomor S-622/PM/1991 menerbitkan prospektus IPO pada 20 Mei 1991 dengan menawarkan sebanyak 40 juta saham pada harga Rp 7.000 per saham dengan menargetkan modal segar Rp 280 miliar. Dari dana segar tersebut sebagian dialokasikan ke pembangunan pabrik semen baru di wilayah Tuban Jawa Timur. Dan sebagian saham digunakan meningkatkan kapasitas produksi dari 1 ton menjadi 1,3 juta ton pertahun.

Pada tanggal 7 Januari 2013 PT Semen Gresik (Persero) Tbk secara resmi berubah nama menjadi PT Semen Indonesia (Persero) Tbk sebagai holding dari tiga entitas anak usaha, yaitu PT Semen Gresik, PT Semen Tonasa, dan PT Semen Padang.

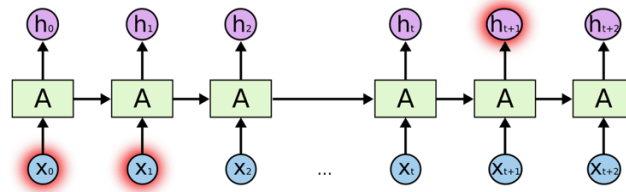
#### **2.1.3 Harga Penutupan**

Harga Penutupan (*Closing Price*) adalah harga yang muncul saat bursa tutup, harga pasar saham yang saat itu sedang berlaku akan menjadi harga penutupan untuk harga di hari itu. Harga penutupan saham hari itu juga akan menjadi acuan harga pembukaan untuk keesokan harinya.

#### **2.1.4 Prediksi Saham**

Prediksi (peramalan) adalah proses dalam menduga atau memperkirakan kejadian yang terjadi diwaktu mendatang dengan memanfaatkan informasi atau data yang relevan pada waktu sebelumnya (historis) melalui metode ilmiah [2]

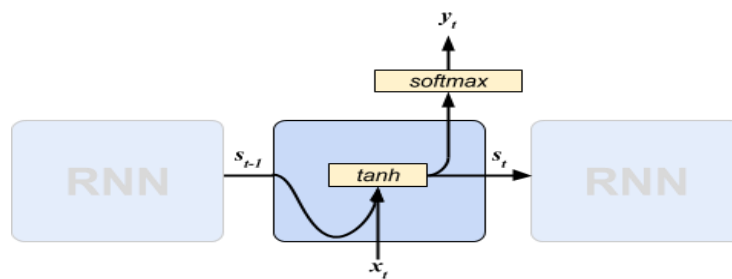
## 2.2 Recurrent neural network (RNN)



Gambar 1 Arsitektur pada RNN

*Recurrent neural network* (Vanilla RNN) tidak dapat belajar menghubungkan informasi jika dalam pembelajarannya memiliki langkah atau data yang terlalu banyak, karena menyebabkan penyimpanan lama yang tersimpan akan tidak berguna dengan seiringnya berjalannya waktu karena akan tertimpa atau tergantikan dengan memori baru (*Vanishing Gradient Problem*), permasalahan ini ditemukan oleh Bengio, et al. (1994). Berbeda dengan RNN, *Long Short Term Memory (LSTM)* tidak memiliki kekurangan tersebut (*Vanishing Gradient Problem*), karena *Long Short Term Memory (LSTM)* dapat mengatur memori pada setiap masukannya dengan menggunakan *memory cells* dan *gate units* [5]. Jenis jaringan saraf ini berguna ketika mengingat urutan data yang panjang dan tidak bergantung pada dataset jendela yang tertinggal sebagai input [3].

Pada pemrosesan dalam suatu modul *Recurrent neural network (RNN)* memiliki lapis tanh saja dan jika output dikehendaki maka akan dilewatkan dengan fungsi aktivasi softmax untuk mendapatkan output. Seperti Gambar 2

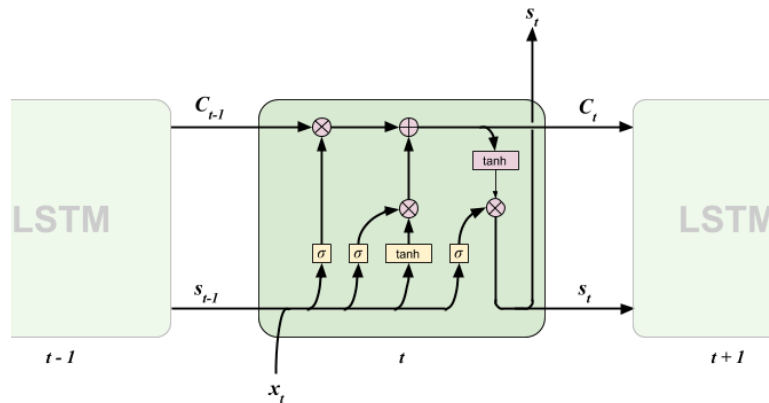


Gambar 2 Memori pada RNN

$$st = \tanh(U \cdot xt + W \cdot st - 1) \dots\dots\dots(2.1)$$

$$\hat{y}t = \text{softmax}(V \cdot st) \dots\dots\dots(2.2)$$

Berbeda pada pemrosesan modul *LSTM* memiliki komputasi yang banyak.



Gambar 3 Memori pada LSTM

Dari modul *Recurrent neural network (RNN)* dan *Long Short Term Memory (LSTM)* memiliki tiga tahapan, yaitu Input nilai data pelatihan sehingga diperoleh nilai output (tahap *Feedforward*), Propagasi balik dari nilai *error* yang diperoleh (tahap *Backpropagation*), didalam kedua modul arsitekturnya terdiri dari tiga layer, yaitu:

- **Input Layer**

Berisi node-node yang mempunyai sebuah nilai masukan yang tidak berubah pada fase latih dan hanya bisa berubah jika diberikan nilai masukan baru. Node pada layer ini tergantung pada banyaknya input dari suatu pola.

- **Hidden Layer**

Layer ini tidak pernah muncul sehingga dinamakan *hidden layer*. Namun semua proses pada fase pelatihan dan fase pengenalan dijalankan di lapisan ini. Jumlah lapisan ini tergantung dari arsitektur yang akan dirancang, tetapi pada umumnya terdiri dari satu lapisan hidden layer yang saling berurutan. Didalam hidden layer terdapat perbedaan antara modul *Recurrent Neural Network (RNN)* dan *Long Short Term Memory (LSTM)* perbedaannya yaitu :

1. *Hidden layer Recurrent Neural Network (RNN) :*

*Hidden layer RNN* terdiri dari sel memori, satu sel memori memiliki satu gerbang dan output dengan fungsi *softmax*, satu gerbang tersebut digunakan untuk menyimpan data dari sel memori sebelumnya untuk diproses dengan fungsi *tanh* bersamaan dengan data masukan sekarang. Arsitektur RNN dapat dilihat diatas pada Gambar 2 dan rumus pada persamaan (2.1) dan (2.2).

2. *Hidden Long Short Term Memory (LSTM)* terdiri dari sel memori, satu sel memori memiliki tiga *gate* yaitu :

### 1. *Forget gate*

*Forget gate* adalah lapisan melupakan yang mengambil *output* pada waktu ( $h^{t-1}$ ) dan *input* pada waktu ( $x^t$ ) parameter tersebut digabungkan dan di proses dengan fungsi sigmoid. Dari output ini menghasilkan antara dua bilangan 0 atau 1 dimana  $f_t = 0$  maka keadaan (*state*) sebelumnya akan dilupakan, sementara jika  $f_t = 1$  *state* sebelumnya tidak berubah.

$$z_t^f = W^f \cdot \begin{bmatrix} x^t \\ h^{t-1} \end{bmatrix} \dots\dots\dots (2.3)$$

$$f_t = \sigma(z_t^f + b_f) \dots\dots\dots (2.4)$$

### 2. *Input gate*

*Input gate* melakukan pembuatan vector baru dan disimpan di *memory cell* terdapat dua proses yaitu :

1. Proses input asli dari perhitungan *output* sebelumnya waktu ( $h^{t-1}$ ) dan *input* baru waktu ( $x^t$ ), diproses dengan fungsi sigmoid. Di *paper* disebut (lapisan sigmoid input) [7].

$$z_t^i = W^i \cdot \begin{bmatrix} x^t \\ h^{t-1} \end{bmatrix} \dots\dots\dots (2.5)$$

$$i_t = \sigma(z_t^i + b_i) \dots\dots\dots (2.6)$$

2. Proses memperhitungkan *output* sebelumnya dan *input* baru ( $x^t$ ) diproses dengan fungsi yang berbeda yaitu tanh. Di *paper* disebut (lapisan kandidat) [7].

$$z_t^{\check{}} = W^{\check{}} \cdot \begin{bmatrix} x^t \\ h^{t-1} \end{bmatrix} \dots\dots\dots (2.7)$$

$$\check{c}_t = \tanh(z_t^{\check{}} + b_{\check{c}_t}) \dots\dots\dots (2.8)$$

Dari proses perkalian dari kedua hasil tersebut menghasilkan vektor baru dan disimpan di *memory cell* sebagai calon masukan ( $i_t \cdot \check{c}_t$ ), calon masukan tersebut akan dilanjutkan ke lapisan pernghubung ( $C_t$ ) antara hasil  $f_t$ , calon masukan ( $i_t \cdot \check{c}_t$ ), dan *state* penghubung sebelumnya ( $C^{t-1}$ ) diproses dengan rumus [7] :

$$c_t = f_t * c_{t-1} + i_t * \check{c}_t \dots\dots\dots (2.9)$$

### 3. *Output gate*

*Output gate* mengontrol seberapa banyak *state* yang lewat ke output. gerbang ini melakukan dua proses yaitu [7] :

1. Perhitungan *output* sebelumnya waktu ( $h^{t-1}$ ) dan *input* baru waktu ( $x^t$ ), diproses dengan fungsi sigmoid.

$$z_t^o = W^o \cdot \begin{bmatrix} x^t \\ h^{t-1} \end{bmatrix} \dots\dots\dots (2.10)$$

$$o_t = \sigma(z_t^o + b_i) \dots\dots\dots (2.11)$$

2. Proses mengontrolnya dua *state* dari *state* penghubung  $c_t$ , dan *state* output  $o_t$  diproses menghasilkan *block output* ( $h^t$ ) dengan fungsi  $\tanh c_t$  [7].

$$h_t = o_t \cdot \tanh(c_t) \dots\dots\dots (2.12)$$

- ***Output Layer***

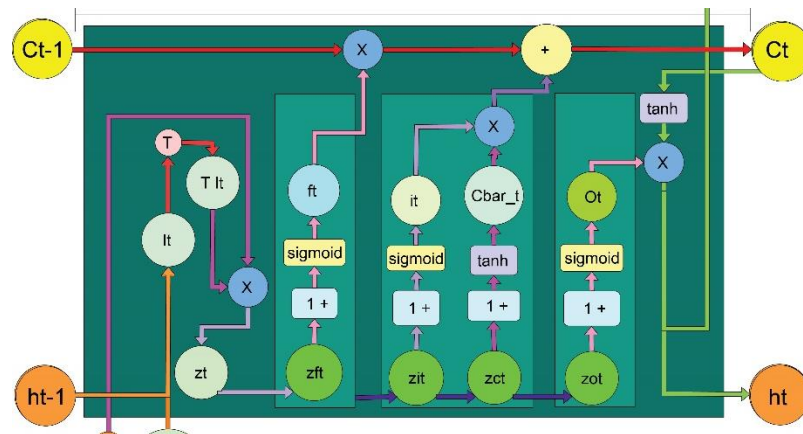
Output layer berfungsi untuk menampilkan hasil perhitungan sistem dengan fungsi gerbang – gerbang pada *hidden layer* berdasarkan input yang diterima.

Secara umum dari modul *Recurrent neural network (RNN)* dan modul *Long Short Term Memory (LSTM)* memiliki arsitektur yang sama yaitu memiliki tiga tahapan yaitu *input*, kemudian *hidden layer* yang berproses berulang didalamnya dan akhirnya akan dilanjutkan pada tahap *output layer*. Setiap literasi dalam *Recurrent Neural Network (RNN)* maupun *Long Short Term Memory (LSTM)* dilakukan dengan dua arah yaitu *Forward* (tahap maju) dan *Backward* (tahap mundur).

## 2.3 *Long Short Term Memory (LSTM)*

*Long Short Term Memory Networks (LSTM)* merupakan salah satu metode dari *Recurrent Neural Network (RNN)*. LSTM diajukan oleh Sepp Hochreiter dan Jurgen Schmidhuber pada tahun 1997. LSTM di *design* untuk menghindari masalah *long term dependency* yang ada pada RNN (Grave, 2014) [4]. Didalam *hidden layer* LSTM yang saling berhubungan dan saling berkaitan, terdapat proses dalam menyeleksi suatu informasi untuk diingat dalam jangka waktu pendek maupun

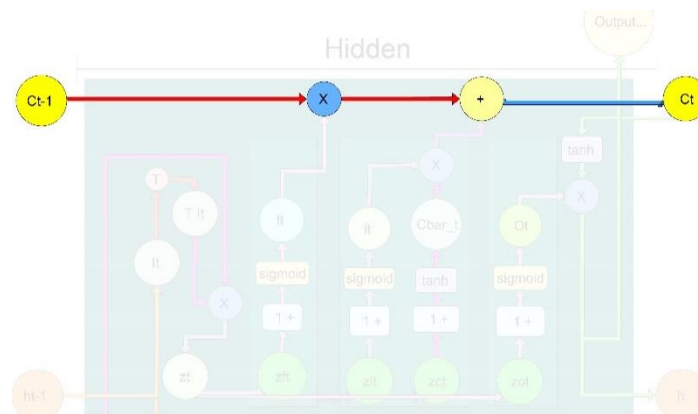
dalam jangka waktu panjang, setiap proses dalam satu urutan akan dijelaskan sebagai berikut :



Gambar 4 Arsitektur *Hidden Layer*

## 1. Status sel

Status sel ini berjalan lurus dan menghubungkan seluruh urutan. (*Long memory*), Sel ini akan menghubungkan dari semua urutan, diawali dari urutan yang pertama dimulai hingga urutan yang berlangsung sekarang atau berproses sekarang.



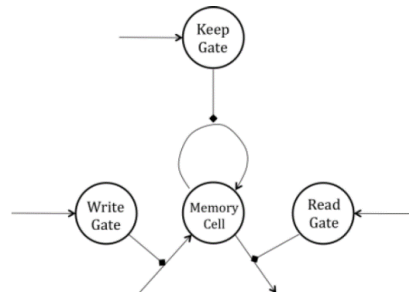
Gambar 5 Status sel LSTM

## 2. Gerbang

Gerbang ini bertujuan mendapatkan informasi secara opsional. Dari satu urutan terdapat 3 gerbang yaitu *Forget gate layer (keep gate)* bertujuan untuk menjaga informasi sebelumnya agar tidak sembarangan masuk kedalam keadaan urutan sekarang, *Forget gate layer (write gate)* bertujuan untuk menuliskan informasi apa yang didapat dari urutan sebelumnya dan disimpan di *memory cell* dan *read gate*



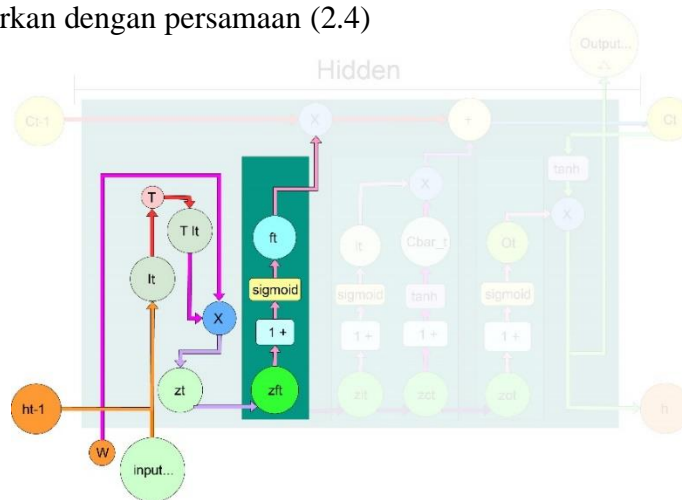
bertujuan untuk menerima informasi yang sudah terseleksi sebelumnya dengan melewati gerbang – gerbang sebelumnya.



Gambar 6 Gerbang *LSTM*

### 3. Penjagaan gerbang (*Keep gate*)

Penjagaan informasi yang akan dibuang dari urutan sebelumnya. Dari gerbang ini diberikan *output* antara dua bilangan 0 atau 1 dimana  $f_t = 0$  maka keadaan sebelumnya akan dilupakan, sementara jika  $f_t = 1$  keadaan sebelumnya tidak berubah. Dijabarkan dengan persamaan (2.4)



Gambar 7 Penjagaan gerbang (*Keep gate*)  $f_t$  *LSTM*

### 4. Menambahkan informasi baru (*write gate*)

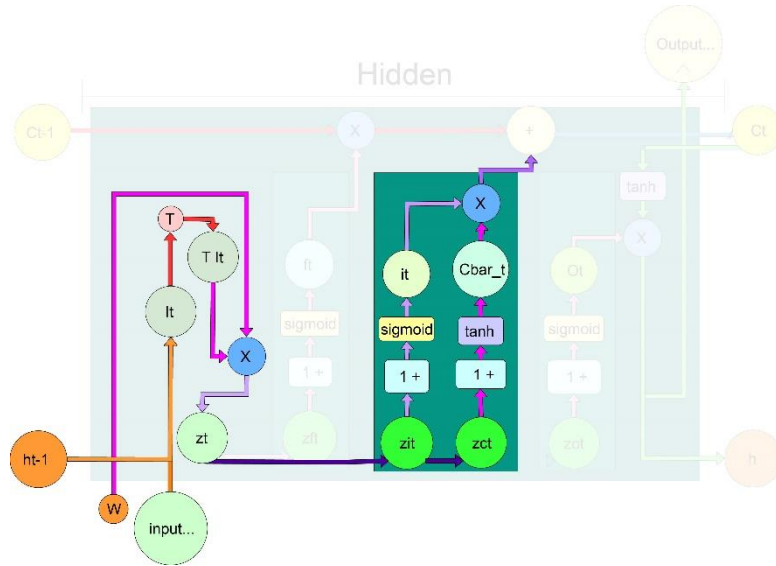
Menambahkan informasi baru Penentuan apa yang disimpan dalam keadaan sel, di dalam gerbang ini terdapat 2 lapisan yaitu :

- lapisan gerbang input.

Menambahkan nilai baru dari input. Dijabarkan dengan persamaan (2.6)

- Lapisan tanh

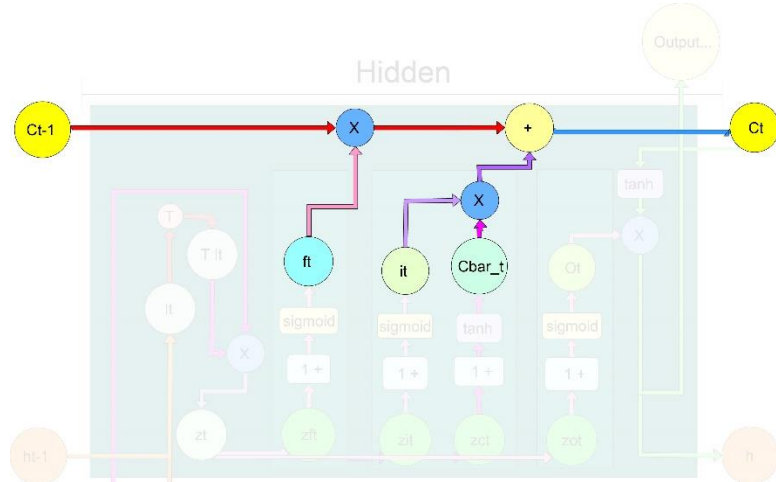
Membuat vektor nilai kandidat baru, Dijabarkan dengan persamaan (2.8)



Gambar 8 Menambahkan informasi baru (write gate) LSTM

##### 5. Update cell state ( $C_t$ )

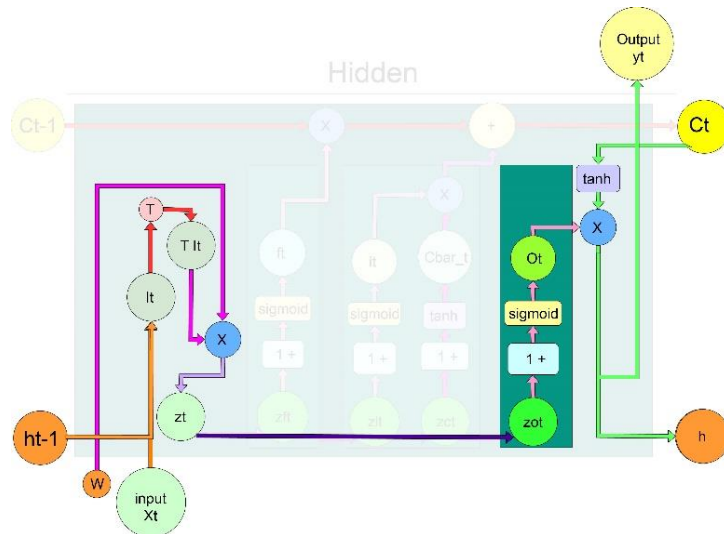
Proses melupakan hal-hal yang sudah diseleksi untuk dilupakan sebelumnya  $f_t * \check{C}_{t-1}$  dan proses menambahkan informasi yang sudah ditentukan sebelumnya  $i_t * \check{C}_t$  Secara rumus dapat dituliskan seperti persamaan (2.9)



Gambar 9 Update cell state LSTM

## 6. Membaca keputusan (*Output*)

Proses ini akan menentukan bagaimana sel mana yang akan dihasilkan dengan bantuan lapisan gerbang *output* dan lapisan *tanh* akan memberikan nilai antara -1 dan +1



Gambar 10 Membaca keputusan (*Output*) LSTM

Lapisan gerbang output dijabarkan dengan persamaan (2.11)

Lapisan *tanh* dijabarkan dengan persamaan (2.12) (*Short memory*)

### 2.3.1 Algoritma pelatihan *Long Short Term Memory Networks*

Algoritma pelatihan *Long Short Term Memory Networks* pada dasarnya terdiri dari beberapa tahapan yaitu :

- Input* nilai data *training* panjang urutan (*Sequence length*)
- Forward LSTM* (tahap maju) sampai diperoleh nilai *output*
- Backpropagation through time LSTM* (tahap mundur) sampai diperoleh nilai *error* yang dipeoleh.
- Penyesuaian bobot (*gradient descent*) sampai mendapatkan bobot minimum

Dari tahapan diatas diulangi terus- menerus sampai mendapatkan nilai *error* yang diinginkan.

Berikut langkah-langkah pelatihannya adalah :

#### Langkah 0 :

- Persiapkan data normalisasi
- Inisialisasi dimensi *input* dan *output* ( $x_t, y_t$ )

- c. Tetapkan *Epcchs* (satu presentasi lengkap dari proses *forward* sampai *Backward* untuk pemrosesan data yang akan dipelajari ke mesin pembelajaran)
- d. Inisialisasi bobot (bobot awal dengan nilai random yang cukup kecil)
- e. Tetapkan maksimum target *error* dan *learning rate*
- f. Tetapkan panjang urutan (*sequence length*)

**Langkah 1 :**

- a. Penggabungan antara input data dan block output dari langkah sebelumnya:

$$I^t = \begin{bmatrix} x^t \\ h^{t-1} \end{bmatrix} \dots\dots\dots(2.13)$$

$I_t$  : Hasil penggabungan *input*

$h^{t-1}$  : Hasil block output sebelumnya

$x_t$  : Input data yang akan diolah

- b. Block input dengan mempersiapkan input  $f_t, i_t, \check{c}_t, o_t$  :

$$W_t = \begin{bmatrix} w^f & U^f \\ w^i & U^i \\ w^c & U^c \\ w^o & U^o \end{bmatrix} \dots\dots\dots (2.14)$$

Catatan :

- Baris  $W_t$  jika simpul 1 maka  $4 \times 1 = 4$
- Kolom  $W_t$  jika *size* input 2 maka  $2 + \text{simpul} = 3$
- Jadi ordo matrik  $W_t$  adalah (4,3)

$$z_t = \left( W_t \cdot \begin{bmatrix} x^t \\ h^{t-1} \end{bmatrix} \right) \dots\dots\dots (2.15)$$

$$z_t = \begin{bmatrix} w^f & U^f \\ w^i & U^i \\ w^c & U^c \\ w^o & U^o \end{bmatrix} \cdot \begin{bmatrix} b \\ x^t \\ h^{t-1} \end{bmatrix} \dots\dots\dots (2.16)$$

$$z_t = \begin{bmatrix} z_t^f \\ z_t^i \\ z_t^{\check{c}} \\ z_t^o \end{bmatrix} \dots\dots\dots(2.17)$$

Keterangan :

$Z_t$  : Hasil *input (hidden layer)* terdapat 4 baris untuk *input*

- $z_t^f$  : *input (hidden layer)* pada gerbang *forget* seperti persamaan (2.3)  
 $z_t^i$  : *input (hidden layer)* pada gerbang *input* seperti persamaan (2.5)  
 $z_t^{\check{c}}$  : *input (hidden layer)* pada *candidate state* seperti persamaan (2.7)  
 $z_t^o$  : *input (hidden layer)* pada gerbang *output* seperti persamaan (2.10)  
 $W_t$  : Bobot (weight) yang menghubungkan input layer  
 $U$  : Data informasi

### 2.3.2 Feedforward

#### Langkah 2 :

- a. Gerbang lupa (*forget gate*):

$$f_t = \sigma(z_t^f + b_f) \dots\dots\dots(2.3)$$

Keterangan :

- $f_t$  : Hasil perhitungan gerbang lupa (*forget gate*)  
 $z_t^f$  : *Block Input forget gate*  
 $Z_i$  : Nilai *input* dari *hidden output*  
 $b_f$  : Bias  
 $\sigma$  : Logistic sigmoid ( $f(x) = \frac{1}{1+e^x}$ )

- b. Gerbang pembuatan vektor baru dan disimpan di *memory cell*:

- $i_t = \sigma(z_t^i + b_i) \dots\dots\dots(2.6)$
- $\check{c}_t = \tanh(z_t^{\check{c}} + b_{\check{c}}) \dots\dots\dots (2.8)$

Keterangan :

- $i_t$  : Hasil perhitungan gerbang *input (input gate)*  
 $\check{c}_t$  : Proses *tanh (candidate gate)*  
 $\tanh$  : Fungsi *tanh* digunakan membuat vektor baru

- c. Keadaan penghubung (*state cell*) :

$$c_t = f_t * c_{t-1} + i_t * \check{c}_t \dots\dots\dots (2.9)$$

Keterangan :

- $c_t$  : Hasil perhitungan keadaan penghubung (*connecting state*)

$c_{t-1}$  : Keadaan penghubung sebelumnya

d. Gerbang *output* (*output gate*) :

$$o_t = \sigma(z_t^o + b_o) \dots\dots\dots(2.11)$$

Keterangan :

$o_t$  : Hasil perhitungan gerbang *output* (*output gate*)

e. *Block output*

$$h_t = o_t \cdot \tanh(c_t) \dots\dots\dots(2.12)$$

Keterangan :

$h_t$  : Hasil perhitungan *output* dan model untuk proses prediksi

### 2.3.3 Backpropagation Through Time (BPTT)

**Langkah 3 :**

$$E = (y_t - \hat{y}_t)$$

Keterangan :

$y_t$  : Harga real (data asli)

$\hat{y}_t$  : Harga prediksi (data prediksi)

Pada proses forward diberikan  $h^t = o^t \odot \tanh(c^t)$  dan ditemukan  $\delta o^t, \delta c^t$

a. Hitung informasi *derivative* gerbang output  $\delta o^t$ :

$$\delta o^t = \delta h^t \odot \tanh(c^t) \dots\dots\dots(2.18)$$

Keterangan :

$\delta o^t$  : Menghitung informasi *derivative* gerbang output

$\delta h^t$  : Informasi *derivative* output block

$c^t$  : keadaan penghubung (*connecting state*)

b. Hitung informasi *derivative* keadaan penghubung (*cell state*)  $\delta c^t$ :

$$\delta c^t = \delta h^t \odot o^t \odot (1 - \tanh^2(c^t)) \dots\dots\dots(2.19)$$

Keterangan :

$\delta c^t$  : Menghitung informasi *derivative* keadaan penghubung

$o^t$  : Gerbang *Output*

#### Langkah 4 :

Pada proses forward diberikan  $c^t = i^t \odot a^t + f^t \odot c^{t-1}$  dan ditemukan  $\delta i^t, \delta a^t, \delta f^t, \delta c^{t-1}$

- a. Hitung informasi derivative gerbang input (*input gate*)  $\delta i^t$  :

$$\delta i^t = \delta c^t \odot a^t \dots\dots\dots(2.20)$$

Keterangan :

$\delta i^t$  : Menghitung informasi *derivative input gate*

$\delta c^t$  : Informasi *derivative* keadaan penghubung

$a^t$  : *candidate gate* ( $\check{C}_t$ )

- b. Hitung informasi *derivative* gerbang *candidate connecting state*  $a^t$

$$\delta a^t = \delta c^t \odot i^t \dots\dots\dots(2.21)$$

Keterangan :

$\delta a^t$  : Informasi *derivative candidate connecting state*

$\delta c^t$  : Informasi *derivative* Keadaan penghubung

$i^t$  : *input gate*

- c. Hitung informasi *derivative* gerbang *forget gate*  $\delta f^t$

$$\delta f^t = \delta c^t \odot c^{t-1} \dots\dots\dots(2.22)$$

Keterangan :

$\delta f^t$  : Informasi *derivative forget gate*

$\delta c^t$  : Informasi *derivative* keadaan penghubung

$c^{t-1}$  : Keadaan penghubung sebelumnya

- d. Hitung informasi derivative Keadaan penghubung sebelumnya

$$\delta c^{t-1} = \delta c^t \odot f^t \dots\dots\dots,,,(2.23)$$

Keterangan :

$\delta c^{t-1}$  : Informasi *derivative* keadaan penghubung sebelumnya

$\delta c^t$  : Informasi *derivative* keadaan penghubung

$f^t$  : Gerbang *forget gate*

**Langkah 5 :**

Pada proses forward diberikan Forward Pass:  $z^t = \begin{bmatrix} \hat{a}^t \\ \hat{i}^t \\ \hat{f}^t \\ \hat{o}^t \end{bmatrix} = W \times I^t$  maka pada fase

backward didapat  $\delta z^t = \delta a^t, \delta i^t, \delta f^t, \delta o^t$ .

- a. Hitung *derivative block input (candidate gate)* :

$$\begin{aligned} a^t &= \check{C}_t \\ \delta z \hat{a}^t &= \delta a^t \odot (1 - \tanh^2(\hat{a}^t)) \dots\dots\dots(2.24) \end{aligned}$$

Keterangan :

$\delta z \hat{a}^t$  : Informasi *derivative input block candidate gate*  
 $\delta a^t$  : Informasi *derivative candidate gate*  
 $\hat{a}^t$  : *block input candidate gate*

- b. Hitung *derivative block input (input gate)*:

$$\delta z \hat{i}^t = \delta i^t \odot I^t \odot (1 - i^t) \dots\dots\dots(2.25)$$

Keterangan :

$\delta z \hat{i}^t$  : Informasi *derivative input block input gate*  
 $\delta i^t$  : Informasi *derivative input gate*  
 $i^t$  : *input gate*

- c. Hitung *derivative block input (forget gate)* :

$$\delta z \hat{f}^t = \delta f^t \odot f^t \odot (1 - f^t) \dots\dots\dots (2.26)$$

Keterangan :

$\delta z \hat{f}^t$  : Informasi *derivative input block forget gate*  
 $\delta f^t$  : Informasi *derivative forget gate*  
 $f^t$  : *forget gate*

- d. Hitung *derivative block input (output gate)* :

$$\delta z \hat{o}^t = \delta o^t \odot o^t \odot (1 - o^t) \dots\dots\dots (2.27)$$

$\delta z \hat{o}^t$  : Informasi *derivative input block output gate*  
 $\delta o^t$  : Informasi *derivative output gate*  
 $o^t$  : *output gate*



- e. Penggabungan dari  $\delta z \hat{a}^t, \delta z \hat{i}^t, \delta z \hat{f}^t, \delta z \hat{o}^t$  menjadi satu untuk menghasilkan  $\delta z^t$  dari *forward* terdapat Forward Pass:  $z^t = W \times I^t$  maka perlu pengembalian  $z^t$  ke  $\delta z^t$  :

Dari forward didapat persamaan (2.17) maka fase backward didapat persamaan

$$\delta z^{t^T} = [\delta z \hat{a}^t, \delta z \hat{i}^t, \delta z \hat{f}^t, \delta z \hat{o}^t]^T \dots\dots\dots (2.28)$$

$$\delta z^{t^T} = \begin{bmatrix} \delta z \hat{a}^t \\ \delta z \hat{i}^t \\ \delta z \hat{f}^t \\ \delta z \hat{o}^t \end{bmatrix} \dots\dots\dots (2.29)$$

### 2.3.4 Mencari Bobot dari proses backpropagation urutan (sequence)

#### Langkah 6 :

Dari tahap *forward pass* didapat persamaan (2.15) dan diberikan  $\delta z^t$  dari langkah sebelumnya maka dicari  $\delta W^t$  dan  $\delta h^{t-1}$

- a. Persamaan (2.29) didapat untuk menghitung  $\delta I^t$

$$\delta I^t = \delta z^t \cdot W^{t^T} \dots\dots\dots (2.30)$$

$$\delta I^t = \begin{bmatrix} \delta z \hat{a}^t \\ \delta z \hat{i}^t \\ \delta z \hat{f}^t \\ \delta z \hat{o}^t \end{bmatrix} \cdot [W^{\check{c}} \ W^i \ W^f \ W^o] \dots\dots\dots (2.31)$$

- b. Dari persamaan (2.31), didapat untuk menghitung *derivative block output* ( $\delta h^{t-1}$ )

$$\delta h^{t-1} = [\delta W^o \cdot \delta z \hat{o}^t] \dots\dots\dots (2.32)$$

- c. Mencari  $\delta W^t$  *derivative* pada bobot yang akan menjadi model dalam prediksi selanjutnya

$$\delta W^t = \delta z^t \cdot I^{t^T} \dots\dots\dots (2.33)$$

$$\delta W^t = \begin{bmatrix} W^{\check{c}} \cdot \delta z \hat{a}^t \\ W^i \cdot \delta z \hat{i}^t \\ W^f \cdot \delta z \hat{f}^t \\ W^o \cdot \delta z \hat{o}^t \end{bmatrix} \cdot [W^{\check{c}} \cdot \delta z \hat{a}^t, W^i \cdot \delta z \hat{i}^t, W^f \cdot \delta z \hat{f}^t, W^o \cdot \delta z \hat{o}^t] \dots\dots\dots (2.34)$$

d. Menghitung Derivatif Bobot

Menggunakan aturan rantai konvensional untuk menghitung turunan dari E sehubungan dengan semua bobot. Dalam kondisi tertentu, ini bisa menjadi bobot pendekatan.[8]

$$\delta W^t = \begin{bmatrix} W^c \cdot \delta z \hat{a}^t \\ W^i \cdot \delta z \hat{i}^t \\ W^f \cdot \delta z \hat{f}^t \\ W^o \cdot \delta z \hat{o}^t \end{bmatrix} \cdot [W^c \cdot \delta z \hat{a}^t, W^i \cdot \delta z \hat{i}^t, W^f \cdot \delta z \hat{f}^t, W^o \cdot \delta z \hat{o}^t] \quad (2.34)$$

$$\frac{\partial E^t}{\partial W} = \delta W^t \dots\dots\dots (2.35)$$

$$\delta W^t(\text{baru}) = \delta W^t(\text{lama}) + \delta W^{t+1} \dots\dots\dots (2.36)$$

Keterangan :

$\delta W^t$  = *derivative* bobot

$\delta W^t(\text{baru})$  dihitung sampai t=1

### 2.3.5 Optimasi untuk update bobot Gradient Descent

**Langkah 7 :** *Gradient descent weight optimization*

Menghitung *Gradient descent* bobot optimasi dari parameter, bobot dan bias, dalam jaringan saraf diperbarui menggunakan data pelatihan sehingga cost average dari semua contoh pelatihan diminimalkan [9].

Dapat dituliskan dengan persamaan :

$$W(\text{final}) = W(\text{Lama}) - \text{learning\_rate} * \delta W^1 \dots\dots\dots (2.37)$$

Keterangan :

*learning\_rate* : Parameter untuk mengontrol perubahan bobot selama pelatihan. Semakin besar learning rate, maka jaringan syaraf tiruan akan semakin cepat belajar tetapi hasilnya kurang akurat. Semakin kecil learning rate, maka jaringan syaraf tiruan akan semakin lambat belajar tetapi hasilnya lebih akurat [10]

$W(\text{final})$  = bobot ini akan digunakan pemodelan.

### 3.3.6 Menghitung *error* dari perbandingan data asli dengan data prediksi

**Langkah 8 :** Hitung total *error*

Menghitung total *error* menggunakan *mean squared error* dengan rumus perhitungan:

$$MSE = \frac{1}{2} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \dots\dots\dots (2.38)$$

Keterangan :

*MSE* : Merupakan total *error*

$y_i$  : Merupakan target *output* ke-i

$\hat{y}_i$  : Merupakan *output* dari pelatihan ke-i

Fase tersebut diulang hingga kondisi *error* terpenuhi.

## 2.4 Penelitian Terkait

Table 1. Tabel Penelitian Terkait

No	Judul Penelitian	Teknik yang dilakukan penelitian	Hasil Penelitian	Objek Penelitian	Relevansi dengan penelitian yang akan dilakuakn	Perbedaan dengan penelitian yang akan dilakukan
1	<b>Comparison of Predictive Algorithms: Backpropagation, SVM, LSTM and Kalman Filter for Stock Market [3]</b>	<ol style="list-style-type: none"> <li>Melakukan pembuatan momentum pada harga saham, penggabungan harga saham yang masuk disektor technology, dan juga index NASDAQ.</li> <li>Algoritma <i>Backpropagation</i>, LSTM, dan SVM</li> </ol>	Perbandingan keakurasian dari beberapa algoritma dalam memprediksi harga saham	Saham Amerika, seperti : <ul style="list-style-type: none"> <li>- Rata – rata Semua saham yang ada di sector technology</li> <li>- Harga saham dari perusahaan itu sendiri</li> <li>- Index NASDAQ</li> </ul>	Relevansi penelitan Tugas Akhir dengan paper penulis yaitu sama-sama melakukan prediksi menggunakan metode LSTM	<b>Paper :</b> <ol style="list-style-type: none"> <li>Saham yang diteliti yaitu saham Amerika.</li> <li>Perbandingan 3 metode.</li> </ol> <b>Tugas Akhir :</b> <ol style="list-style-type: none"> <li>Saham yang diteliti yaitu saham Indonesia.</li> <li>Hanya menerapkan metode LSTM</li> </ol>

No	Judul Penelitian	Teknik yang dilakukan penelitian	Hasil Penelitian	Objek Penelitian	Relevansi dengan penelitian yang akan dilakuakn	Perbedaan dengan penelitian yang akan dilakukan
2	<b>Stock Price Prediction Using LSTM on Indian Share Market</b> [11]	Melakukan prediksi performansi atau trend dari pasar saham dengan LSTM	Hasil dari studi mengkonfirmasi bahwa <i>LSTM</i> sanggup untuk memprediksi performansi dari saham	Pasar saham India	Persamaan penelitian Tugas Akhir dengan paper yaitu sama-sama melakukan prediksi saham dengan <i>LSTM</i>	<b>Paper :</b> 1. Objek yang di teliti yaitu pasar saham India <b>Tugas Akhir :</b> 1. Objek saham harga saham Indonesia PT Semen Indonesia (persero) Tbk
3.	<b>Forecasting saham syariah dengan Menggunakan lstm</b> [12]	Melakukan prediksi pada pasar saham dengan LSTM	Menunjukkan grafik penutupan (Close) saham JII periode 2013-2019	Harga penutupan (Close) saham JII	Persamaan penelitian Tugas Akhir dengan paper yaitu sama-sama ujicoba epoch yang berbeda	<b>Paper :</b> 1. Objeck yang diteliti Index saham JII <b>Tugas Akhir :</b> 1. Objeck yang diteliti saham PT Semen Indonesia (persero) Tbk

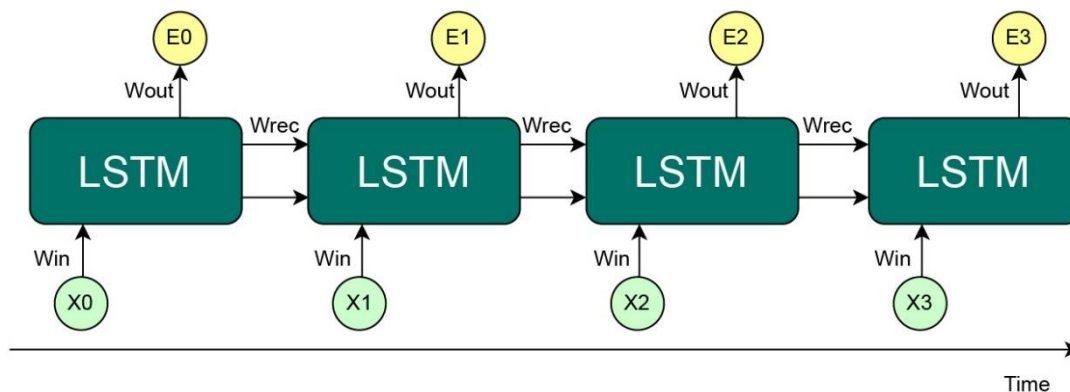
Berdasarkan penelitian diatas yang berjudul *Comparison of Predictive Algorithms: Backpropagation, SVM, LSTM and Kalman Filter for Stock Market* menyimpulkan bahwa prediksi saham menggunakan LSTM lebih baik dari backpropagation, dan SVM, penelitian yang berjudul *Stock Price Prediction Using LSTM on Indian Share Market* dan *Forecasting saham syariah dengan menggunakan LSTM* sama-sama menyimpulkan juga bahwa prediksi saham India dan saham Indonesia dengan menggunakan *LSTM* menghasilkan hasil yang baik.

## BAB III METODE USULAN

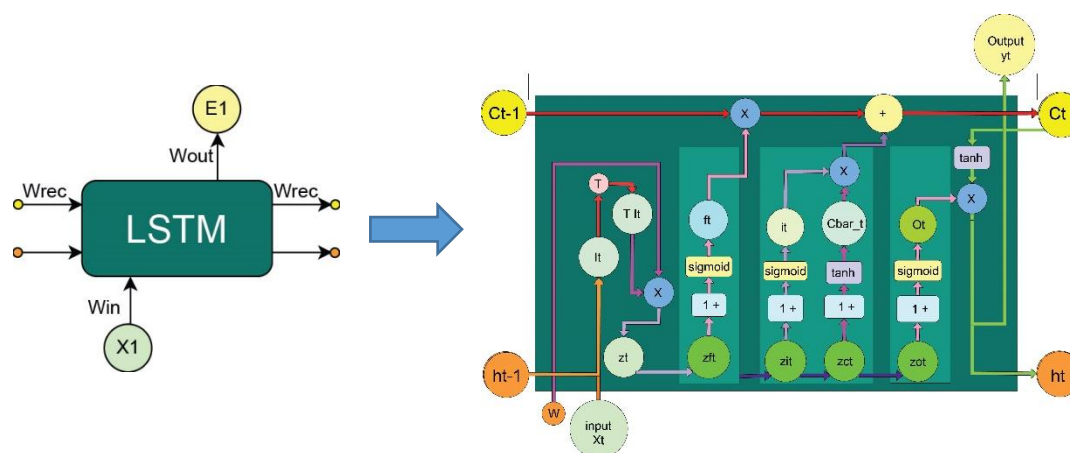
### 3.1 Metode

#### 3.1.1 Rancangan Arsitektur LSTM

Arsitektur ini adalah Rancangan LSTM yang mendeskripsikan alur algoritma LSTM dari proses *forward* sampai *backward*, Alur algoritmanya adalah sebagai berikut :



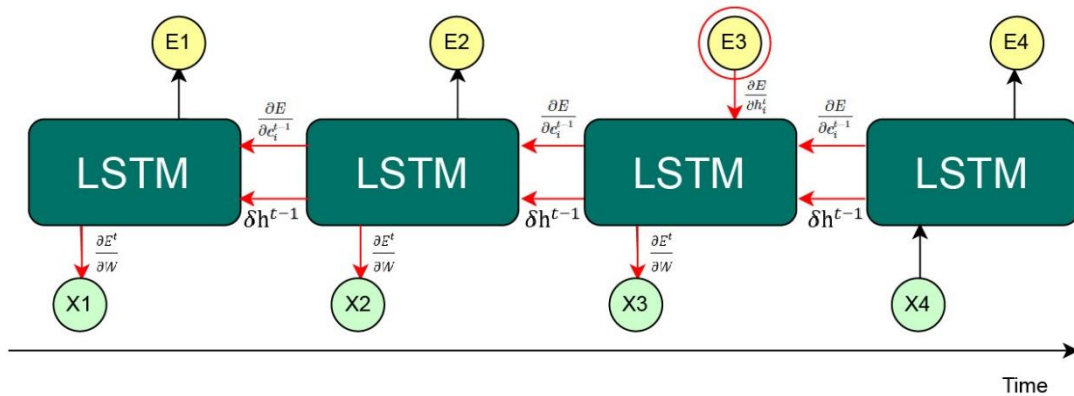
Gambar 11 Arsitektur *LSTM forward*



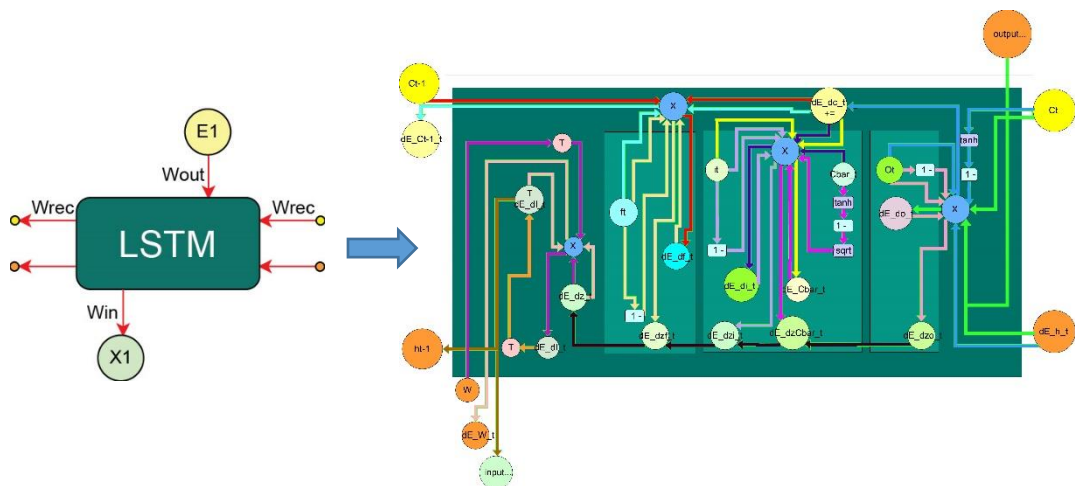
Gambar 12 Penjabaran dari *Hidden Layer LSTM*

Pada gambar 5 setiap urutan  $x_0, x_1, \dots, x_t$  terdapat hidden layer yang berhubungan satu sama lain dan terdapat gerbang – gerbang yang bertujuan untuk menyeleksi informasi yang tidak perlu diingat dalam urutan selanjutnya seperti ditunjukkan pada gambar 12.

Dari tahap forward yang ditunjukkan pada gambar 11 didapatlah bobot baru ( $W^0$ ), bobot tersebut memiliki hasil *error* yang besar, jadi dibutuhkanlah proses *Backward* untuk mendapatkan *error* bobot terbaik.



Gambar 13 Arsitektur *LSTM Backward*



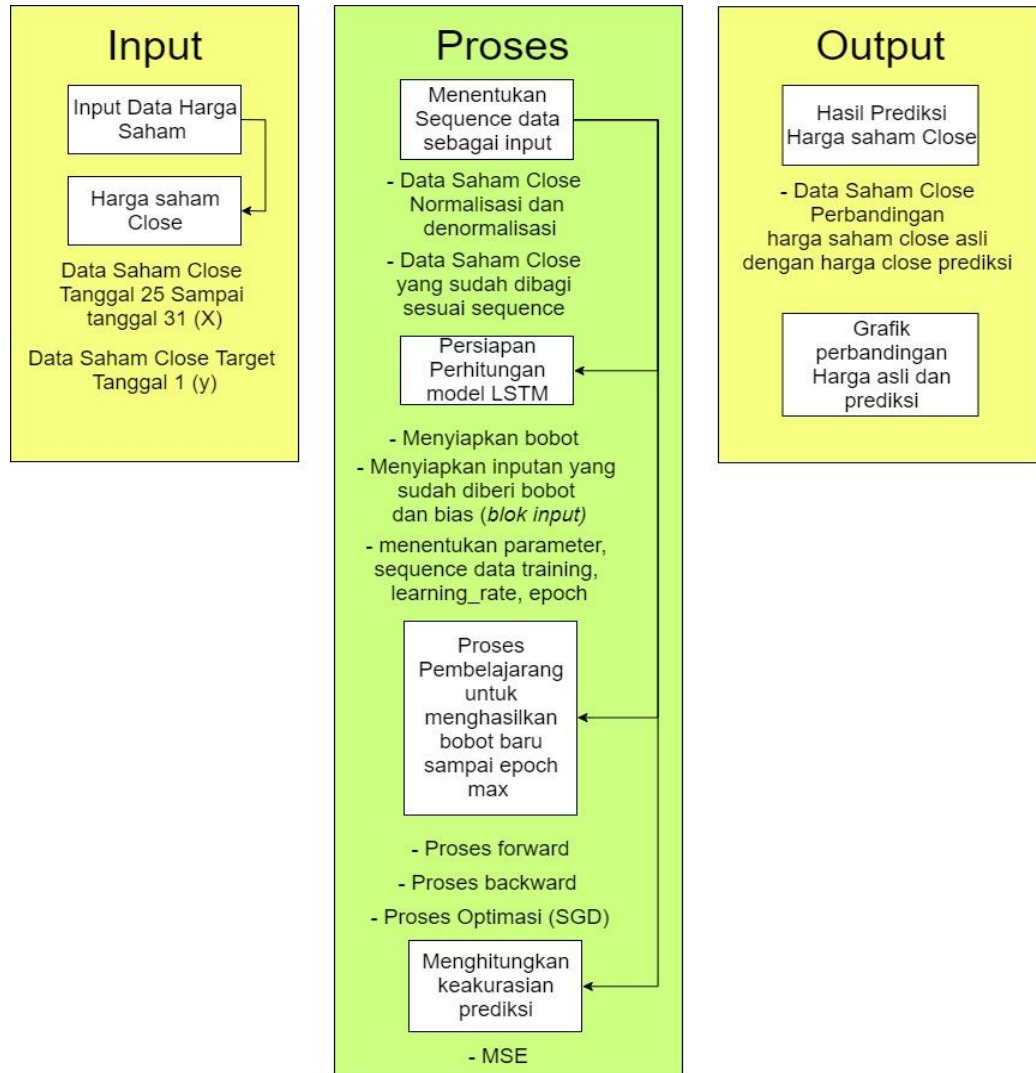
Gambar 14 Penjabaran dari *Hidden Layer LSTM Backward*

Pada gambar 13 ditunjukkan langkah pertama di mulai dari *output* terakhir berproses menuju ke urutan *hidden layer* sampai urutan pertama, didalam *hidden layer* memberikan 3 keluaran yaitu, *derivative block input* ( $\delta h^{t-1}$ ), *derivative* keadaan penghubung (*cell state*)  $\delta c^{t-1}$ , dan bobot *derivative* ( $\delta W^t$ ).

*Derivative block input* ( $\delta h^{t-1}$ ) dan *derivative* keadaan penghubung (*cell state*)  $\delta c^{t-1}$ , ini akan menghubungkan dari urutan ke urutan sebelumnya sampai ke urutan pertama untuk mendapatkan *derivative minimum*, Dan tiap urutan *hidden layer* menghasilkan bobot *derivative* ( $\delta W^t$ ) yang akan di proses lagi dengan perhitungan update bobot dengan metode *Gradient descent weight*.

### 3.1.2 Rancangan Arsitektur Sistem

Arsitektur ini adalah perancangan menggunakan diagram I-P-O. Adapun rancangan sistem yang mendeskripsikan alur sistem dari awal hingga akhir dapat dilihat pada Gambar 15 :



Gambar 15 Diagram IPO Arsitektur Sistem

Rancangan sistem menjelaskan proses berjalannya program. Dimulai dengan pengambilan data harga saham. Kemudian diolah menjadi dataset untuk diproses menjadi data *training*.

Tahap proses menginisialisasi data deret waktu (*sequence*), Tahap selanjutnya mempersiapkan perhitungan LSTM dengan menyiapkan bobot, menyiapkan masukan yang sudah diberi bobot dan bias (*block input*) didalam proses training ada dua pembelajaran yaitu *forward propagation* dan *backward propagation* dari proses *backpropagation* bertujuan untuk mendapatkan nilai *error* yang terbaik. Hasil



*training* berbentuk bobot baru yang akan digunakan untuk pemodelan, setelah mendapatkan model proses prediksi harga didapatkan, dilakukan evaluasi dengan metode evaluasi *mse* dengan membandingkan masing-masing hasil prediksi dengan data aktual. Evaluasi ini digunakan untuk mengukur seberapa akurat hasil prediksi dengan menggunakan LSTM. Output proses menampilkan hasil prediksi dan grafik perbandingan harga asli dan harga prediksi.

## **3.2 Analisa Kebutuhan**

### **3.2.1 Kebutuhan Hardware**

Hardware adalah komponen pada komputer yang terlihat dan dapat disentuh yang digunakan untuk melakukan pembuatan sistem dan penginputan data. Hardware yang digunakan dalam pembuatan sistem penerapan metode *LSTM* untuk penerapan metode *LSTM* dalam prediksi harga saham adalah sebagai berikut :

Laptop dengan spesifikasi sebagai berikut :

- Processor : Intel® Core™ i5-4200U CPU @ 1.60GHz
- Memori : 8 RAM

### **3.2.2 Kebutuhan Software**

Software adalah komponen yang penting dalam komputer. software memiliki tugas untuk menjalankan perintah. Software ini dibuat dengan Bahasa pemograman oleh programmer yang kemudian di hubungkan dengan hardware. Software yang digunakan dalam pembuatan sistem penerapan metode *LSTM* untuk penerapan metode *LSTM* dalam prediksi harga saham adalah sebagai berikut :

1. Operating sistem : Windows 10 Pro 64-bit
2. *Program Tool* : IDLE Python
3. *Text Editor* : Visual studio code

## **3.3 Analisa Input**

Inputan dalam penelitian ini menggunakan data yang terdapat di Alphavantage dan Yahoo finance yaitu data harga saham PT Semen Indonesia (Persero) Tbk, data yang digunakan adalah data harga saham, Jumlah data yang terhimpun yaitu sebanyak 1500 data.

### 3.4 Analisa Proses

Penelitian dalam menerapkan metode LSTM terdapat beberapa proses yang diataranya yaitu :

- a. Pembuatan data untuk LSTM:
  - Normalisasi data
  - data deret waktu yang dinormalisasi
- b. Pengembangan Model:
  - LSTM adalah jenis Jaringan Syaraf Berulang (RNN).
    - Inisialisasi panjang urutan (*sequence*) :
    - *Forward Propagation*
    - *Backpropagation* kesalahan
    - Pembelajaran jaringan
- c. Prediksi harga saham.

### 3.5 Analisa Output

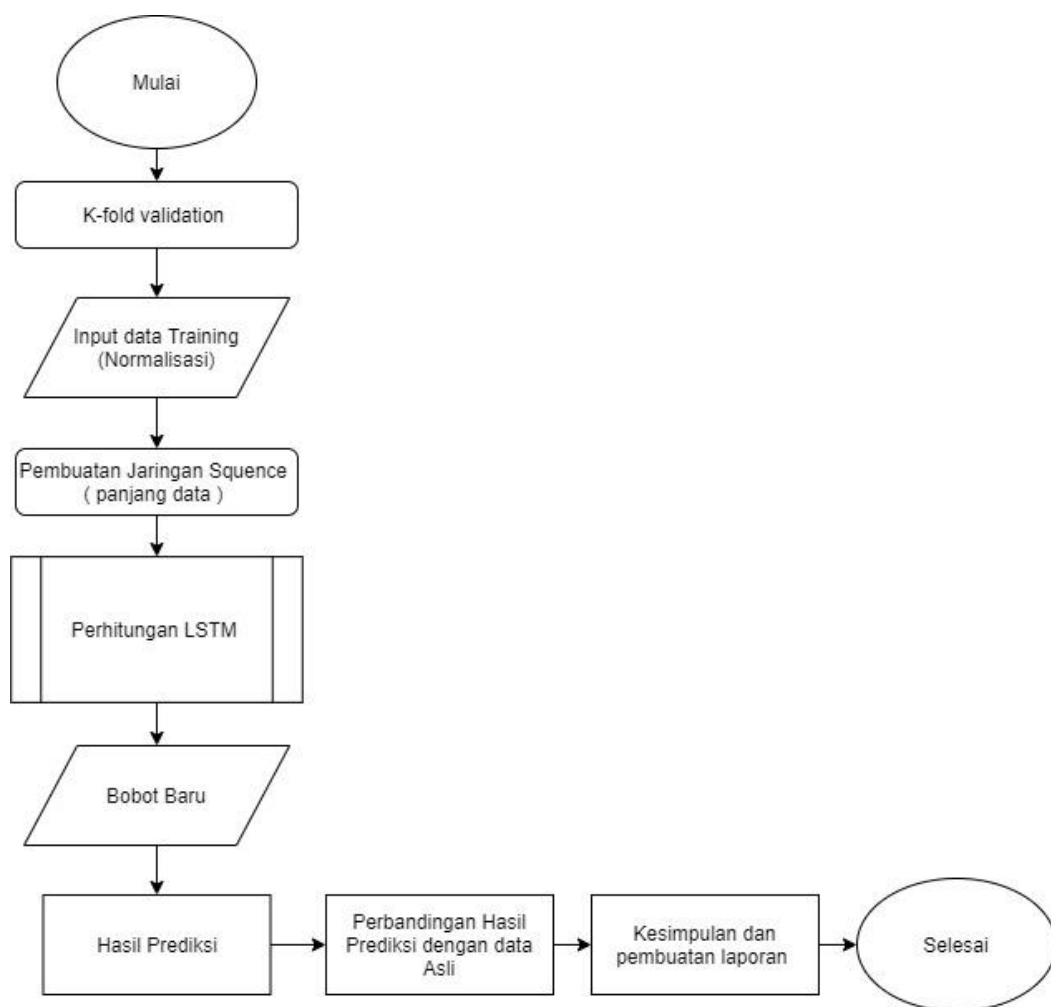
Hasil dari prediksi harga saham, saham akan ditampilkan dalam bentuk grafik perbandingan harga dari harga prediksi dengan harga aktual dan menampilkan juga grafik *error* dalam meprediksi harga saham.

### 3.6 Desain Rancangan

Perancangan sistem yang dibuat untuk membangun sistem prediksi dengan *flowchart* yang akan diimplementasikan. Tujuan agar dalam pembuatan sistem akan lebih terkonsep dan memiliki acuan sehingga akan memudahkan ketika melakukan implemtasi pada bahasa pemrograman.

#### 3.6.1 Flowchart Diagram

*Flowchart* merupakan kerangka untuk menunjukkan alur sistem. Diagram ini dapat memberi solusi langkah demi langkah untuk menyelesaikan masalah yang ada dalam algoritma tersebut. Berikut merupakan *flowchart* sistem secara umum.



Gambar 16 Flowchart Sistem Keseluruhan

Keterangan dari Gambar 16 Secara garis besar, peramalan harga saham menggunakan Metode *LSTM* dapat dituliskan sebagai berikut :

1. Mulai

2. Masukan Inputan

Variabel input yang digunakan dalam penelitian ini adalah data jumlah harga saham dari tahun 2014 sampai 2020, yaitu data *close*.

3. Menentukan panjang urutan (*Sequence length*)

Karena dalam pasar modal dalam seminggu ada 5 hari kerja, maka data yang digunakan untuk memprediksi adalah 5 hari kedepan. Jadi perlu dilakukan penentuan panjang urutan dalam pembagian data tiap 5 hari sebagai masukan dan 1 output.

4. Normalisasi Data

Proses Normalisasi yang digunakan dalam sistem ini menggunakan normalisasi minimum-maximum. Data-data yang ada dilakukan normalisasi dengan membagi nilai data tersebut dengan nilai *range* data (nilai data maksimum-nilai data minimum). Normalisasi data *input* bertujuan untuk menyesuaikan nilai *range* data dengan fungsi aktivasi dalam sistem *LSTM*. Ini berarti nilai kuadrat *input* harus berada pada *range* 0 sampai 1. Sehingga *range input* yang memenuhi syarat adalah nilai data *input* dari 0 sampai 1 atau dari -1 sampai 1. Oleh karena itu *output* yang dihasilkan pun akan berada pada *range* 0 sampai 1. kemudian untuk mendapatkan nilai sebenarnya dari *output* perlu dilakukan proses denormalisasi.

5. *Multiple Train-Test Split (K - Fold Validation)*

Proses membagi data menjadi data *training* dan data *testing*. Yang di bagi menjadi 3 skenario.

6. Perhitungan *LSTM*

Terdiri dari beberapa tahap yaitu fase pertama propagasi maju. Kedua propagasi mundur dan yang ketiga perubahan bobot. Ketiga fase tersebut diulang terus hingga kondisi yang diinginkan terpenuhi (jumlah iterasi atau kesalahan *error*). Dalam menggunakan algoritma *LSTM*, proses pelatihan dengan proses pengujian memiliki langkah yang berbeda. Dalam proses pelatihan terdiri dari dua proses utama, *feed forward* dan *backpropagation of error* atau *backward*. Sedangkan untuk proses pengujian hanya menggunakan proses *feed forward*.

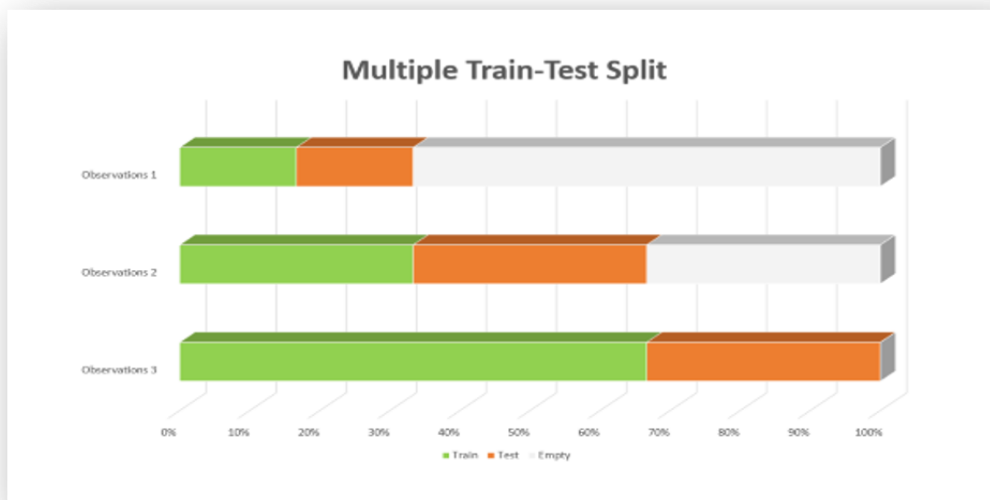
7. Menghasilkan *output*

*Output* yang dihasilkan pada penelitian ini yaitu sistem peramalan harga saham periode berikutnya.

8. Selesai

### 3.6.1 Flowchart Multiple Train-Test Split

Di bawah ini merupakan *Flowchart* sistem proses *K-Folds Cross Validation*.



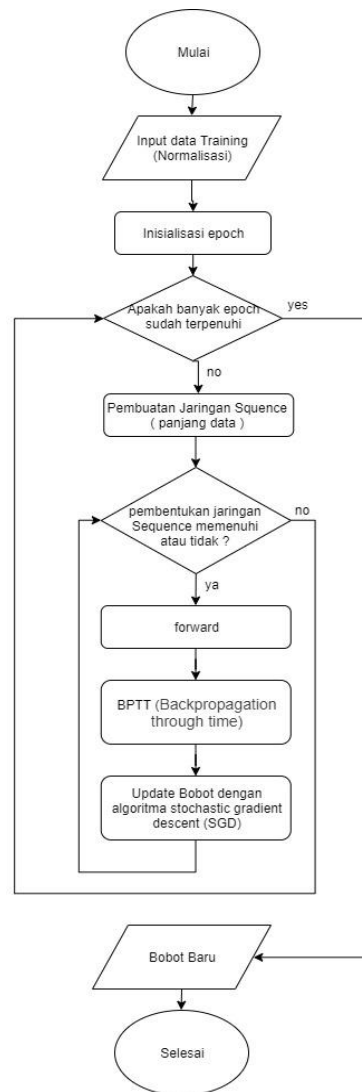
Gambar 17 Multiple Train-Test Split

Menentukan data *training* dan data *testing* pada penelitian ini menggunakan *Multiple Train-Test Split*.

Misal ditentukan pembagian 3 skenario sehingga akan didapat *dataset* skenario 1, skenario 2, dan skenario 3. Dari 3 skenario tersebut akan dipilih 2 menjadi *data training* dan 1 menjadi *data testing*. Misalkan untuk data *training* dipilih skenario 1 dan skenario 2, sedangkan skenario 3 menjadi data *testing*. Semakin banyak pembagian skenario semakin banyak dan semakin bervariasi uji coba yang dilakukan akan tetapi pengerjaan suatu sistem akan semakin lama.

### 3.6.2 Flowchart Tahap Pelatihan LSTM

Tahap ini adalah proses pengenalan pola-pola data yang telah dinormalisasi agar sistem dapat menentukan bobot-bobot yang dapat memetakan antara data input dengan data target output yang diinginkan.

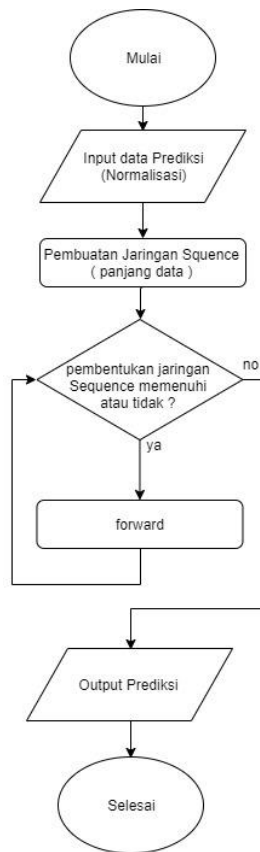


Gambar 18 *Flowchart* pelatihan *LSTM*

Pelatihan ini menggunakan dua looping yaitu looping pertama digunakan seberapa banyak pelatihan dalam setiap urutan (*sequence*) dan looping kedua digunakan untuk menghitung proses pembelajaran dalam pembelajaran *LSTM* menerapkan *BPTT (Backpropagation through time)* dan *update* bobot menerapkan *Stochastic gradient descent (SGD)*

### 3.6.3 Flowchart Tahap Pengujian *LSTM*

Setelah bobot yang terbaik pada tahap pelatihan didapat, maka nilai pembobot tersebut digunakan untuk mengolah data masukan untuk menghasilkan keluaran yang sesuai. Hal ini digunakan untuk menguji apakah pembelajaran *LSTM* dapat bekerja dengan baik yaitu dapat memprediksi pola data yang telah dilatihkan dengan tingkat kesalahan yang kecil.



Gambar 19 *Flowchart* pengujian *LSTM*

### 3.7 Skenario Uji Coba dan Evaluasi

Tahap ini akan dilakukan uji coba Pada penelitian 1500 data yang dibagi menjadi 3 bagian dengan cara *Multiple Train-Test Split*, yang dibagi menjadi 3.

Tabel 2 Uji coba dengan pembagian data

Ujicoba	Trainig	Testing
1	250 DATA	250 DATA
2	500 DATA	500 DATA
3	1000 DATA	500 DATA

### 3.8 Pembuatan Laporan Tugas Akhir

Setelah semua proses selesai maka selanjutnya penulisan laporan Tugas Akhir sesuai dengan sistematika penulisan laporan Tugas Akhir Jurusan Teknik Informatika Universitas Trunojoyo Madura.

Tabel 3. Tabel Perkiraan Jadwal

NO	Kegiatan	Bulan																	
		I		II				III				IV				V			
		3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Persiapan																		
2	Pengumpulan data																		
3	Analisa Kebutuhan Sistem																		
4	Perancangan Sistem																		
5	Implementasi Sistem																		
6	Uji coba Sistem																		
7	Penyusunan laporan Tugas Akhir																		



## DAFTAR PUSTAKA

- [1] A. Santoso and S. Hansun, “Prediksi IHSG dengan Backpropagation Neural Network,” *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 3, no. 2, pp. 313–318, 2019, doi: 10.29207/resti.v3i2.887.
- [2] A. Wanto and A. P. Windarto, “Analisis Prediksi Indeks Harga Konsumen Berdasarkan Kelompok Kesehatan Dengan Menggunakan Metode Backpropagation,” *J. Penelit. Tek. Inform. Sink.*, vol. 2, no. 2, pp. 37–43, 2017.
- [3] D. Karmiani, R. Kazi, A. Nambisan, A. Shah, and V. Kamble, “Comparison of Predictive Algorithms: Backpropagation, SVM, LSTM and Kalman Filter for Stock Market,” *Proc. - 2019 Amity Int. Conf. Artif. Intell. AICAI 2019*, pp. 228–234, 2019, doi: 10.1109/AICAI.2019.8701258.
- [4] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997, doi: 10.1162/neco.1997.9.8.1735.
- [5] A. Arfan, “Prediksi Harga Saham Di Indonesia Menggunakan Algoritma Long Short-Term Memory,” vol. 3, 2019.
- [6] R. Maulana and D. Kumalasari, “Analisis Dan Perbandingan Algoritma Data Mining Dalam Prediksi Harga Saham GGRM,” *J. Inform. Kaputama*, vol. 3, no. 1, pp. 22–28, 2019.
- [7] L. Wiranda, M. Sadikin, J. T. Informatika, and F. I. Komputer, “PENERAPAN LONG SHORT TERM MEMORY PADA DATA TIME SERIES UNTUK MEMPREDIKSI PENJUALAN PRODUK PT . METISKA FARMA,” vol. 8, pp. 184–196, 2019.
- [8] P. J. Werbos, “Backpropagation Through Time: What It Does and How to Do It,” *Proc. IEEE*, vol. 78, no. 10, pp. 1550–1560, 1990, doi: 10.1109/5.58337.
- [9] S. Øyen, “Forecasting Multivariate Time Series Data Using Neural Networks,” *Nor. Univ. Sci. Technol. Dep. Eng. Cybern.*, no. June, 2018, doi: 10.1080/13540602.2012.629837.

- [10] C. Paper, W. Setiawan, and U. Trunojoyo, "Feedforward Network Dengan," no. October, 2016, doi: 10.13140/2.1.3467.5525.
- [11] A. Ghosh, S. Bose, G. Maji, N. Debnath, and S. Sen, "Stock Price Prediction Using LSTM on Indian Share Market," vol. 63, pp. 101–90, 2019, doi: 10.29007/qgcz.
- [12] U. Islam, N. Imam, and B. Padang, "FORECASTING SAHAM SYARIAH DENGAN," 2019.
- [13] R. Yunitarini, P. Studi, T. Informatika, F. Teknik, and U. Trunojoyo, "IMPLEMENTASI METODE BACKPROPAGATION," vol. 1, no. 1, pp. 5–13, 2014.
- [14] Mallya, A. (2017, January 24). *[http://arunmallya.github.io/writeups/nn/lstm/index.html#/. \(A. Mallya, Ed.\)](http://arunmallya.github.io/writeups/nn/lstm/index.html#/)*  
Retrieved from <http://arunmallya.github.io>:  
<http://arunmallya.github.io/writeups/nn/lstm/index.html#/>