

Prediksi Pergerakan Harga Saham Pada Sektor Farmasi Menggunakan Algoritma *Long Short-Term Memory*

Ardiyana Agusta¹, Iin Ernawati², Anita Muliawati³
 Informatika / Fakultas Ilmu Komputer

Universitas Pembangunan Nasional Veteran Jakarta

Jl. Rs. Fatmawati, Pondok Labu, Jakarta Selatan, DKI Jakarta, 12450, Indonesia

¹ ardiyana@upnvj.ac.id, ² iinernawati@upnvj.ac.id, ³ anitamuliawati@upnvj.ac.id

Abstrak. Saham adalah dokumen berharga sebagai bukti kepemilikan bagian suatu perusahaan. Harga saham bersifat fluktuatif dikarenakan berbagai faktor internal dan eksternal perusahaan. Di tengah pandemi Covid-19 yang terjadi sangat berdampak bagi harga saham tiap perusahaan, salah satunya perusahaan yang bergerak di bidang farmasi. Perusahaan farmasi diperkirakan mengalami penurunan saham karena pandemi, tetapi perusahaan juga bisa mendapatkan kenaikan harga saham karena makin banyaknya penjualan dan riset obat bagi masyarakat. Dengan adanya perubahan harga saham yang tak menentu ini maka diperlukan suatu sistem untuk memprediksi pergerakan harga saham. Dalam memprediksi harga saham, penelitian ini menggunakan algoritma *Long Short-Term Memory*, dimana data perusahaan Kalbe Farma dipilih sebagai salah satu perusahaan farmasi. Data yang telah diperoleh dari situs *yahoo finance* akan dilakukan pengolahan data dengan pengujian pada model yang dibentuk dengan menggunakan parameter *hidden layer*, units serta variasi *epoch* dan *batch size* yang menghasilkan hasil prediksi harga saham dengan rerata nilai RMSE 27.310.

Kata Kunci: Prediksi, Saham, *Long Short-Term Memory*

1. Pendahuluan

Dunia investasi dan pasar modal di Indonesia beberapa tahun belakangan ini sudah menjadi perhatian banyak pihak. Bentuk investasi banyak bentuknya, salah satu diantaranya adalah investasi saham. Investasi dalam bentuk saham dapat membuat seorang investor mendapatkan keuntungan yang besar, tetapi dibalik itu terdapat risiko yang besar pula [1]. Investasi saham memiliki risiko yang besar dikarenakan dalam waktu yang singkat terjadinya perubahan harga saham, sehingga menyebabkan keraguan pada calon investor. Fluktuasi dari perubahan harga saham dipengaruhi oleh berbagai faktor, contohnya penawaran dan permintaan, inflasi, suku bunga, serta laporan keuangan dan kinerja perusahaan. Oleh sebab itu, seorang calon investor harus menganalisis pergerakan indeks harga saham sebelum menanamkan modal untuk mengurangi kerugian yang dialami. Prediksi harga saham memungkinkan untuk dapat menjadi referensi tambahan seseorang dalam menjual dan membeli saham sehingga mendapatkan keuntungan maksimal.

Di tengah terjadinya pandemi Covid-19 di era sekarang ini yang menyebar ke seluruh penjuru dunia diperkirakan hal tersebut tidak akan memengaruhi pasar modal, namun keadaan yang terjadi tidak sesuai dengan yang diperkirakan, dikarenakan beberapa perusahaan di dunia sempat mengalami penurunan indeks saham masing-masing [2]. Berkembangannya teknologi di bidang kesehatan membuat perusahaan yang bergerak pada bidang farmasi mampu memproduksi obat-obatan maupun vaksin untuk pencegahan pandemi yang telah terjadi. Disisi pasar modal tentunya dapat membuat para perusahaan yang bergerak di bidang kesehatan ini mendapatkan peningkatan keuntungan harga saham [3].

Perkembangan teknologi yang semakin pesat membuat pemodelan *Machine Learning* dan *Deep Learning* juga mengalami pengembangan, dimana dapat melakukan metode data mining untuk melakukan prediksi pada saham. Pada tahun 1997 pada pengembangan model Jaringan Syaraf Tiruan (JST), dikembangkan model jaringan terbaru untuk menangani terjadinya permasalahan pada dependensi jangka panjang yang biasa dikenal dengan model *Long Short-Term Memory* [1].

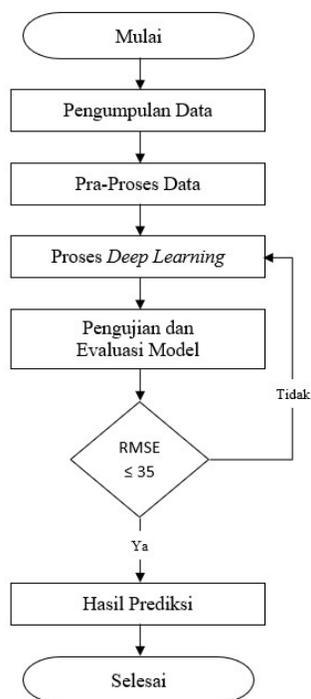
Penelitian terkait pernah dilakukan menggunakan LSTM untuk mengetahui tingkat keakuratan model dalam melakukan peramalan data harga saham dengan pembentukan model yang optimal. Penelitian dilakukan dengan membandingkan tiap model LSTM dengan parameter *optimizer* dan *hidden neuron*. Dari tiap pengujian model yang dibuat mendapatkan hasil peramalan harga saham dan nilai RMSE yang relatif kecil [4].

Dengan melihat hasil penelitian terkait dan permasalahan yang didapati, maka penelitian ini diusulkan untuk memprediksi pergerakan harga saham. Hal ini diusulkan dikarenakan LSTM memiliki kelebihan dalam

memprediksi data yang berbentuk deret waktu, dikarenakan LSTM memprediksi hasil dengan mempelajari memori jangka pendek dan jangka panjang yang disimpan di dalam algoritma.

2. Metodologi Penelitian

Beberapa tahapan penelitian digambarkan dalam *flowchart* sebagai berikut:



Gambar 1. *Flowchart* tahapan penelitian yang dilakukan.

Penjelasan dari tahapan penelitian yang terdapat pada Gambar 1. adalah sebagai berikut:

2.1. Pengumpulan Data

Pada tahap ini, data awal yang digunakan pada penelitian adalah data yang berbentuk deret waktu (*timeseries*) atau data historis. Data *timeseries* mempunyai nilai variabel tiap selang waktu harian yang bersifat numerik. Data saham perusahaan diperoleh dari situs *yahoo finance*, dikarenakan situs tersebut sudah dikenal menjadi salah satu situs untuk mendapatkan berbagai data historis saham suatu perusahaan dan situs ini sifatnya terbuka untuk umum. ada penelitian ini perusahaan yang dipilih adalah PT Kalbe Farma Tbk. dengan kode saham KLBF.JK.

Data yang diperoleh dari situs tersebut meliputi atribut antara lain, *date* (tanggal), *open* (harga pembuka), *close* (harga penutup), *high* (harga tertinggi), *low* (harga terendah), *adjusting close* (harga penutup yang telah disesuaikan) dan *volume* (jumlah saham yang diperdagangkan). Terdapat 761 data yang tersedia selama 3 tahun dari Januari 2018 sampai dengan Desember 2020.

Tanggal merupakan tanggal dimana harga diajukan, harga pembuka merupakan harga pada saat pasar modal buka pada hari tersebut, harga penutup merupakan harga pada saat pasar modal tutup pada hari tersebut, harga tertinggi dan terendah merupakan harga pergerakan pada titik tertinggi dan terendah pada hari tersebut, *adjusted close* merupakan harga saham penutup yang telah disesuaikan dikarenakan adanya kebijakan dari korporasi ketika melakukan *dividen* maupun pembagian saham, *volume* merupakan jumlah saham yang diperdagangkan pada hari tersebut [5]. Data yang telah diperoleh akan di praproses terlebih dahulu untuk mendapatkan bentuk data yang dibutuhkan pada prediksi. Berikut pada **Tabel 1.** disajikan data saham awal perusahaan.

Tabel 1. Data saham awal PT Kalbe Farma Tbk.

No	Date	Open	High	Low	Close	Adj Close	Volume
1	01/01/2018	1690	1690	1690	1690	1566.120117	0
2	02/01/2018	1690	1730	1675	1730	1603.18811	17137900
3	03/01/2018	1730	1730	1675	1695	1570.753662	15571500
...
759	28/12/2020	1465	1505	1455	1500	1471.906372	28002400
760	29/12/2020	1500	1505	1475	1490	1462.093628	26060300
761	30/12/2020	1490	1500	1475	1480	1452.281006	24632300

2.2. Pra-Proses Data

Pada tahap ini, data yang telah diperoleh akan dilakukan pembersihan data untuk menghasilkan data yang berkualitas baik. Tahapan pra-proses yang dilakukan diawali dengan mengecek data apakah terdapat *missing value*, *noise*, atau inkonsistensi pada data. Baris data yang memiliki *missing value* akan berpengaruh buruk dalam proses prediksi, *missing value* biasanya ditandai dengan adanya “?” atau “NaN” yang terdapat pada dataset. Jika terdapat *missing value* maka dapat diatasi dengan melakukan reduksi data, jika tidak maka langkah berikutnya yaitu dengan memilih kolom atribut yang akan digunakan [6]. Data saham pada penelitian setelah dilakukan pengecekan terhadap 761 data menunjukkan bahwa data tidak memiliki *missing value* sehingga tidak perlu melakukan penghapusan baris data pada dataset.

Pada penelitian ini atribut yang dibutuhkan untuk melakukan proses prediksi harga saham hanyalah dua, yaitu atribut *date* dan *close* sehingga atribut selain itu tidak akan digunakan atau dihapus. Pemilihan atribut *date* dibutuhkan tentunya untuk mengetahui tanggal berapa saja terjadinya transaksi saham perusahaan pada bursa efek sedangkan pemilihan atribut *close* dibutuhkan karena harga penutupan adalah harga yang terakhir muncul pada saham sebelum bursa efek ditutup. Kedua atribut ini paling berpengaruh karena dapat mengetahui pola pergerakan harga saham dari waktu ke waktu. Berikut pada **Tabel 2.** disajikan hasil dari pembersihan data serta pemilihan dua atribut.

Tabel 2. Hasil pra-proses data PT Kalbe Farma Tbk.

No	Date	Close
1	01/01/2018	1690
2	02/01/2018	1730
3	03/01/2018	1695
...
759	28/12/2020	1500
760	29/12/2020	1490
761	30/12/2020	1480

Setelah data dari atribut yang telah ditentukan dan dibersihkan, berikutnya data akan melewati tahap normalisasi terlebih dahulu. Pada tahap ini, normalisasi data yang digunakan yaitu dengan metode *MinMax*. Metode *MinMax* digunakan karena *MinMax Scaling* dapat menghasilkan data rentang 0 hingga 1 yang bernilai positif sehingga dapat membantu proses pelatihan model berjalan lebih baik. Proses normalisasi dibutuhkan untuk menghilangkan kerangkapan data serta meminimalisir *error* [7]. Data yang telah dinormalisasi inilah yang nantinya akan digunakan pada tahapan pelatihan dan pengujian model. Atribut yang dinormalisasi yaitu ‘*Close*’ yang memuat data harga saham penutup. Berikut pada **Tabel 3.** disajikan data yang telah dilakukan normalisasi.

Tabel 3. Data Close sebelum dan sesudah dinormalisasi

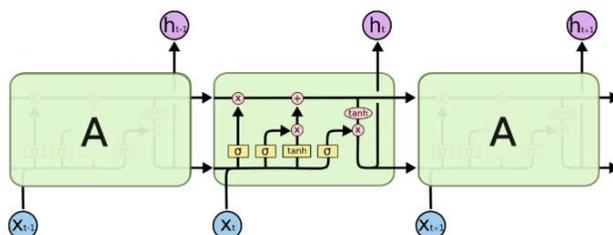
No	Close	Scaled Close
1	1690	0.896739
2	1730	0.940217

3	1695	0.902174
...
759	1500	0.690217
760	1490	0.679348
761	1480	0.668478

Sebelum dilanjutkan pada proses berikutnya, maka data yang sudah di normalisasi akan dilakukan pembagian data menjadi dua bagian yaitu sebesar 80% untuk data latih (*training data*) dan sebesar 20% untuk data uji (*testing data*). Data dibagi tanpa dilakukan pengacakan pada data dikarenakan untuk melakukan prediksi data saham diperlukan data yang berurutan.

2.3. Proses Deep Learning

Pada tahap ini, proses *deep learning* dilakukan dengan algoritma *Long Short-Term Memory* (LSTM) yang dibuat menggunakan bahasa pemrograman python. Pemilihan algoritma LSTM dikarenakan dapat mengatasi hilangnya nilai gradient (nilai mendekati 0 hingga 0) atau bisa disebut *vanishing gradient* yang dilakukan dengan mekanisme *gate* yang dimiliki [8]. LSTM mempunyai empat *layer* yang saling berinteraksi untuk melakukan pengulangan pada tiap modelnya seperti pada **Gambar 2**.



Gambar 2. Struktur model LSTM

Terdapat 3 jenis mekanisme *gates* pada LSTM, antara lain:

1. *Forget gate* sebagai gerbang yang berfungsi untuk menghapus informasi dari *cell*.
2. *Input gate* sebagai gerbang yang berfungsi untuk menentukan dan memperbaharui nilai *input* pada memory state.
3. *Output gate* sebagai gerbang yang berfungsi untuk mendapatkan hasil berdasarkan nilai *input* yang tersimpan pada memori *cell*.

LSTM bekerja melalui tahapan-tahapan sebagai berikut:

1. Tahapan pertama pada LSTM yaitu pada *forget gate layer* akan memutuskan informasi apa saja yang akan dihapus dari *cell state*. Proses yang terjadi pada *forget gate layer* menghasilkan nilai *output* yaitu angka 0 dan 1 yang artinya jika nilai 0 maka tidak membiarkan informasi apapun untuk diteruskan dan nilai 1 membiarkan seluruh informasi untuk diteruskan.

Persamaan pada *forget gate* dijabarkan melalui persamaan (1).

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

2. Tahapan kedua yaitu menentukan informasi apa yang akan disimpan pada *cell state*. Pertama, *input gate layer* menentukan nilai masukkan mana yang akan diperbaharui, kemudian *cell state* menambahkan nilai baru dengan fungsi *tanh layer*. Kedua, memperbaharui *cell state* berdasarkan nilai hasil dari gabungan *input gate layer* dan *tanh layer*.

Persamaan pada *input gate* dijabarkan melalui persamaan (2).

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{2}$$

Persamaan nilai baru dijabarkan melalui persamaan (3).

$$C\bar{t} = \tanh(WC \cdot [ht-1, xt] + bC) \quad (3)$$

- Tahapan ketiga yaitu memperbarui *cell state* menjadi sebuah *cell state* yang baru. Kemudian *cell state* akan dikalikan dengan *state* yang lama melalui penghapusan informasi yang sudah diputuskan terlebih dulu oleh *forget gate layer*. Lalu, hasil yang didapatkan akan ditambahkan dengan nilai terbaru yang digunakan untuk memperbarui *state*.

Persamaan *cell state* dijabarkan melalui persamaan (4).

$$Ct = ft * Ct-1 + it * C\bar{t} \quad (4)$$

- Tahapan keempat yaitu menentukan hasil *output*. Hasil *output* didapatkan berdasarkan *cell state* yang sebelumnya sudah diproses. Awalnya *sigmoid layer* menentukan *cell state* untuk *output*, yang mana hasil dari *cell state* diproses pada *tanh layer* lalu mengalikannya dengan gerbang *sigmoid* untuk menghasilkan *output* yang sesuai dengan putusan sebelumnya.

Persamaan pada *output gate* dijabarkan melalui persamaan (5).

$$ot = \sigma(Wo \cdot [ht-1, xt] + bo) \quad (5)$$

Persamaan untuk mendapatkan nilai *output* orde ke-t dijabarkan melalui persamaan (6).

$$ht = ot * \tanh(Ct) \quad (6)$$

Pembentukan model LSTM diawali dengan menginisialisasi paramater yang dibutuhkan yaitu *hidden layer* (lapisan tersembunyi), *units* (memori sel), *epoch* (putaran), dan *batch size* (jumlah sampel data) [1]. Setelah model dibentuk maka data akan dilatih dengan melewati mekanisme *gates* pada LSTM. Data akan dilatih terus hingga mencapai batas *error* yang diinginkan dengan penentuan serta pengubahan paramater yang digunakan.

Ketika data sudah mencapai target yang diinginkan, proses iterasi akan berhenti dan berikutnya model akan diuji dengan data pengujian atau dapat mengulang kembali proses pelatihan. Proses iterasi ini juga diolah dengan menggunakan fungsi optimasi dan *dropout*. Optimasi berguna untuk menentukan bobot optimal dan mengurangi kesalahan sehingga dapat memaksimalkan keakuratan model. Fungsi optimasi yang digunakan adalah adam, sedangkan *dropout* berguna untuk mencegah terjadinya *overfitting* pada model.

2.4. Pengujian dan Evaluasi Model

Pada tahap ini, setelah data telah dilatih maka data akan diuji dengan membandingkan data yang sudah dihasilkan pada proses pelatihan dengan data uji yang telah ditentukan. Model prediksi akan menghasilkan output berupa data yang masih dalam bentuk normalisasi, sehingga data perlu dirubah untuk mengembalikan nilai atau data ke bentuk semula melalui proses denormalisasi. Tiap model dievaluasi untuk melihat besarnya tingkat keakuratan saat memprediksi harga saham dengan melihat nilai *Root Mean Square Error* (RMSE) yang dihasilkan. Evaluasi dilakukan pada tiap model optimasi berdasarkan paramater yang telah ditentukan. Jika nilai RMSE semakin kecil maka model yang digunakan semakin optimal dalam memprediksi harga saham.

Model dapat dikatakan optimal bila mendapatkan hasil prediksi harga saham dengan nilai RMSE kurang dari sama dengan 35, jika kondisi ini tidak terpenuhi maka akan dilakukan kembali proses *deep learning* hingga mendapatkan hasil yang diinginkan. RMSE digunakan untuk melihat keakuratan model dalam memprediksi dan menjadi acuan validasi model yang telah dihasilkan oleh algoritma LSTM dapat direkomendasikan untuk memprediksi data harga saham.

2.5. Hasil Prediksi

Pada tahap ini, prediksi yang telah didapatkan dari model yaitu hasil prediksi harga saham, hasil ini akan dibandingkan dengan data aktual harga saham sehingga dapat memvalidasi performa model. Data aktual yang dimaksud disini adalah harga saham penutup yang sebenarnya. Hasil prediksi menggunakan algoritma *long short-term memory* didapatkan berdasarkan model dengan konfigurasi parameter paling optimal.

3. Hasil Dan Pembahasan

3.1. Pembagian Data

Data bersih yang telah dinormalisasi, berikutnya akan dibagi menjadi dua yaitu menjadi data latih dan data uji. Pembagian data yang digunakan pada penelitian ini yaitu dengan membagi data dari data total menjadi 80% untuk *training data* (data latih) dan 20% sisanya untuk *testing data* (data uji).

Dikarenakan dalam memprediksi harga saham membutuhkan data historis yang berurutan maka data latih dan data uji tidak diacak pada saat pembagian data, hal ini untuk mendapatkan pola pergerakan harga saham. Jumlah data latih memiliki presentase lebih besar supaya proses pelatihan terlatih dengan baik untuk mempelajari model. Dengan demikian model yang telah dibuat nantinya akan memberikan hasil prediksi yang lebih baik.

Data keseluruhan yang telah dibagi menjadi data latih sebanyak 609 data yang memuat data dari tanggal 01 Januari 2018 hingga 12 Mei 2020. Sedangkan data uji sebanyak 152 data yang memuat data dari tanggal 13 Mei 2020 hingga 30 Desember 2020.

3.2. Pelatihan Model LSTM

Sebelum dilakukan pelatihan pada model LSTM, diawali dengan membuat model LSTM. Pembuatan model dengan algoritma *Long Short-Term Memory* dibuat dengan menggunakan bahasa pemrograman *python*, dimana *library* yang diperlukan yaitu fungsi *sequential* dari *library keras*, menggunakan *optimizer adam* dan *loss function* menggunakan *mean squared error*.

Pelatihan model digunakan untuk mencari model yang dapat memberikan hasil prediksi paling optimal pada tahap pengujian dan evaluasi model dengan mengubah parameter yang dipakai. Parameter yang digunakan pada penelitian ini antara lain dengan menggunakan dua *hidden layer*, jumlah unit, fungsi *loss*, *batch size*, dan jumlah putaran (*epoch*).

Pelatihan model LSTM dilakukan berulang dikarenakan setiap model dilatih selalu menghasilkan hasil prediksi yang berbeda. Perulangan dilakukan sebanyak tiga kali untuk mengetahui model mana yang paling optimal dalam memprediksi harga saham berdasarkan perubahan parameter yang akan dilakukan, pemilihan model yang paling optimal nantinya dilihat dari evaluasi hasil prediksi model dengan *root mean squared error*.

Setelah penentuan parameter yang dibutuhkan, model LSTM bekerja dengan cara sebagai berikut:

1. Pemrosesan LSTM diawali dengan penginisialisasian bobot awal lalu sel menentukan nilai informasi mana yang akan digunakan dan yang tidak, melalui mekanisme *forget gate* dengan menggunakan persamaan (1).
2. Berikutnya dengan mekanisme *input gate* untuk mendapatkan nilai *input* pada sel dengan menghitung persamaan (2) dan (3).
3. Lalu sel akan diperbaharui nilainya dengan mekanisme *cell state* dengan menggunakan persamaan (4).
4. Setelah itu mekanisme *output gate* akan dilakukan untuk menentukan nilai *output* dengan menghitung persamaan (5) dan (6).
5. Proses diatas terus berulang dan berlangsung sebanyak jumlah *epoch* yang sudah ditentukan. Jika sudah maka proses berhenti, tetapi jika masih belum maka akan dioptimasi dengan menggunakan *optimizer adam* dan dengan *loss function* untuk tiap nilai *loss* yang didapatkan.
6. Proses yang telah dilakukan lalu kemudian menghasilkan nilai prediksi, yang berikutnya akan dievaluasi dengan menghitung nilai RMSE dari hasil nilai prediksi yang telah dihasilkan.

3.3. Pengujian Model LSTM

Setelah struktur model LSTM telah selesai dibuat, maka langkah berikutnya yaitu menguji hasil prediksi yang didapatkan oleh tiap model dengan menggunakan data uji yang sudah disiapkan sebanyak 152 data. Proses pengujian model dilakukan dengan proses perubahan parameter yang akan dimulai dari penentuan nilai *epoch* lalu penentuan ukuran *batch*. Sebelum dilakukan pengujian, akan dibuat terlebih dahulu kombinasi parameter *default* dari pengujian model antara lain menggunakan *batch size 25*, jumlah units sebanyak 50, dan nilai *dropout* sebesar 0,2. Parameter yang paling optimal akan ditentukan berdasarkan nilai RMSE terkecil pada tiap model.

3.3.1. Pengujian berdasarkan parameter jumlah epoch

Proses pengujian model pertama dilakukan dengan optimasi berdasarkan jumlah *epoch*. *Epoch* adalah banyaknya proses untuk semua data untuk melewati semua node pada LSTM yang berguna untuk menentukan berapa kali model neural network yang dipakai untuk memproses dataset secara keseluruhan. Daftar *epoch* yang akan diuji pada penelitian antara lain sebanyak 50, 100, 150 dan 200 kali. Setiap *epoch* akan diuji menggunakan nilai parameter yang telah ditentukan pada nilai *default*. Hasil yang didapatkan merupakan rerata hasil program yang dijalankan sebanyak tiga kali, dimana hasil hanya dipilih bila nilai RMSE lebih kecil atau sama dengan 35.

Tabel 4. Hasil uji dengan paramater jumlah *epoch*

Epoch	Batch Size	AvgRMSE	Avg Loss	Avg Time (Detik)
50	25	33.783	0.007974	39.13
100	25	31.708	0.005960	77.76
150	25	29.561	0.005176	115.29
200	25	28.163	0.004269	153.66

Tabel 4. menunjukkan model dengan parameter terbaik berdasarkan jumlah *epoch* yaitu dengan jumlah *epoch* sebanyak 200 yang menghasilkan nilai rerata RMSE sebesar 28.163, rerata nilai loss 0,0042 dan memiliki rerata waktu komputasi selama 153,66 detik. Tabel menunjukkan semakin banyak jumlah epoch yang dilakukan maka semakin kecil nilai rerata RMSE walaupun waktu proses komputasi semakin lama.

3.3.2. Pengujian berdasarkan parameter jumlah *batch size* dan *epoch*

Proses pengujian model kedua yaitu dengan melakukan optimasi berdasarkan jumlah *batch size* pada setiap banyaknya *epoch* yang telah ditentukan. *Batch size* akan menentukan total jumlah sampel data yang disebar dan diproses ke model LSTM. Banyaknya *batch size* akan dibagi dengan jumlah training dataset yang digunakan. Sebagai contoh pada penelitian menggunakan 609 data latih, dan dikarenakan 60 data latih terakhir dijadikan *timestep* pada model, maka menjadi 549 data latih.

Dari banyaknya data latih yang digunakan, supaya dapat menyeimbangkan sebaran sampel data pada penelitian ini, maka penggunaan daftar *batch size* yang akan diuji antara lain sebanyak 10, 25, 35, dan 50. Setiap *batch size* akan diuji menggunakan nilai parameter yang telah ditentukan pada nilai *default* dan dikombinasikan dengan daftar *epoch* pada tahapan sebelumnya. Hasil yang didapatkan merupakan rerata hasil program yang dijalankan sebanyak tiga kali, dimana hanya dipilih bila nilai RMSE lebih kecil atau sama dengan 35.

Tabel 5. Hasil uji *batch size* pada paramater *epoch* sebanyak 50

Batch Size	Epoch	AvgRMSE	Avg Loss	Avg Time (Detik)
10	50	30.344	0.006202	83.27
25	50	32.725	0.008187	41.21
35	50	34.647	0.009652	37.18
50	50	35.272	0.010936	33.77

Tabel 5. menunjukkan model dengan parameter terbaik berdasarkan jumlah *batch size* pada paramater *epoch* sebanyak 50 yaitu dengan *batch size* sebanyak 10 yang menghasilkan nilai rerata RMSE sebesar 30.344, rerata nilai loss 0.0062 dan memiliki rerata waktu komputasi selama 83.27 detik.

Tabel 6. Hasil uji *batch size* pada paramater *epoch* sebanyak 100

Batch Size	Epoch	AvgRMSE	Avg Loss	Avg Time (Detik)
10	100	29.450	0.004221	162.07
25	100	31.480	0.005963	81.37
35	100	33.255	0.006862	75.48
50	100	33.603	0.008232	68.81

Tabel 6. menunjukkan model dengan parameter terbaik berdasarkan jumlah *batch size* pada parameter *epoch* sebanyak 100 yaitu dengan *batch size* sebanyak 10 yang menghasilkan nilai rerata RMSE sebesar 29.450, rerata nilai loss 0.0042 dan memiliki rerata waktu komputasi selama 162.07 detik.

Tabel 7. Hasil uji *batch size* pada parameter *epoch* sebanyak 150

Batch Size	Epoch	AvgRMSE	Avg Loss	Avg Time (Detik)
10	150	28.206	0.003397	243.95
25	150	29.323	0.005071	120.18
35	150	33.256	0.006862	75.48
50	150	33.603	0.008233	68.80

Tabel 7. menunjukkan model dengan parameter terbaik berdasarkan jumlah *batch size* pada parameter *epoch* sebanyak 150 yaitu dengan *batch size* sebanyak 10 yang menghasilkan nilai rerata RMSE sebesar 28.206, rerata nilai loss 0.0033 dan memiliki rerata waktu komputasi selama 243.95 detik.

Tabel 8. Hasil uji *batch size* pada parameter *epoch* sebanyak 200

Batch Size	Epoch	AvgRMSE	Avg Loss	Avg Time (Detik)
10	200	27.310	0.002929	322.28
25	200	28.642	0.004232	159.86
35	200	29.186	0.005180	143.47
50	200	29.814	0.005805	134.34

Tabel 8. menunjukkan model dengan parameter terbaik berdasarkan jumlah *batch size* pada parameter *epoch* sebanyak 200 yaitu dengan *batch size* sebanyak 10 yang menghasilkan nilai rerata RMSE sebesar 27.310, rerata nilai loss 0.0029 dan memiliki rerata waktu komputasi selama 322.28 detik. Setiap tabel dengan pengujian *batch size* pada *epoch* menunjukkan semakin kecil *batch size* maka semakin baik hasil RMSE walaupun waktu proses komputasi semakin lama.

Setelah melihat kelima tabel hasil pengujian pada tiap model, didapatkan dua kondisi kombinasi optimal berdasarkan perubahan parameter yaitu, pertama pada tabel dengan kombinasi parameter dengan perubahan jumlah banyaknya *epoch*, mendapatkan model dengan kombinasi parameter dengan jumlah *epoch* sebanyak 200 yang menghasilkan nilai rerata RMSE sebesar 28.163 dan memiliki rerata waktu komputasi selama 153,66 detik. Tabel menunjukkan bahwa banyaknya *epoch* membuat nilai RMSE semakin kecil walaupun waktu proses komputasi semakin lama.

Kedua, berdasarkan keempat tabel dengan kombinasi parameter dengan perubahan *batch size* dengan tiap jumlah *epoch* mendapatkan model dengan kombinasi parameter dengan jumlah *epoch* sebanyak 200 dan *batch size* 10 yang menghasilkan nilai rerata RMSE sebesar 27.310 dan memiliki rerata waktu komputasi selama 322.28 detik. Setiap empat tabel pengujian menunjukkan bahwa semakin kecilnya *batch size* membuat nilai RMSE semakin kecil walaupun waktu proses komputasi semakin lama. Model dengan kombinasi ini menjadi model yang paling optimal dalam memprediksi harga saham penutup pada PT Kalbe Farma Tbk dengan algoritma *Long Short-Term Memory*.

3.4. Hasil Prediksi

Setelah mencoba berbagai model dengan berbagai kombinasi parameter, maka didapati hasil prediksi harga saham penutup terbaik dengan kombinasi parameter yang paling optimal. Kombinasi parameter tersebut yaitu dengan penggunaan jumlah *epoch* sebanyak 200 dan *batch size* sebesar 10 pada jumlah unit sebanyak 50, nilai *dropout* 0,2 dan dioptimasi dengan adam *optimizer*. Kombinasi parameter pada model tersebut menghasilkan rerata hasil RMSE senilai 27.310, rerata nilai loss 0.0029 dan waktu komputasi selama 322.28 detik.

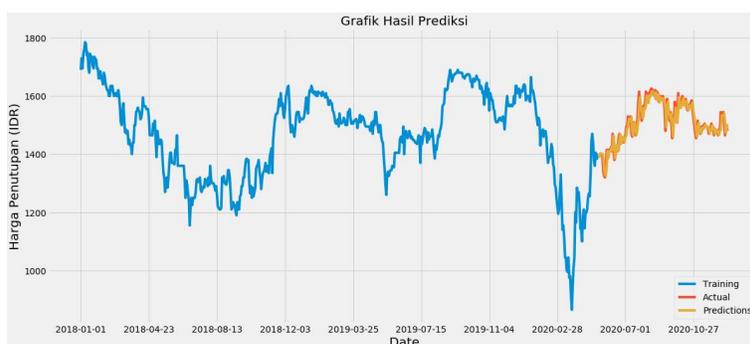
Untuk mendapatkan nilai RMSE pada tiap model, diawali dengan melakukan denormalisasi 152 data hasil prediksi saham yang masih dalam bentuk skala. Proses denormalisasi ini berguna untuk mengembalikan bentuk data yang sudah diskalakan menjadi bentuk data awal. Setelah data harga prediksi saham penutup telah didapatkan melalui proses denormalisasi maka berikutnya dapat dilakukan perhitungan RMSE untuk mengevaluasi model.

Tabel 9. Perbandingan hasil aktual dengan hasil prediksi

No	Tanggal	Harga Aktual	Harga Prediksi
1	13/05/2020	1400	1390.570
2	14/05/2020	1400	1406.516
3	15/05/2020	1400	1407.017
...
150	28/12/2020	1500	1475.318
151	29/12/2020	1490	1503.025
152	30/12/2020	1480	1497.414

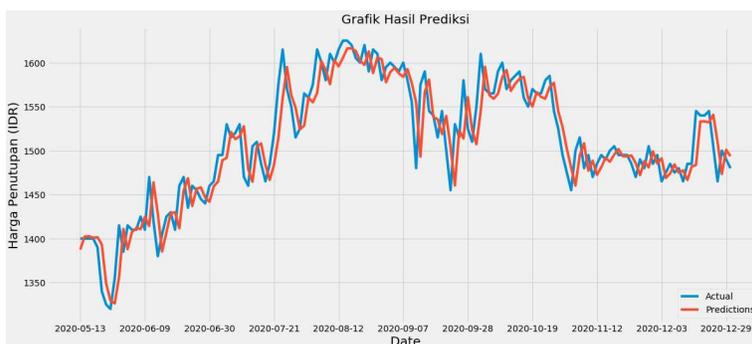
Tabel 9. menunjukkan perbandingan harga penutupan saham aktual dengan harga saham hasil prediksi. Dapat dilihat bahwa harga yang dihasilkan oleh prediksi paling optimal cukup mendekati harga saham penutup yang sebenarnya (harga saham aktual).

Untuk melihat visualisasi dari perbandingan data harga saham aktual dengan harga saham penutup hasil prediksi menggunakan model LSTM dengan parameter paling optimal ditampilkan pada grafik sebagai berikut:



Gambar 3. Grafik Hasil Prediksi Harga Saham Keseluruhan

Grafik yang dihasilkan pada **Gambar 3.** menunjukkan gambaran penuh bagaimana fluktuasi pergerakan harga saham penutup PT Kalbe Farma Tbk dari bulan Januari 2018 hingga bulan Desember 2020. Garis biru pada grafik menunjukkan data harga saham penutup untuk data latih, garis merah menunjukkan data harga saham penutup untuk data uji, dan garis kuning menunjukkan data hasil prediksi harga saham penutup.



Gambar 4. Grafik Hasil Prediksi Harga Saham

Grafik yang dihasilkan pada **Gambar 4.** menunjukkan gambaran fluktuasi harga saham penutup PT Kalbe Farma Tbk. pada data uji yaitu 152 hari terakhir yang dimulai dari tanggal 13 Mei 2020 hingga 30 Desember 2020. Garis biru pada gambar menunjukkan data harga saham penutup asli (data aktual) dan garis merah menunjukkan data harga saham penutup hasil prediksi. Visualisasi grafik yang ditampilkan mengilustrasikan bahwa alur pergerakan

data hasil prediksi dengan metode LSTM mempunyai bentuk yang menyerupai grafik harga asli saham penutup pada perusahaan.

4. Penutup

4.1. Kesimpulan

Berdasarkan penelitian yang telah dilakukan maka dapat disimpulkan bahwa performa yang dihasilkan oleh kombinasi parameter terbaik yang digunakan oleh model LSTM dengan berdasarkan pengubahan jumlah *batch size* dengan tiap jumlah *epoch*, yaitu dengan menggunakan jumlah epoch sebanyak 200, ukuran *batch size* sebesar 10 serta jumlah unit 50 yang dioptimasi dengan adam. Kombinasi tersebut menghasilkan rerata nilai RMSE sebesar 27.310 dengan rerata waktu komputasi selama 322.28 detik. Maka model ini dapat menjadi model paling optimal untuk memprediksi harga saham penutup pada perusahaan Kalbe Farma. Hasil dari visualisasi yang ditampilkan oleh grafik serta tabel prediksi yang dihasilkan oleh *algoritma long short-term memory* menunjukkan tingkat keakuratan model dengan baik dalam memprediksi pergerakan harga saham penutup suatu perusahaan yang bergerak pada sektor farmasi.

4.2. Saran

Saran dari penelitian yang telah dibuat antara lain menggunakan kombinasi parameter lain pada *algoritma long short-term memory* ataupun menggunakan algoritma lain untuk menjadi perbandingan dalam memprediksi harga saham dengan tingkat ketepatan yang lebih baik. Serta diharapkan dapat melakukan prediksi harga penutup saham pada hari mendatang diluar dari data saham yang dimiliki.

Referensi

- [1] A. Arfan and L. ETP, "Perbandingan Algoritma Long Short-Term Memory dengan SVR Pada Prediksi Harga Saham di Indonesia," *PETIR*, 2020, doi: 10.33322/petir.v13i1.858.
- [2] A. Satyo, B. Karno, W. Hastomo, E. Nisfiani, and S. Lukman, "Optimais Deep Learning untuk Prediksi Data Saham Di Era Pandemi Covid -19," *SANTEI*, pp. 43–54, 2020.
- [3] J. V Mangindaan and H. Manossoh, "Analisis Perbandingan Harga Saham PT Garuda Indonesia Persero (Tbk.) Sebelum dan Sesudah Pandemi Covid-19," *J. Adm. Bisnis (JAB ...)*, 2020.
- [4] R. Yotenka and F. F. El Huda, "Implementasi Long Short-Term Memory Pada Harga Saham Perusahaan Perkebunan Di Indonesia," *Unisda J. Math. Comput. Sci.*, vol. 6, no. 01, pp. 9–18, 2020, doi: 10.52166/ujmc.v6i01.1927.
- [5] M. Awaludin and Y. Z. Rahwanto, "PENGEMBANGAN ALGORITMA NEURAL NETWORK BERDASARKAN RENTANG WAKTU UNTUK PREDIKSI HARGA PERDAGANGAN VALUTA ASING," *CKI SPOT*, 2017.
- [6] R. B. Purnama, "Perancangan Prediksi Untuk Menentukan Indeks Harga Saham Menggunakan Jaringan Syaraf Tiruan," *Kinetik*, 2017, doi: 10.22219/kinetik.v2i2.190.
- [7] M. L. Ashari and M. Sadikin, "Prediksi Data Transaksi Penjualan Time Series Menggunakan Regresi Lstm," *J. Nas. Pendidik. Tek. Inform.*, vol. 9, no. 1, p. 1, 2020, doi: 10.23887/janapati.v9i1.19140.
- [8] M. K. Nammous, K. Saeed, and P. Kobojeck, "Using a small amount of text-independent speech data for a BiLSTM large-scale speaker identification approach," *J. King Saud Univ. - Comput. Inf. Sci.*, 2020, doi: 10.1016/j.jksuci.2020.03.011.