

Implementasi *Long Short-Term Memory* pada Prediksi Harga Saham PT Aneka Tambang Tbk

Sari Dwi Kartika dan Karmilasari

Program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Gunadarma
Jl. Margonda Raya No. 100, Depok 16424, Jawa Barat
E-mail: saridwikartika@student.gunadarma.ac.id, karmila@staff.gunadarma.ac.id

Abstrak

Salah satu perusahaan yang termasuk dalam Bursa Efek Indonesia adalah PT Aneka Tambang Tbk (PT Antam), perkembangan saham PT Antam mengalami penurunan dan kenaikan harga yang tidak stabil sehingga sangat sulit untuk di prediksi. Namun, dengan adanya *machine learning* para pemilik saham dapat memprediksi pergerakan harga saham dengan menggunakan data harga saham sebelumnya yang ada pada website harga saham. Penelitian ini bertujuan untuk melakukan implementasi dan analisis dalam memprediksi harga saham PT Aneka Tambang Tbk (PT Antam) menggunakan metode *Long Short Term Memory* (LSTM). Penelitian ini menggunakan data harga saham PT Aneka Tambang Tbk pada *website finance.yahoo.com* periode 29 September 2005 hingga 13 April 2021. Tahapan penelitian ini meliputi : pengumpulan data, *data preprocessing* (terdiri dari cleansing data, scaling data, dan pembagian data), olah data LSTM (terdiri dari perancangan dan pelatihan model LSTM), perhitungan tingkat *error* dan akurasi, dan visualisasi hasil prediksi. Pada penelitian ini dilakukan 6 kali percobaan menggunakan epochs yang berbeda – beda. Hasilnya, percobaan yang memiliki hasil prediksi harga saham yang data prediksinya paling mendekati dengan harga aktual adalah menggunakan epochs 100 dengan *Root Mean Squared Error* (RMSE) sebesar 0.3304 dan *R squared* (R²) sebesar 0.9902.

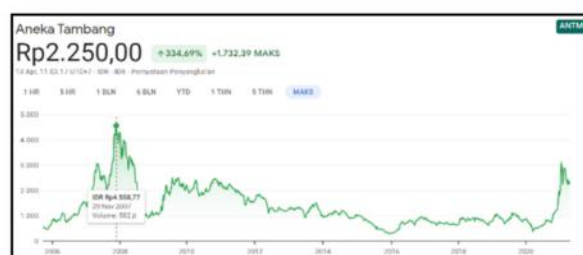
Kata kunci : *Machine Learning*, LSTM, Prediksi Harga Saham, RMSE, *R Squared*

Pendahuluan

Bursa Efek Indonesia merupakan salah satu dari bursa saham yang terdapat di Indonesia. Tercatat telah ada 735 perusahaan yang terdaftar di Bursa Efek Indonesia, salah satunya adalah PT Aneka Tambang Tbk [1]. PT Aneka Tambang Tbk. atau biasa disebut dengan PT Antam merupakan perusahaan di bidang pertambangan yang 65 persen sahamnya dipegang oleh Inalum dan 35 persen milik publik dengan kepemilikan saham kurang dari 5 persen. PT Antam didirikan pada tanggal 5 Juli 1968. Melalui wilayah operasi yang tersebar di seluruh Indonesia yang kaya akan bahan mineral, kegiatan PT Antam mencakup eksplorasi, penambangan, pengolahan serta pemasaran dari komoditas bijih nikel, feronikel, emas, perak, bauksit dan batubara [2].

Pada Gambar 1 yang bersumber dari *Google Finance* [3], dapat dilihat bahwa perkembangan harga saham PT Antam mengalami penurunan dan kenaikan yang tidak stabil. Dimana dalam kurun waktu tahun 2006 hingga 13 April 2021 terjadi kenaikan harga saham tertinggi pada 29 November 2007 dengan harga Rp4.558,77 per lembar saham.

Lalu pada 13 April 2021 harga saham PT Antam ada di harga Rp.2.250 per lembar saham.



Gambar 1: Grafik Harga Saham PT Antam Tahun 2006 – 13 April 2021

Harga saham bergerak sangat cepat dan dinamis atau tidak memiliki alur yang pasti. Bagi pemilik saham mengetahui dan memahami pergerakan harga saham dengan baik akan memberikan keuntungan, dengan cara menjual kepemilikan saham jika harga saham sedang naik ataupun membeli kepemilikan saham pada saat harga saham sedang

turun atau anjlok. Pergerakan harga saham dapat di prediksi agar para pemilik saham dapat memperkirakan kapan akan menjual dan membeli kepemilikan saham. Prediksi saham salah satunya dapat menggunakan teknologi terkini yaitu dengan *Machine Learning*.

Saat ini *machine learning* dapat menjadi penyelesaian permasalahan dalam memprediksi saham kedepannya. *Machine learning* dapat diartikan sebagai metode komputasi berdasarkan pengalaman untuk meningkatkan performa atau membuat prediksi yang akurat. Arti dari pengalaman disini adalah informasi sebelumnya yang telah tersedia dan dapat dijadikan data pembelajar. [4]. Pembuatan algoritma atau model statistik yang dilatih dapat membuat prediksi pada data tertentu merupakan tugas utama dari machine learning. Salah satu metode yang dapat digunakan dan baik dalam melakukan prediksi yaitu metode *Long Short Term Memory* atau biasa disingkat LSTM. LSTM merupakan salah satu jenis dari *Recurrent Neural Network* (RNN) dimana dilakukan modifikasi pada RNN dengan menambahkan *memory cell* yang dapat menyimpan informasi untuk jangka waktu yang lama[5]. LSTM telah banyak digunakan untuk memprediksi seperti pada Prediksi Nilai Tukar Mata Uang IDR terhadap USD, Prediksi Mata Uang Bitcon, Prediksi Data Transaksi Penjualan.

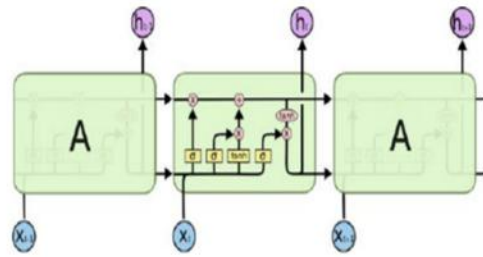
Penelitian ini bertujuan untuk melakukan prediksi harga saham pada PT Aneka Tambang Tbk menggunakan metode LSTM. Tingkat kesalahan pada model prediksi hitung dalam penelitian ini akan dihitung menggunakan *Root Mean Square Error* (RMSE) dan akurasi dihitung menggunakan *R Squared* (R2).

Metode Penelitian

Pada metode penelitian akan dijelaskan mengenai tahapan yang akan dilakukan dalam melakukan penelitian tentang implementasi *Long Short Term Memory* (LSTM) pada prediksi harga saham PT Aneka Tambang Tbk, mulai dari Analisa kebutuhan, pengumpulan dataset, preprocessing, olah data LSTM, hitung nilai *error* dan akurasi, serta visualisasi grafik hasil prediksi.

Long Short Term Memory (LSTM)

Pada Jurnal Laras Wiranda, Mujiono Sadikin *Long Short Term Memory* (LSTM) disebutkan untuk pertama kali pada tahun 1997 dijelaskan oleh Hochreiter dan Schmidhuber. LSTM disebut juga sebagai jaringan saraf dengan arsitektur yang mudah beradaptasi, sehingga bentuknya dapat disesuaikan, tergantung pada aplikasinya [5]. Semua RNN memiliki bentuk rangkaian modul jaringan syaraf yang berulang. LSTM juga memiliki struktur yang sama namun memiliki tambahan fitur berupa gerbang pada sel [6].



Gambar 2: . Detail struktur LSTM [7]

LSTM akan menentukan informasi apa yang akan dibuang dari sel. Keputusan ini dibuat oleh *forget gate layer*. Layer ini akan memperhatikan h_{t-1} dan x_t sehingga akan menghasilkan keluaran antara 0 dan 1. Keluaran 0 merepresentasikan bahwa informasi akan dilupakan sedangkan keluaran 1 merepresentasikan bahwa informasi tidak akan dilupakan [8].

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

dengan f_t adalah *forget gate layer*, σ adalah fungsi sigmoid, W_f dan b_f merupakan matriks, h_{t-1} adalah vektor *hidden state* dalam *timestep* sebelumnya $t-1$, x_t adalah input data (vektor input x dalam *timestep* t).

Langkah selanjutnya adalah menentukan apakah informasi akan disimpan pada sel. Pertama, sebuah layer sigmoid bernama "*input gate layer*" menentukan nilai mana yang akan diperbaharui. Selanjutnya, sebuah *layer tanh* membuat vektor dari nilai kandidat baru (C_t) yang dapat ditambah ke *state*. Langkah selanjutnya, kedua layer ini akan dikombinasikan untuk memperharui *state*.

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

dengan i_t adalah *input gate layer*, W_i dan b_i adalah matriks.

$$\tilde{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \quad (3)$$

dengan \tilde{C}_t adalah nilai kandidat baru, \tanh adalah, W_c dan b_c adalah matriks.

Selanjutnya, *state* lama akan diperbaharui, C_{t-1} ke *state* sel baru C_t . Kemudian, f_t akan dikalikan dengan *state* lama dengan mengabaikan informasi yang sudah dilupakan sebelumnya. Lalu, i_t ditambahkan dengan C_t .

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad (4)$$

dengan C_t adalah *cell state* Baru, f_t adalah nilai *forget gate layer*, C_{t-1} adalah *cell state* sebelumnya, dan i_t adalah nilai dari *input gate layer*.

Langkah terakhir adalah menentukan apa keluarannya. Pertama, *Layer sigmoid* akan menentukan bagian dari *sel* yang akan dikeluarkan. Kemudian, *sel* tersebut akan dilewatkan pada *Layer tanh* (untuk memaksa nilai keluaran diantara -1 dan 1) dan mengalikan dengan keluaran dari gerbang sigmoid.

$$O_t = (W_o [ht-1, xt] + b_o) \quad (5)$$

$$O_t = (W_o . [ht - 1, xt] + b_o) \quad (5)$$

dengan O_t adalah output gate layer, W_o dan b_o adalah matriks.

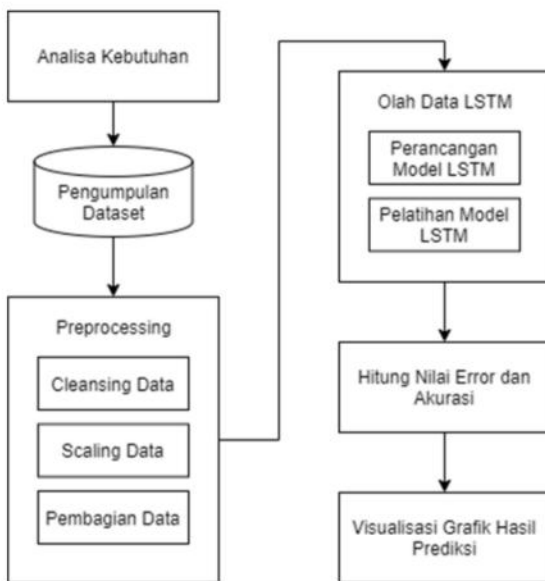
$$H_t = O_t * \tanh(C_t) \quad (6)$$

$$H_t = O_t * \tanh(C_t) \quad (6)$$

dengan H_t adalah *hidden state*, dan o_t adalah nilai *output gate layer*.

Tahapan Perancangan Program

Tahapan – tahapan dalam perancangan program dapat dilihat pada Gambar 3 tentang tahapan yang akan dilakukan dalam perancangan program.



Gambar 3: Tahapan Perancangan Program

1. Analisa Kebutuhan

Pada tahapan ini analisa kebutuhan yang diperlukan adalah kebutuhan dari perangkat keras (*hardware*) maupun kebutuhan perangkat lunak (*software*).

2. Pengumpulan Dataset

Pada tahapan pengumpulan data, data yang digunakan yaitu berupa data kenaikan dan penurunan harga pada saham PT Aneka Tambang Tbk (PT Antam) periode waktu 29 September 2005 hingga 13 April 2021 yang mana data tersebut diperoleh dari *website yahoo.finance.com* yang berjumlah 3806 dataset.

3. Preprocessing

Pada tahapan ini akan dilakukan tahap *preprocessing*, dimana sebelum data digunakan maka perlu melakukan *preprocessing* terlebih dahulu agar data dapat siap dipakai dengan model arsitektur yang dibangun. *Preprocessing* yang dilakukan yaitu *cleansing*, *scaling* dataset, pembagian data.

4. Olah Data LSTM

Pada tahapan ini akan dilakukan perancangan pada model LSTM yaitu melakukan inialisasi parameter yang diproses oleh LSTM, parameter yang digunakan dan pelatihan model LSTM yaitu akan dilakukan beberapa percobaan menggunakan *epochs* yang berbeda-beda agar dapat mengetahui nilai yang akan menghasilkan prediksi terbaik.

5. Hitung Nilai Error dan Akurasi

Tahapan ini dilakukan menggunakan *Root Mean Squared Error* (RMSE) untuk mencari nilai kesalahan pada program prediksi dan *R Squared* (R2) untuk mencari nilai akurasi.

6. Visualisasi Grafik Hasil Prediksi

Tahapan ini dilakukan untuk memudahkan para pembeli ataupun penjual saham dalam membaca hasil prediksi harga saham, dimana grafik tersebut akan menampilkan harga prediksi dan juga harga sebenarnya dari data harga saham dalam periode tertentu.

Hasil dan Pembahasan

Analisa Kebutuhan

Berikut kebutuhan hardware dan software yang dibutuhkan untuk pembuatan program :

1. Laptop Lenovo,
2. Central Processing Unit (CPU), Processor Intel Core i7 Generasi 6,
3. Random Access Memory (RAM), 8 GB,
4. Graphics Processing Unit (GPU), NVIDIA GeForce GTX 960M,
5. Microsoft Office 2016,
6. Bahasa Pemrograman Python 3,
7. Library Python,
8. Google Colaboratory.

Hasil Pengumpulan Dataset

Data yang digunakan dalam penelitian ini adalah data harga saham PT Aneka Tambang Tbk (PT Antam) dari periode 29 September 2005 hingga 13 April 2021 yang di dapatkan melalui *website finance.yahoo.com*. Berikut ini merupakan sintaks program untuk mengambil dataset yang digunakan untuk program:

```
dataframe = yfi.download(tickers=
'ANTM.JK', start='2005-09-
29', end='2021-04-14')

dataframe
```

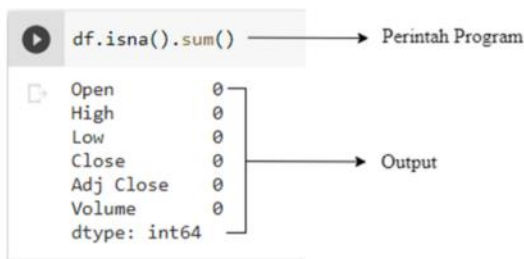

Dapat dilihat pada Gambar 4 dimana menampilkan data frame yang digunakan pada program yang memiliki 6 kolom yaitu *Open*, *High*, *Low*, *Close*, *Adj Close*, *Volume* dan 1 *index* yaitu *Date* serta terdapat 3806 *records*. Pada program ini digunakan kolom “*Close*” untuk memprediksi harga saham pada PT Antam dimana kolom “*Close*” merupakan harga penutupan pada hari tertentu.

Date	Open	High	Low	Close	Adj Close	Volume
2005-09-29	432.588074	436.787964	407.388763	432.588074	302.704529	76180670
2005-09-30	457.787384	457.787384	432.588074	457.787384	320.337860	105493978
2005-10-03	470.387024	474.586914	440.987854	470.387024	329.154419	59712955
2005-10-04	474.586914	482.986694	466.187164	474.586914	332.093292	56236668
2005-10-05	482.986694	482.986694	466.187164	482.986694	337.971069	31319315
...
2021-04-07	2330.000000	2370.000000	2270.000000	2360.000000	2342.971436	237921000
2021-04-08	2360.000000	2480.000000	2340.000000	2470.000000	2452.177734	327839300
2021-04-09	2450.000000	2490.000000	2340.000000	2340.000000	2323.115723	264837100
2021-04-12	2310.000000	2310.000000	2210.000000	2250.000000	2233.765137	194738500
2021-04-13	2250.000000	2300.000000	2220.000000	2270.000000	2253.620850	111017700

Gambar 4: Data *Frame* Pada Program

Hasil Data Cleansing

Sebelum dataset digunakan untuk program, maka akan dilakukan data *cleansing*. Data *cleansing* adalah proses mendeteksi nilai yang kosong, *missing value* atau tidak akurat pada dataset yang digunakan program. Perintah untuk mengetahui karakter atau kata yang tidak diperlukan dapat dilihat pada Gambar 5 berikut.



Gambar 5: Perintah Program dan Hasil dari *Missing Value*

Gambar 5 merupakan perintah untuk mengetahui kolom pada dataset yang memiliki nilai kosong, kemudian hasil dari perintah tersebut menunjukkan bahwa kolom *Open*, *High*, *Low*, *Close*, *Adj Close*, *Volume* menghasilkan output berupa angka “0” berarti semua kolom tersebut tidak memiliki missing value atau nilai yang kosong. Pada Tabel 1 ditampilkan hasil dari data *cleansing* yang telah dilakukan dimana tidak ada missing value pada dataset. Dapat dilihat pada kolom *Open*, *High*, *Low*, *Close*, *Adj Close*, dan *Volume* memiliki hasil “0” yaitu tidak ada missing value atau tidak ada nilai yang kosong pada setiap kolom.

Tabel 1: Hasil Data Cleansing

<i>Open</i>	<i>High</i>	<i>Low</i>	<i>Close</i>	<i>Adj Close</i>	<i>Volume</i>
0	0	0	0	0	0

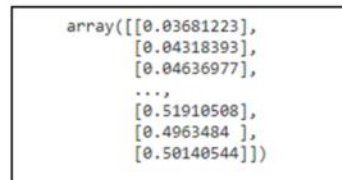
Hasil Data Scaling

Data scaling dilakukan untuk meminimalkan error pada program, dimana data scaling pada penelitian ini menggunakan metode Min-Max Scaling. Metode tersebut dapat mengubah ukuran data dari rentang asli, menjadi nilai dengan rentang 0 hingga 1. Persamaanya dapat dilihat pada persamaan (7) [9].

$$X_i = New_{min} + (New_{max} - New_{min}) \cdot \left(\frac{(x_i - x_{min})}{(x_{max} - x_{min})} \right) \tag{7}$$

dengan X_i adalah data hasil *scaling*, New_{min} adalah batas minimum yang diberikan, New_{max} adalah batas maksimum yang diberikan, x_i adalah data nilai asli, x_{min} adalah nilai minimum dari data per kolom, dan x_{max} adalah nilai batas maksimum dari data per kolom.

Dapat dilihat pada Gambar 6 dimana gambar tersebut menampilkan hasil dari output program dari data *scaling* yang nilai datanya telah berubah menjadi rentang 0 hingga 1.



Gambar 6: Output Program Data Scaling

Tabel 2: Hasil Data Scaling

Sebelum Scaling	Sesudah Scaling
457,7874	0,043184
470,387	0,04637
474,5869	0,047432
482,9867	0,049556
466,1872	0,045308
...	...
2330	0,516577
2360	0,524162
2470	0,551976
2340	0,519105
2250	0,496348

Untuk mempermudah dalam pembacaan output program dari data scaling maka dibuat tabel yang berisikan hasil data sebelum dan sesudah scaling,

dapat dilihat pada Tabel 2 dimana terdapat hasil data sebelum dan sesudah scaling, sebelum scaling nilai dalam rentang nilai 432 – 2270 sedangkan sesudah scaling nilai data berubah menjadi nilai dalam rentang 0 – 1

Hasil Pembagian Data

Sebelum data dapat digunakan untuk membuat hasil prediksi, data akan dibagi agar mendapatkan nilai prediksi yang mendekati nilai sebenarnya dan menghasilkan akurasi yang baik. Dalam pembagian data ini, data akan dibagi menjadi 2 bagian yaitu data train digunakan untuk melatih data, dan data test digunakan untuk menghasilkan data hasil prediksi. Berikut adalah sintaks program untuk pengambilan 80% data untuk data *train*.

```
training_data_size = math.ceil(len
(data_saham)*.8)
```

Dapat dilihat pada Tabel 3 terdapat data keseluruhan yang berjumlah 3806 data, dimana data train menggunakan 80% dari jumlah data seluruhnya yaitu sebanyak 3045 data dan data test menggunakan 20% dari jumlah data yaitu sebanyak 761 data.

Tabel 3: Hasil Pembagian Data

Jumlah Data	Data Train (80%)	Data Test (20%)
3806	3045	761

Berdasarkan hasil perhitungan yang dilakukan pada setiap kategori kriteria penilaian dapat diperoleh kesimpulan bahwa setiap kriteria yang diuji berada pada kategori “sangat baik (SB)”. Berdasarkan hasil perhitungan akhir, pengembangan aplikasi pembelajaran bahasa Korea mendapatkan persentase nilai indeks akhir sebesar 89,91%. Nilai persentase tersebut menunjukkan bahwa pengembangan aplikasi pembelajaran bahasa Korea berada dalam kategori “sangat baik” secara keseluruhan. Dapat dikatakan bahwa aplikasi telah memenuhi tujuan penelitian, yaitu membantu dan mempermudah pengguna untuk mempelajari bahasa Korea berbasis *mobile learning* tanpa membutuhkan jaringan internet

Hasil Pengujian Model

Hasil pengujian model akan didapatkan setelah membangun model LSTM dan melakukan pelatihan data. Pada table 4 ditamapilkan model LSTM yang akan digunakan pada penelitian ini yaitu 1 *layer* LSTM dengan menggunakan 64 *neuron*, *dropout*

untuk mencegah terjadinya *overfitting* dan mempercepat *learning* pada program, *batch size* 32, dan *output layer* dengan 1 *neuron*, perhatikan Tabel 4.

Tabel 4: Model LSTM Pada Program

Parameter	Jumlah
1 Layer LSTM	64 neuron
Dropout	0.1
Batch Size	32
Output Layer	1 neuron

Lalu pelatihan data digunakan dengan beberapa *epochs* atau putaran yang berbeda agar mendapatkan hasil prediksi yang akurasinya baik sehingga mendapatkan nilai yang mendekati nilai sebenarnya. Dapat dilihat pada Tabel 5 yaitu terdapat nilai dari hasil prediksi yang telah dilakukan, dimana digunakan 6 *epochs* yang berbeda yaitu 10, 15, 25, 50, 100, 150 dan *batch size* 32 pada periode waktu tertentu yaitu dari 07 April hingga 13 April 2021 dimana setiap *epochs* memiliki hasil prediksi yang berbeda-beda.

Hasil Perhitungan Tingkat Error dan Akurasi

Hasil perhitungan tingkat *error* menggunakan *Root Mean Square Error* (RMSE) yaitu metode alternatif untuk mengevaluasi kinerja model dalam peramalan yang digunakan dimana hasil RMSE yang baik yaitu makin mendekati nilai 0 [10]. Hasil perhitungan tingkat akurasi menggunakan R^2 (*R Squared*) yaitu angka yang berkisar antara 0 sampai 1 yang mengindikasikan besarnya kombinasi variabel independen secara bersama – sama mempengaruhi nilai variabel dependen. Semakin mendekati angka satu, model yang dikeluarkan oleh regresi tersebut akan semakin baik [11].

Gambar 7 menampilkan tentang perintah dari program untuk menghitung nilai kesalahan menggunakan RMSE dan Gambar 8 menampilkan tentang perintah untuk menghitung nilai akurasi menggunakan R^2 untuk menghitung nilai akurasi pada program.

```
rmse = np.sqrt( np.mean( prediksi - y_test )**2 )
rmse
```

Gambar 7: Perintah Program Menghitung RMSE

```
r_squared = r2_score(y_test, prediksi)
print(f'R-squared: {r_squared}')
```

Gambar 8: Perintah Program Menghitung R^2

Tabel 5: Data Hasil Prediksi Periode Tertentu

No.	Epochs	Periode	Data Valid	Hasil Prediksi
1.	10	07/04/21	2.360	2.223,491
		08/04/21	2.470	2.241,808
		09/04/21	2.340	2.279,57
		12/04/21	2.250	2.299,013
		13/04/21	2.270	2.298,94
2.	15	07/04/21	2.360	2.264,1433
		08/04/21	2.470	2.289,1624
		09/04/21	2.340	2.338,894
		12/04/21	2.250	2.361,9707
		13/04/21	2.270	2.355,522
3.	25	07/04/21	2.360	2.264,1433
		07/04/21	2.360	2.280,548
		08/04/21	2.470	2.318,152
		09/04/21	2.340	2.386,174
		12/04/21	2.250	2.398,239
4.	50	13/04/21	2.270	2.367,849
		07/04/21	2.360	2.260,084
		08/04/21	2.470	2.321,834
		09/04/21	2.340	2.415,803
		12/04/21	2.250	2.373,681
5.	100	13/04/21	2.270	2.277,585
		07/04/21	2.360	2.322,2766
		08/04/21	2.470	2.362,9866
		09/04/21	2.340	2.468,1243
		12/04/21	2.250	2.349,8584
6.	150	13/04/21	2.270	2.253,1428
		07/04/21	2.360	2.380,803
		08/04/21	2.470	2.416,131
		09/04/21	2.340	2.529,403
		12/04/21	2.250	2.404,523
		13/04/21	2.270	2.314,912

Tabel 6: Hasil Perhitungan RMSE dan R²

Epochs	Nilai RMSE	Nilai R ²
10	3,8641	0,9722
15	2,4339	0,9761
25	1,3369	0,9849
50	1,4327	0,9866
100	0,2920	0,9900
150	1,0739	0,9899

Visualisasi Hasil Prediksi

Visualisasi hasil prediksi akan dibuat dalam bentuk grafik menggunakan library matplotlib. Library matplotlib adalah salah satu dari library yang ada pada bahasa pemrograman python yang dengan baik bisa membuat visualisasi data yang statis dan juga interaktif dalam bentuk dua dimensi (2D) dan tiga dimensi (3D).

Pada Gambar 9 hingga 14 akan menampilkan grafik – grafik hasil visualisasi data yang merupakan *training* dan *validation loss* dari tiap percobaan yang menggunakan epochs berbeda – beda. Grafik tersebut berisi 2 garis dengan warna berbeda, garis berwarna hijau merupakan representasi dari *training loss*, sedangkan garis berwarna merah merupakan representasi dari *validation loss*.

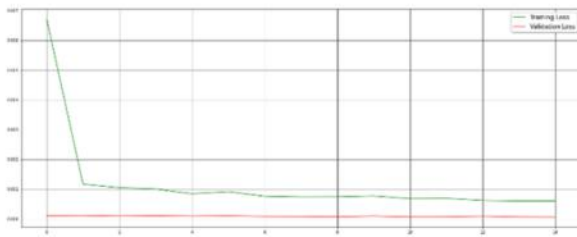


Gambar 9: Visualisasi Grafik *Training* dan *Validation Loss* Pada *Epochs* 10

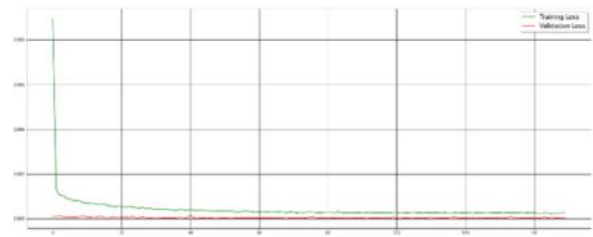


Gambar 10: . Visualisasi Grafik *Training* dan *Validation Loss* Pada *Epochs* 15

Dapat dilihat pada Tabel 6 yaitu hasil perhitungan RMSE dan R² dimana hasil yang paling baik terdapat pada pengujian model dengan jumlah epochs 100 dengan nilai RMSE 0,292 dan nilai R² 0,9900. Hasil tersebut dikatakan paling baik karena nilai RMSE pada *epochs* 100 paling mendekati angka 0 dimana semakin mendekati angka 0 maka tingkat *error* pada program makin kecil, dan juga nilai R² mendekati angka 1 dimana semakin mendekati angka 1 maka tingkat akurasi program makin baik. Selain itu, setelah percobaan yang dilakukan sebanyak 6 kali menggunakan *epochs* yang berbeda terlihat perbedaan



Gambar 11: Visualisasi Grafik *Training* dan *Validation Loss* Pada *Epochs* 25



Gambar 14: Visualisasi Grafik *Training* dan *Validation Loss* Pada *Epochs* 150



Gambar 12: Visualisasi Grafik *Training* dan *Validation Loss* Pada *Epochs* 50

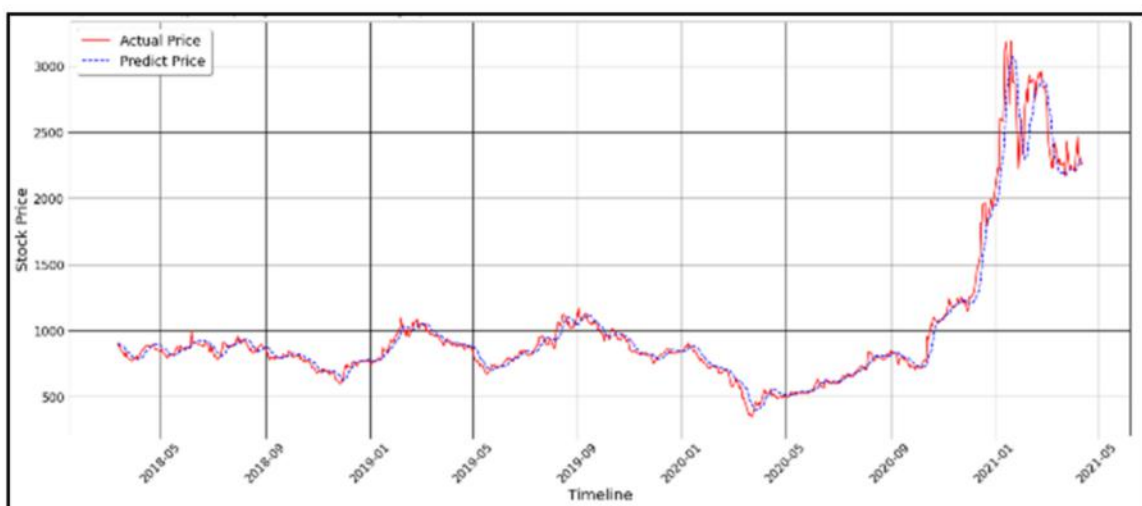


Gambar 13: Visualisasi Grafik *Training* dan *Validation Loss* Pada *Epochs* 100

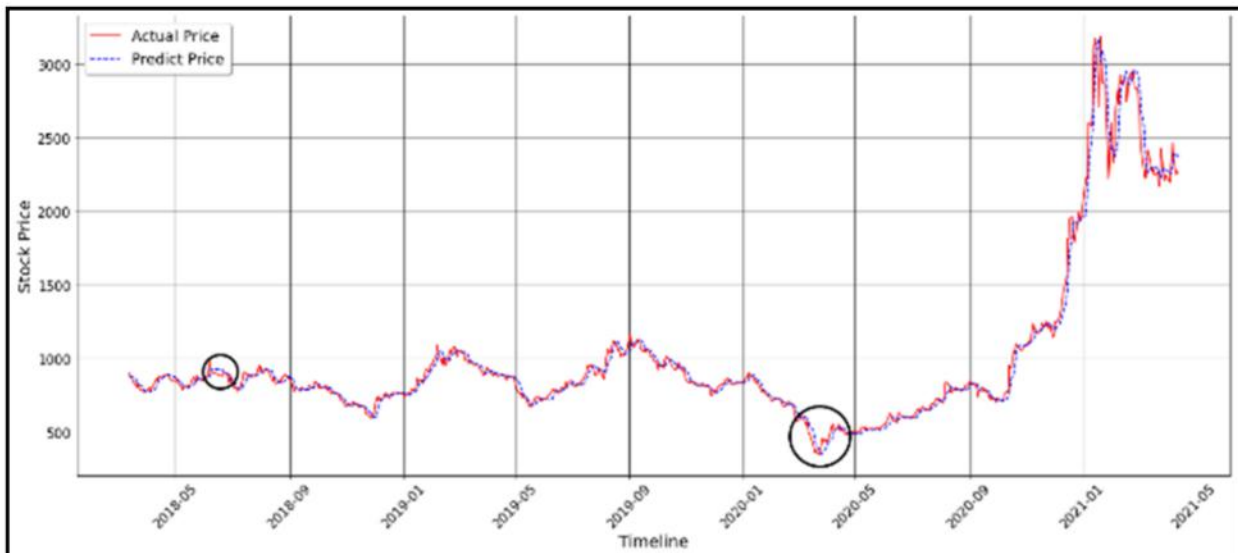
Berdasarkan dari visualisasi grafik hasil prediksi yang terdapat pada gambar 9 hingga 14 dapat disimpulkan bahwa visualisasi grafik terbaik terdapat pada percobaan menggunakan *epochs* 100 dimana tidak ada perbedaan yang signifikan antara garis hasil prediksi dan data aktual. Selain itu, pada percobaan tersebut nilai RMSE atau nilai errornya dan nilai R2 atau nilai akurasi sudah bagus.

Gambar 15 hingga 20 akan menampilkan hasil visualisasi data prediksi dan data aktual dalam bentuk grafik atau plot, dimana garis berwarna merah merupakan data aktual dan garis berwarna biru putus – putus merupakan data hasil prediksi. Grafik – grafik tersebut memiliki garis x dan y, dimana x merupakan *timeline* atau tanggal prediksi dan y merupakan *stock price* atau harga saham. Pada Gambar 15 menampilkan visualisasi data aktual dan prediksi pada saat pelatihan data menggunakan *epochs* 10, dapat dilihat bahwa pada gambar tersebut terdapat banyak garis yang tidak sama dengan garis dari data sebenarnya atau data aktual yang telah ada.

Pada Gambar 16 menampilkan gambar visualisasi hasil prediksi menggunakan *epochs* 15, dimana pada gambar tersebut dapat dilihat ada 3 lingkaran merah yang menandakan hasil dari prediksi dan data aktual ada yang berbeda sekitar di tahun 2018 bulan 12, di tahun 2020 bulan 3, dan di tahun 2021 bulan 3.



Gambar 15: Visualisasi Grafik Hasil Prediksi dengan *Epochs* 10



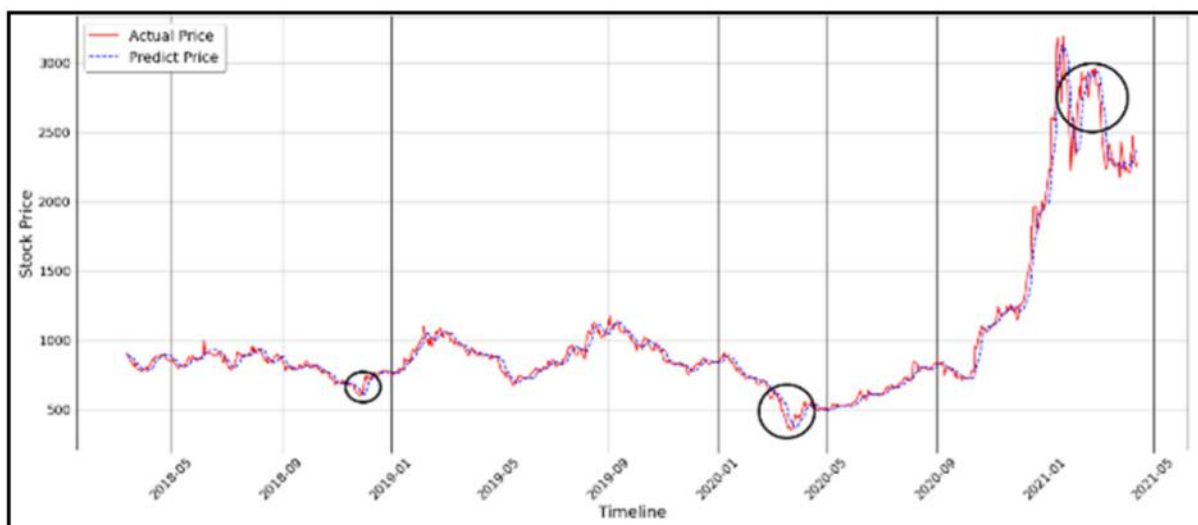
Gambar 16: Visualisasi Grafik Hasil Prediksi dengan *Epochs* 15

Gambar 17 menampilkan hasil visualisasi dari hasil prediksi menggunakan *epochs* 25, dimana pada gambar tersebut memiliki garis pada hasil prediksi yang berbeda dengan garis data aktual yang ditandai dengan lingkaran merah. Perbedaan garis tersebut yaitu terlihat sekitar tahun 2020 bulan 4.

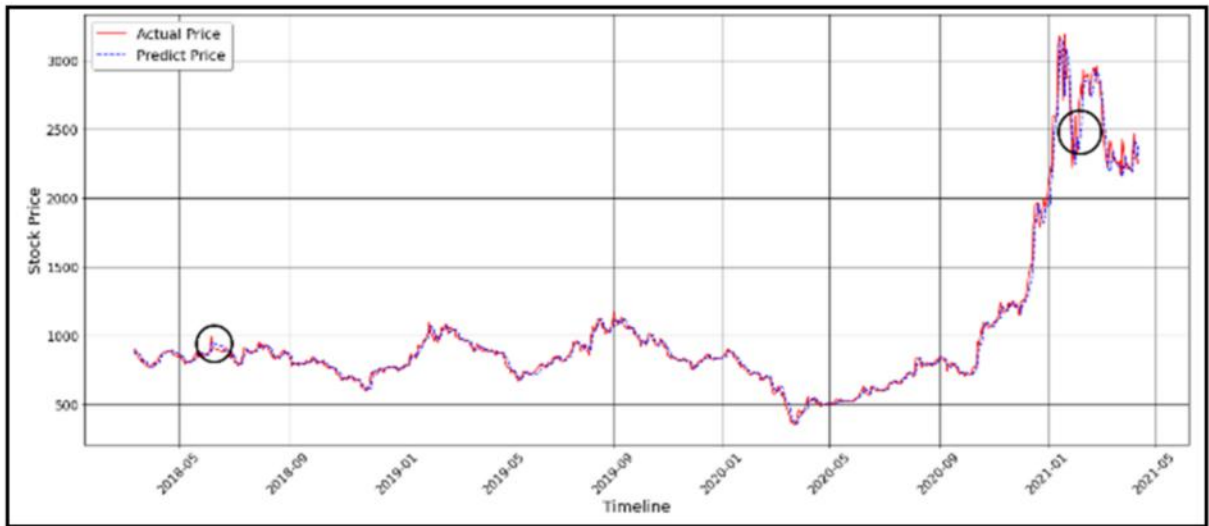
Pada Gambar 18 terdapat visualisasi grafik dari hasil prediksi menggunakan *epochs* 50 yang memiliki 2 garis hasil prediksi yang berbeda dengan data aktual. Terlihat pada lingkaran merah yang menandakan garis berbeda tersebut dimana terletak sekitar tahun 2021 bulan 2.

Pada Gambar 19 visualisasi grafik hasil prediksi menggunakan *epochs* 100, dimana pada grafik dapat dilihat garis hasil prediksi dan data aktual sudah hampir sama atau tidak memiliki perbedaan yang signifikan.

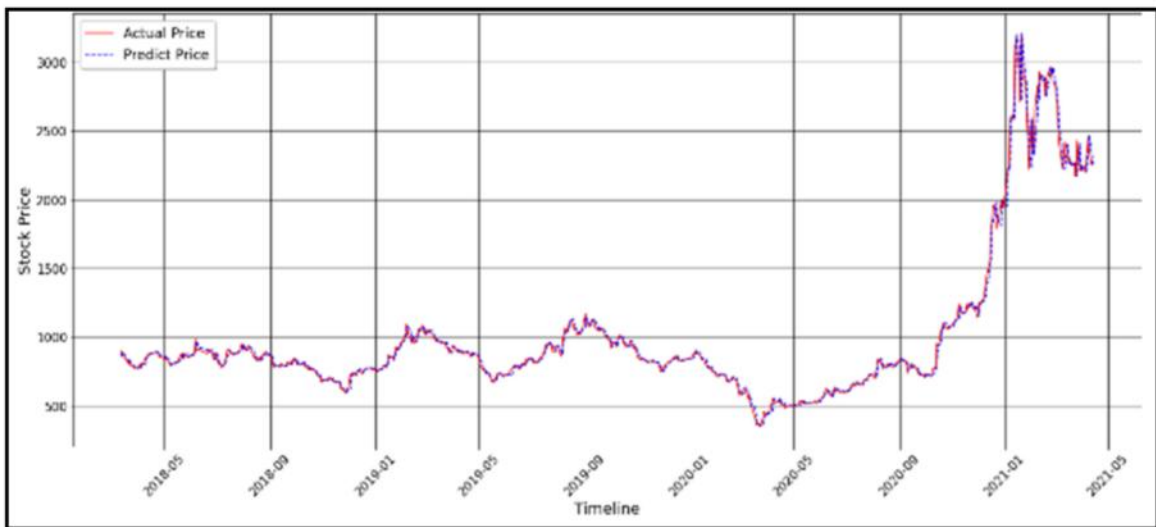
Gambar 20 merupakan gambar grafik visualisasi hasil prediksi menggunakan *epochs* 150, dimana dapat dilihat bahwa pada grafik terdapat 1 perbedaan garis yang terlihat. Garis tersebut ditandai dengan lingkaran merah yang terdapat pada sekitar tahun 2021 bulan 3.



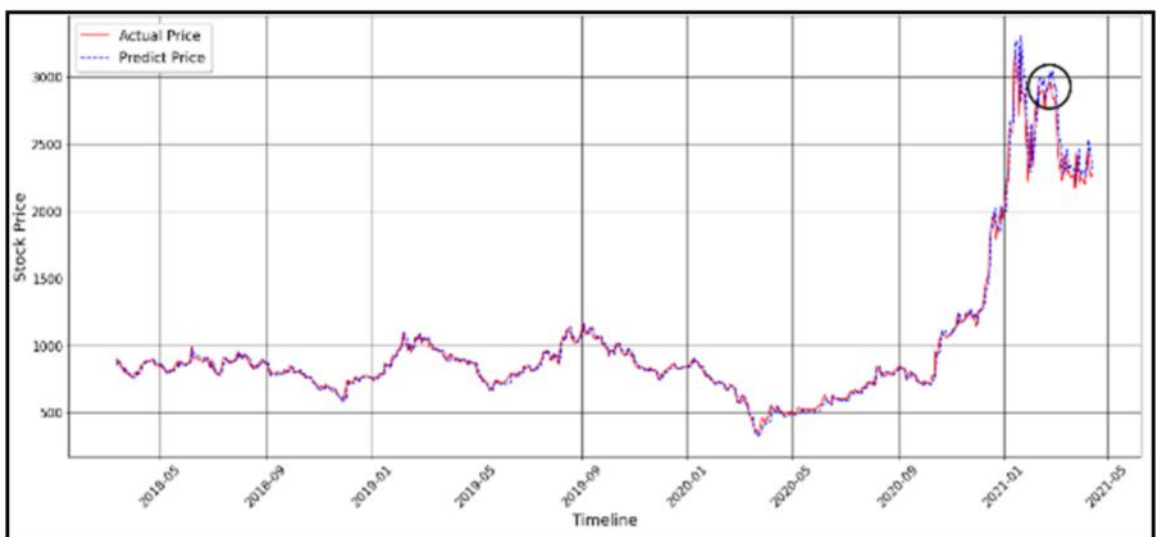
Gambar 17: Visualisasi Grafik Hasil Prediksi dengan *Epochs* 25



Gambar 18: Visualisasi Grafik Hasil Prediksi dengan *Epochs* 50



Gambar 19: Visualisasi Grafik Hasil Prediksi dengan *Epochs* 100



Gambar 20: Visualisasi Grafik Hasil Prediksi dengan *Epochs* 150

Penutup

Pada penelitian ini telah dilakukan penerapan *machine learning* menggunakan metode *long short term memory* (LSTM) untuk memprediksi harga saha pada PT Aneka Tambang Tbk (PT Antam), dimana penerapannya menggunakan *Google Colab* dan bahasa pemrograman *pyhton*. Dataset yang digunakan pada penelitian ini yaitu data harga saham PT Antam dari periode 29 September 2005 hingga 13 April 2021. Pada penelitian ini terdapat tahapan yaitu menganalisa kebutuhan, mengumpulkan dataset, melakukan *preprocessing*, olah data LSTM, menghitung nilai error dan akurasi serta visualisasi hasil prediksi.

Pada *preprocessing* data dilakukan *cleaning data*, *scaling data* dengan *Min-Max Scaling*, dan membagi data menjadi 2 yaitu 80% untuk data train dan 20% untuk data test. Model LSTM yang digunakan yaitu 1 layer dengan neuron 64 dan dimensi (1,1), 1 dropout dengan nilai 0.1, 1 output neuron dengan nilai 1, *batch size* 32. Penelitian ini telah melakukan 6 kali percobaan dengan menggunakan 6 *epochs* yang berbeda yaitu *epochs* 10, 15, 25, 50, 100, 150. Perhitungan nilai *error* pada program menggunakan *Root Mean Squared Error* (RMSE) dan perhitungan nilai akurasi menggunakan *R Squared* (R2). Pada penelitian ini visualisasi hasil prediksi menggunakan library *matplotlib* dimana akan divisualisasikan dalam bentuk grafik.

Berdasarkan percobaan yang telah dilakukan 6 kali dengan menggunakan *epochs* yang berbeda nilai RMSE dan R2 pada tiap percobaan dengan *epochs* yang berbeda tidak terlalu memiliki penurunan atau kenaikan yang signifikan. Pada penelitian ini didapatkan hasil prediksi yang baik diantara percobaan tersebut dimana hasil prediksi telah mendekati dengan nilai aktual dengan menggunakan *epochs* 100. Pada *epochs* 100 didapatkan nilai RMSE yang mendekati nilai 0 sebesar 0.3304 dan nilai R2 yang hampir mendekati nilai 1 yaitu sebesar 0.9902.

Penelitian ini masih harus dikembangkan lagi karena masih adanya kekurangan yaitu saat melakukan *train* data ulang yang menyebabkan perubahan hasil disetiap pelatihan. Penelitian selanjutnya dapat dilakukan untuk mendapatkan prediksi yang lebih baik lagi dapat menggunakan model LSTM dengan nilai neuron yang berbeda pada *layer*, jumlah *epochs* lain, dan juga *batch size*.

Daftar Pustaka

- [1] Anonim, "Profil Perusahaan Tercatat", Bursa Efek Indonesia, 2021 . Diakses 13 April

2021, dari <https://www.idx.co.id/perusahaan-tercatat/profil-perusahaan-tercatat/>

- [2] Anonim, "Company History", PT ANTAM Tbk, 2021 Diakses 16 April 2021, dari <https://www.antam.com/id/company-history>
- [3] Anonim, " Harga Saham PT. Antam ", Google Finance, 2021. Diakses pada 17 April 2021, dari <https://www.google.com/finance/quote/ANTM-IDX>
- [4] Mehryar Mohri, Afshin Rostamizadeh and Ameet Talwalkar, "Foundations of Machine Learning. Cambridge (Adaptive Computation and Machine Learning series) ", The MIT Press, ISBN-13:978-0262018258, 2012.
- [5] L. Wiranda dan M.Sadikim, "Penerapan Long Short Term Memory Pada Data Time Series Untuk Memprediksi Penjualan Produk PT. Metiska Farma", J. Nas. Pendidik. Tek. Inform., vol. 8, no. 3, pp. 184-196, 2019.
- [6] D. Cohen and W. B. Croft, "End to End Long Short Term Memory Networks for Non-Factoid Question Answering", *Proceedings of the 2016 ACM International Conference on the Theory of Information Retrieval*, pp. 143-146, (2016).
- [7] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory", *Neural Computation*, 8, 9, 1735–1780, 1997.
- [8] P. Jason Brownlee, "How to Scale Data for Long Short-Term Memory Networks in Python", *Machine Learning Mastery*, 7 Juli 2017. Diakses Juni 2021 dari <https://machinelearningmastery.com/how-to-scale-data-for-long-short-term-memory-networks-in-python/>
- [9] W. Li and Z. Liu, "A method of SVM with normalization in intrusion detection", *Procedia Environ. Sci.*, vol. 11, pp. 256– 262, 2011.
- [10] T. Chai and R.R. Draxler, "Root mean square error (RMSE) or mean absolute error (MAE)", *Geoscientific Model Development Discussions*, 7(1), 1525-1534, 2014.
- [11] D. N. A. Janie, "Statistik Deskriptif & Regresi Linier Berganda dengan SPSS", Semarang University Press, ISBN: 978-602-9019-98-8, April 2012.