

Prediksi Harga Emas Menggunakan Metode *Time Series Long Short - Term Memory Neural Network*

Fatimah Defina Setiti Alhamdani^{*1}, Gita Indah Marthasari², Christian Sri Kusuma Aditya³

^{1,2,3}Universitas Muhammadiyah Malang

defina.a19@webmail.umm.ac.id^{*1}, gita@umm.ac.id², christianskaditya@umm.ac.id³

Abstrak

Emas merupakan salah satu alat investasi populer dikalangan masyarakat yang tahan akan inflasi. Namun kegiatan investasi emas memiliki resiko berjenis data time series. Sehingga masyarakat perlu memiliki ilmu sebagai pegangan saat melakukan kegiatan investasi emas yaitu dengan memprediksi harga emas di masa depan untuk meminimalisasi resiko. Long Short-Term Memory merupakan turunan dari metode RNN yang dapat digunakan dalam memprediksi pada data time series. Penelitian ini bertujuan untuk memprediksi harga emas dari data time series per 1 hari yang telah dikumpulkan dari website harga-emas.org untuk mengetahui nilai error prediksi menggunakan metode LSTM. Analisis parameter yang dilakukan pada penelitian ini adalah jumlah neuron hidden, learning rate, dan epoch. Kombinasi parameter terbaik yang dihasilkan pada penelitian ini adalah 16 neuron hidden, learning rate 0.01, dan 100 epoch. Nilai terbaik yang dihasilkan pada penelitian ini adalah RMSE 9139,14318 dan MAPE 0,69794%. Perhitungan error MAPE terbaik pada penelitian ini dengan penelitian "Prediksi Harga Emas Menggunakan Feed Forward Neural Network dengan Metode Extreme Learning Machine" yang menghasilkan MAPE terbaik 0,8065%. Hasil tersebut menunjukkan bahwa error MAPE pada penelitian ini lebih baik daripada penelitian tersebut dan model yang terbentuk dapat dikatakan sangat bagus karena nilai MAPE terbaik yang dihasilkan dibawah 10%.

Kata Kunci: Prediksi, Emas, Long Short-Term Memory Neural Network

Abstract

Gold is one of the popular investment tools among people who are resistant to inflation. However, gold investment activities have time series data type risks. So that people need to have knowledge as a guide when carrying out gold investment activities, namely by predicting the future price of gold to minimize risk. Long Short-Term Memory is a derivative of the RNN method which can be used in predicting time series data. This study aims to predict the price of gold from the 1-day time series data that has been collected from the harga-emas.org website to determine the value of prediction errors using the LSTM method. The parameter analysis carried out in this study was the number of hidden neurons, the learning rate, and the epoch. The best combination of parameters produced in this study were 16 hidden neurons, a learning rate of 0.01, and 100 epochs. The best values produced in this study were RMSE 9139.14318 and MAPE 0,69794%. The best MAPE error calculation in this study is the research "Gold Price Prediction Using Feed Forward Neural Network with the Extreme Learning Machine Method" which produces the best MAPE 0,8065%. These results indicate that the MAPE error in this study is better than that research and the model formed can be said to be very good because the best MAPE value generated is below 10%.

Keywords: Prediction, Gold, Long Short-Term Memory Neural Network

1. Pendahuluan

Kebutuhan hidup manusia akan semakin meningkat. Adanya inflasi yang terus terjadi akan mengakibatkan harga – harga barang dan jasa mengalami kenaikan serta timbul kekhawatiran di kalangan masyarakat akan pendapatan riil mereka yang akan terus menurun [1], [2]. Kekhawatiran masyarakat tersebut meliputi tidak cukupnya pendapatan yang didapatkan untuk membiayai hidup mereka baik sekarang maupun di masa depan. Sehingga untuk mendapatkan pendapatan tambahan, masyarakat banyak yang menyimpan tabungan masa depan mereka bukan dalam bentuk tabungan uang biasa karena pergerusan nilai mata uang akibat inflasi yang merugikan. Salah satu opsi agar tabungan masyarakat tidak tergerus inflasi dan malah menjadi

pendapatan tambahan adalah dengan menaruh tabungannya di bidang investasi. Investasi yang dapat dilakukan oleh masyarakat memiliki berbagai macam bentuk antara lain investasi di bidang properti, tanah, saham, pemodaln usaha serta logam mulia berupa emas.

Masyarakat memiliki anggapan bahwa nilai emas akan selalu naik terhadap rupiah. Hal tersebut tidak dapat dikatakan sebagai sebuah kepastian karena investor emas juga harus menyadari bahwa kegiatan berinvestasi dengan emas tidak terlepas dari menebak naik maupun turunnya harga emas yang beredar di dunia mengingat resiko yang umum terjadi adalah adanya fluktuasi harga setiap harinya yang disebut sebagai jenis investasi *data time series* [3]. Sehingga masyarakat perlu memiliki ilmu yang dapat digunakan sebagai pegangan saat melakukan kegiatan investasi emas yaitu dengan memprediksi harga emas di masa depan untuk meminimalisasi resiko. Prediksi merupakan ramalan atau perkiraan yang berdasarkan pada sebuah metode ilmiah maupun asumsi belaka [4]. Peramalan adalah proses memperkirakan kuantitas permintaan masa depan yang mencakup dari segi ukuran kuantitas, kualitas, waktu dan lokasi untuk memenuhi permintaan barang maupun jasa (Pakaja & Naba, 2015). Model *time series* dapat digunakan untuk pendugaan masa depan menggunakan serangkaian data pengamatan yang dikumpulkan dalam jangka waktu tertentu dengan memperhatikan *error* [5]. Terdapat beberapa metode yang dapat digunakan dalam memprediksi dengan menggunakan *data time series* salah satunya adalah *Long Short - Term Memory*. Metode turunan dari RNN (*Recurrent Neural Network*) adalah LSTM yang dirancang untuk mengatasi permasalahan *vanishing* dan *exploding gradient* dalam penggunaan RNN [6]. RNN merupakan jaringan yang dapat menampung output jaringan untuk digunakan kembali sebagai input pada jaringan untuk menghasilkan output jaringan selanjutnya (Walid, 2019). Tidak seperti RNN, LSTM memiliki unit penyimpanan atau *memory cell* dan *input gates* untuk menyimpan informasi memori yang telah disimpan dalam waktu lama (Faishol et al., 2020).

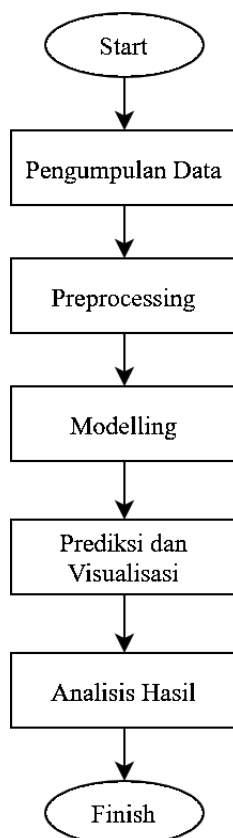
Terdapat beberapa penelitian terdahulu menggunakan metode LSTM. Pada karya Analisis dan Implementasi Jaringan Safaf Tiruan *Long Short Term Memory* untuk Prediksi Harga Bitcoin (Aldi et al., 2018) dengan menguji 70% dari parameter komposisi data latih dan 30% dari data pengujian, model LSTM dibangun untuk memprediksi harga Bitcoin, 1 parameter mode time series, 25 neuron hidden, dan maksimum epoch 100, sehingga keakuratan rerata data latih adalah 95.36% dan akurasi rata – rata data uji 93.5%. Pada karya Deteksi Emosi Pada Teks Menggunakan *Nested LSTM* (Haryadi & Kusuma, 2019) dilakukan perbandingan terhadap metode SVM, LSTM, dan *Nested LSTM* dalam mengklasifikasikan 7 jenis emosi seperti anger (kemarahan), fear (ketakutan), joy (kegembiraan), love (cinta), sadness (kesedihan), surprise (kejutan), dan thankfulness (rasa syukur). Metode *Nested LSTM* menghasilkan tingkat akurasi paling tinggi untuk mendeteksi multi-kelas emosi yaitu 99,167% dan metode LSTM tidak berbeda jauh yaitu 99,154%. Namun LSTM memiliki performa yang lebih baik pada precision, recall, dan F1-score. Pada karya Perbandingan ARIMA dan LSTM dalam Peramalan Time Series studi empiris yang dilakukan menunjukkan bahwa algoritma berbasis deep learning seperti LSTM lebih baik daripada algoritma berbasis algoritma tradisional, seperti model ARIMA. Khususnya pada rata-rata penurunan error rates yang diperoleh LSTM berada diantara 84% sampai 87% sehingga dapat membuktikan keunggulan LSTM terhadap ARIMA. Pada karya Prediksi Harga Emas Menggunakan Jaringan Saraf Tiruan *Feed Forward* dan *Extreme Learning Machine* (Izati et al., 2019) menunjukkan nilai error MAPE terbaik yang diperoleh dalam penelitian adalah 0,8065% dan dapat dikatakan sangat baik sebab dibawah 10%.

Penelitian ini menggunakan metode LSTM untuk memprediksi harga emas. Dengan melakukan perhitungan yang baik dan benar setidaknya dapat membantu meningkatkan probabilitas keakuratan harga emas di masa depan dalam meminimalisasi resiko yang mungkin terjadi pada saat investasi.

2. Metode Penelitian

Penelitian ini menggunakan dua penelitian terdahulu yang dijadikan sebagai acuan. Penelitian terdahulu yang digunakan sebagai rujukan dalam membangun pemodelan LSTM adalah “Analisis dan Implementasi Jaringan Saraf Tiruan *Long Short Term Memory* untuk Prediksi Harga Bitcoin”. Beberapa aspek yang digunakan sebagai rujukan pada penelitian ini adalah komposisi pembagian data dengan hasil terbaik pada penelitian tersebut, penggunaan min-max scalling, optimizer Adam, serta alur skenario pemodelan dalam menemukan kombinasi parameter yang optimal untuk penelitian ini. Sementara penelitian terdahulu kedua digunakan

sebagai pembandingan hasil error MAPE yang dihasilkan dengan penelitian ini pada sub-bab 2.5. Analisis Hasil. Tahapan penelitian ini tertera pada Gambar 1.



Gambar 1. Flowchart tahapan penelitian

2.1 Pengumpulan Data

Data yang digunakan untuk penelitian ini dikumpulkan dari *website* harga-emas.org berupa data *time series* harga emas per 1 hari mulai 1 Agustus 2017 sampai 30 Maret 2021. Data yang didapat menggunakan satuan IDR per 1 gram. Data yang telah dikumpulkan disimpan dalam format *Comma Separated Values* (CSV). Jumlah data yang digunakan sebanyak 1339 sebagai acuan penelitian dalam menganalisa. Sampel dataset tertera pada Tabel 1.

Tabel 1. Sampel dataset yang digunakan

Date	IDR/gram
8/1/2020	930.758
8/2/2020	930.758
8/3/2020	931.589
8/4/2020	943.220
8/5/2020	955.805
8/6/2020	965.726
8/7/2020	954.125
8/8/2020	958.405
8/9/2020	958.405
8/10/2020	964.253
8/11/2020	920.871
8/12/2020	925.797
8/13/2020	934.047
8/14/2020	928.449
8/15/2020	932.458
8/16/2020	932.458

8/17/2020	953.268
8/18/2020	962.507
8/19/2020	932.426
8/20/2020	923.637
8/21/2020	920.171
8/22/2020	922.667
8/23/2020	922.667
8/24/2020	918.134
8/25/2020	901.574
8/26/2020	915.264
8/27/2020	908.717
8/28/2020	930.613
8/29/2020	928.708
8/30/2020	928.708
8/31/2020	921.937

2.2 Preprocessing

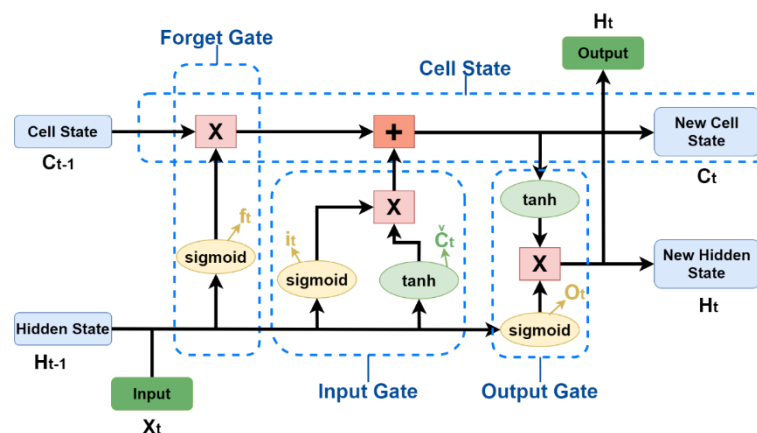
Langkah pertama yang dilakukan pada tahap preprocessing ini adalah normalisasi data. Normalisasi memiliki tujuan untuk menghilangkan redudansi data agar dapat diproses secara optimal. Normalisasi pada dataset penelitian ini dilakukan dengan mengubah nilai data *real* menjadi nilai yang berada diantara 0 dan 1 menggunakan proses normalisasi *min-max scaling*. Rumus yang digunakan dalam *min-max scaling* tertera pada Persamaan 1.

$$x_i' = \frac{x_i - \min(x)}{\max(x) - \min(x)} \quad (1)$$

Setelah melewati tahap normalisasi, dataset tersebut dibagi menjadi data *train* dan data *test*. Setelah dibagi, masing-masing dari data *train* dan data *test* akan dikonversikan *array* nilai menjadi dataset *matrix*. Setelah itu dilakukan *reshape* pada *input* menjadi [*samples, time steps, feature*] sesuai dengan yang diperlukan metode LSTM pada data *train* dan data *test* yang telah dibentuk.

2.3 Modeling

Long Short - Term Memory (LSTM) memiliki suatu *memory cell* untuk menyimpan data dalam waktu yang lama. Pada setiap *time steps*, sel LSTM mengambil 3 bagian informasi yaitu data *input* saat ini, memori jangka pendek (*hidden state*) dari sel sebelumnya dan memori jangka panjang (*cell state*). Sel kemudian menggunakan *gate* untuk mengatur informasi yang akan disimpan atau dikeluarkan pada setiap *time step* sebelum meneruskan informasi *cell state* dan *hidden state* ke sel berikutnya. Sehingga *gate* berperan penting dalam menghilangkan informasi yang tidak relevan dan hanya berpegang pada informasi yang berguna. Pada Gambar 2 terlihat *gate* yang dimaksud adalah *forget gate*, *input gate*, dan *output gate*.



Gambar 2 Arsitektur Long Short - Term Memory

Forget gate menentukan informasi dari *cell state* yang akan disimpan atau dikeluarkan dengan mengalikan *cell state* yang akan masuk dan *forget vector* yang dihasilkan oleh *input* saat ini dan *hidden state* yang akan masuk. Untuk mendapatkan *forget vector*, *hidden state*, dan *input* saat ini dimasukkan dalam fungsi aktivasi *sigmoid*, seperti pada *layer* pertama di *input gate*, namun dengan bobot yang berbeda. Rumus pada *forget gate* dapat dilihat pada Persamaan 2.

$$f_t = \sigma(W_f \cdot (H_{t-1}, x_t) + bias_f) \quad (2)$$

Input gate menentukan informasi baru mana yang akan disimpan pada *long-term memory*. Pada *gate* ini terdapat dua *layer*. Pada Persamaan 3, *Layer* pertama digunakan untuk memperbarui informasi yang relevan menggunakan fungsi aktivasi *sigmoid*. Pada *layer* kedua menggunakan fungsi aktivasi *tanh* dalam menentukan vektor nilai yang baru untuk disimpan pada *memory cell* dapat dilihat pada Persamaan 4. Rumus pada *input gate* sebagai berikut.

$$i_t = \sigma(W_i \cdot (H_{t-1}, x_t) + bias_i) \quad (3)$$

$$\check{C}_t = \tanh(W_c \cdot (H_{t-1}, x_t) + bias_c) \quad (4)$$

Output dari *input gate* dan *forget gate* akan mengalami penambahan *pointwise* untuk membentuk *cell state* versi baru yang akan diteruskan ke sel berikutnya. *Cell state* baru ini akan digunakan pada *gate* terakhir yaitu *output gate*. Penentuan *cell state* baru dapat dilihat pada Persamaan 5.

$$C_t = C_{t-1} * f_t + i_t * \check{C}_t \quad (5)$$

Output gate akan mengambil *input* saat ini, *hidden state* sebelumnya, dan *cell state* yang baru dihitung untuk menghasilkan *hidden state* baru yang akan diteruskan ke *time step* berikutnya. Pada *output gate* ini digunakan dua *layer*. *Layer* pertama menggunakan fungsi aktivasi *sigmoid* untuk menentukan informasi pada *memory cell* yang akan dikeluarkan seperti pada Persamaan 6. Kemudian akan dimasukkan nilai tersebut pada *memory cell* menggunakan fungsi aktivasi *tanh* seperti pada Persamaan 7. Rumus *output gate* sebagai berikut.

$$O_t = \sigma(W_o \cdot (H_{t-1}, x_t) + bias_o) \quad (6)$$

$$H_t = O_t * \tanh(C_t) \quad (7)$$

Pada tahap awal pembuatan model digunakan *sequential model*. Tahap *modeling* ini akan dilakukan tiga skenario. Skenario tersebut antara lain:

- Skenario 1: *Neuron hidden* yang akan dijalankan adalah 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, dan 25 dengan *learning rate* 0.1 dan *epoch* 100.
- Skenario 2: Hasil terbaik dari *neuron hidden* pada Skenario 1 digunakan pada Skenario 2 dengan *learning rate* 0.1, 0.01, 0.001, 0.0001, 0.00001 dan *epoch* 100.
- Skenario 3: Mengambil kombinasi dengan akurasi terbaik pada *neuron hidden* dan *learning rate* pada Skenario 2, *epoch* yang digunakan 50, 100, 250, 500, 1000.

Salah satu permasalahan yang terjadi ketika proses *training* pada *Neural Network* adalah *overfitting*. *Overfitting* dapat diatasi dengan menambahkan *dropout layer*. Hal ini dikarenakan *dropout layer* dapat meningkatkan generalisasi data pada model. Proses training dioptimalkan dengan menggunakan optimasi Adam serta melakukan *update* bobot dan bias pada sistem dalam setiap *epoch* yang dijalankan. Proses tersebut akan terus berulang sebanyak *epoch* yang telah ditentukan,

Penelitian ini menggunakan dua perhitungan *error* untuk mengetahui ketepatan prediksi yaitu *Mean Absolute Percentage Error* (MAPE) dan *Root-Mean-Square Error* (RMSE). Rumusan dari RMSE dan MAPE dapat dilihat pada Persamaan 8 dan Persamaan 9.

$$MAPE = \frac{1}{n} \sum \frac{|x_i - x'_i|}{x_i} \times 100 \quad (8)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - x'_i)^2} \quad (9)$$

Dari ketiga skenario tersebut akan dilihat nilai RMSE dan MAPE yang dihasilkan untuk mengambil keputusan dalam memilih model yang akan digunakan sebagai *final model*. Semakin kecil nilai kesalahan RMSE dan MAPE yang dihasilkan pada model yang diimplementasikan maka semakin dekat nilai prediksi dengan data *real*.

2.4 Prediksi dan Visualisasi

Setelah mendapatkan kombinasi terbaik dari ketiga skenario yang telah dijalankan di tahap *training*, dibentuklah *final model* untuk melakukan prediksi. Prediksi akan divisualisasikan dalam bentuk *plot* pada data *real*, hasil prediksi *train* dan hasil prediksi *test* dengan menggunakan tiga warna yang berbeda. Kemudian dibentuk *plot* untuk menampilkan hasil peramalan untuk 300 hari mendatang.

2.5 Analisis Hasil

Pada tahapan ini dilakukan analisis dengan membandingkan hasil perhitungan *error* pada penelitian ini dengan penelitian yang menggunakan topik harga emas dan pemodelan *neural network*. Perhitungan *error* MAPE pada penelitian ini akan dibandingkan dengan penelitian "Prediksi Harga Emas Menggunakan *Feed Forward Neural Network* dengan *Metode Extreme Learning Machine*".

3. Hasil Penelitian dan Pembahasan

Data *time series* merupakan data historis yang dikumpulkan dalam kurun waktu tertentu secara berurutan. Data *time series* pada penelitian ini digunakan sebagai acuan dalam menganalisa hasil dan memprediksi nilai masa depan. Pada tahap implementasi ini terdapat tiga tahapan, yaitu:

3.1 Preprocessing

Pada tahap ini dilakukan normalisasi terhadap data yang telah dikumpulkan. Hasil dari normalisasi data tersebut disimpan bersamaan dengan data asli sebelum dinormalisasi pada *dataframe* dengan nama *Normalization_df* seperti pada Gambar 3. Komposisi pembagian data *train* dan *test* yang digunakan pada penelitian ini adalah 70% *train* : 30% *test*. Sehingga terbentuk 937 data *train* dan 402 data *test*.

	Date	IDR/gram	IDR/gram norm
0	8/1/2017	544868	0.016544
1	8/2/2017	545151	0.017206
2	8/3/2017	543649	0.013696
3	8/4/2017	538713	0.002162
4	8/5/2017	539236	0.003384
...
1334	3/27/2021	804719	0.623761
1335	3/28/2021	804719	0.623761
1336	3/29/2021	795944	0.603256
1337	3/30/2021	784587	0.576717
1338	3/31/2021	801947	0.617283

1339 rows × 3 columns

Gambar 3. *Normalization_df*

3.2 Modeling

Skenario pengujian pada penelitian ini meliputi pengujian jumlah *neuron hidden*, *learning rate*, dan *epoch*. Optimasi yang digunakan dalam proses pelatihan model penelitian ini adalah Adam (*Adaptive Moment Estimation*) dan dioptimalkan dengan *Mean Squared Error* (MSE). Lalu dihitung nilai RMSE dan MAPE yang dihasilkan pada setiap skenario yang dijalankan.

3.2.1 Skenario 1

Skenario 1 mengkombinasikan parameter awal *learning rate* 0.1 dan *epoch* 100. Skenario ini mengubah nilai *neuron hidden* pada setiap pengujiannya. Hasil dari pelaksanaan skenario 1 tertera pada Tabel 2.

Tabel 2. Hasil Skenario 1

<i>Neuron Hidden</i>	<i>Learning Rate</i>	<i>Epoch</i>	RMSE	MAPE
10	0,1	100	19245196,20220	238,82304
11	0,1	100	57665,76378	3,74616
12	0,1	100	25515995,40140	862,34994
13	0,1	100	203719596,87404	4636,57577
14	0,1	100	344288,61546	15,29282
15	0,1	100	3740848,43626	75,71102
16	0,1	100	12848,87216	1,12627
17	0,1	100	17669,64935	1,76788
18	0,1	100	124659,53323	11,11462
19	0,1	100	42651,90285	4,81773
20	0,1	100	16407,27571	1,40881
21	0,1	100	1127792,88944	48,69348
22	0,1	100	37535,38679	4,01350
23	0,1	100	140873,63212	9,43260
24	0,1	100	37960,46094	4,14089
25	0,1	100	16744,39165	1,64830

3.2.2 Skenario 2

Skenario 2 dilakukan dengan menggunakan kombinasi parameter yang menghasilkan RMSE dan MAPE terbaik pada skenario 1 yaitu 16 *neuron hidden*, *learning rate* 0.1, dan 100 *epoch*. Kemudian pada skenario 2 dilakukan pengujian dengan mengubah nilai *learning rate*. Hasil dari pelaksanaan skenario 2 tertera pada Tabel 3.

Tabel 3. Hasil Skenario 2

<i>Neuron Hidden</i>	<i>Learning Rate</i>	<i>Epoch</i>	RMSE	MAPE
16	0,1	100	12848,87216	1,12627
16	0,01	100	9139,14318	0,69794
16	0,001	100	24386,47118	2,52411
16	0,0001	100	38534,09335	3,75439
16	0,00001	100	162391,35700	18,57427

3.2.3 Skenario 3

Skenario 3 dilakukan dengan menggunakan kombinasi parameter yang menghasilkan RMSE dan MAPE terbaik pada skenario 2 yaitu 16 *neuron hidden*, *learning rate* 0.01, dan 100 *epoch*. Kemudian pada skenario 2 dilakukan pengujian dengan mengubah nilai *learning rate*. Hasil dari pelaksanaan skenario 3 tertera pada Tabel 4. Hasil terbaik yang diperoleh pada skenario 3 ini yaitu kombinasi parameter 16 *neuron hidden*, *learning rate* 0.01 dan 100 *epoch*. Nilai *error* yang dihasilkan pada kombinasi parameter tersebut adalah RMSE 9139,14318 dan MAPE 0,69794%. Sehingga kombinasi parameter tersebut yang digunakan sebagai final model pada penelitian ini untuk melakukan peramalan nilai masa depan.

Tabel 4. Hasil Skenario 3

<i>Neuron Hidden</i>	<i>Learning Rate</i>	<i>Epoch</i>	RMSE	MAPE
16	0,01	50	9661,20390	0,79044
16	0,01	100	9139,14318	0,69794
16	0,01	250	2100,94650	1,79578
16	0,01	500	238956,42032	14,22833
16	0,01	1000	207959,92517	20,11776

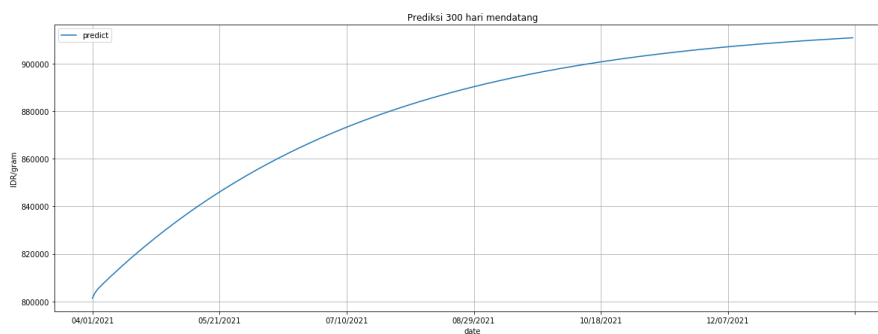
3.3 Prediksi dan Visualisasi

Pada tahap ini dibentuk fungsi untuk meramalkan nilai masa depan berdasarkan model dengan kombinasi terbaik 16 *neuron hidden*, 0.01 *learning rate*, dan 100 *epoch* menghasilkan nilai MAPE terbaik sebesar 0,69794%. Visualisasi dari data *real*, hasil prediksi *train*, dan hasil prediksi *test* tertera pada Gambar 4.



Gambar 4. Visualisasi data real, prediksi train, dan prediksi test

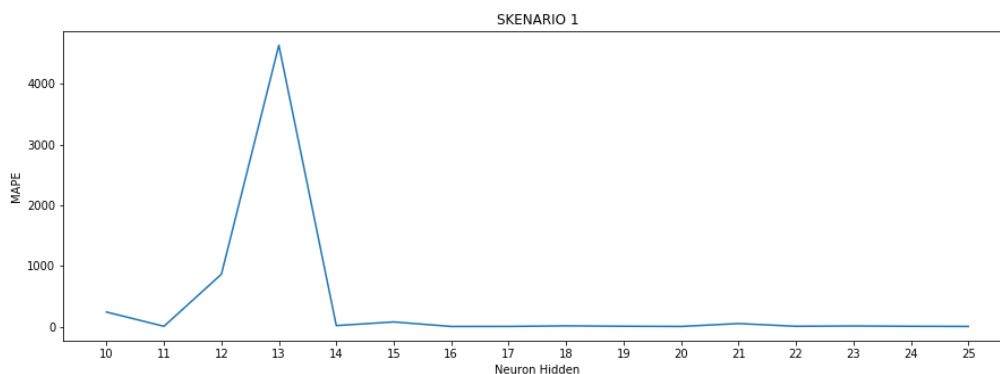
Setelah dilakukan visualisasi data *real*, hasil prediksi *train*, dan hasil prediksi data *test* dilakukan peramalan nilai masa depan untuk 300 hari mendatang. Visualisasi dari hasil peramalan tersebut tertera pada Gambar 5.



Gambar 5. Visualisasi hasil peramalan 300 hari mendatang

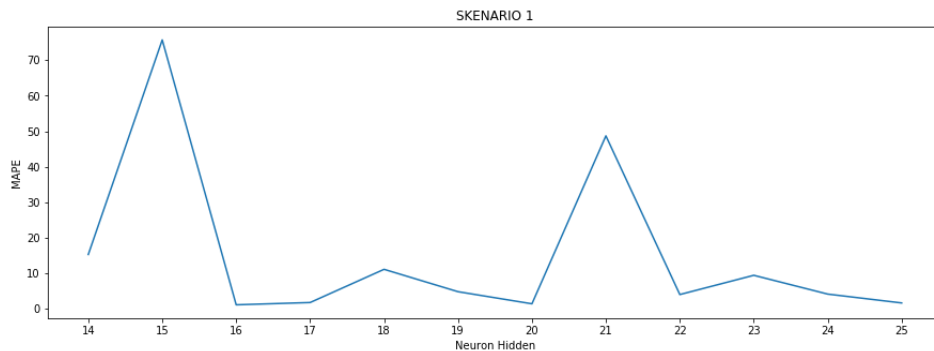
3.4 Analisis Hasil

Berdasarkan pengujian ini, menghasilkan grafik untuk masing-masing skenario yang telah dilakukan. Skenario 1 melakukan pengujian yang mengubah *neuron hidden* dengan menggunakan *learning rate* 0.01 dan *epoch* 100. Pada Gambar 6 menunjukkan bahwa semakin besar nilai *neuron hidden* tidak selalu menghasilkan nilai MAPE yang lebih kecil. Nilai MAPE terkecil yang dihasilkan berada pada *neuron hidden* 16.



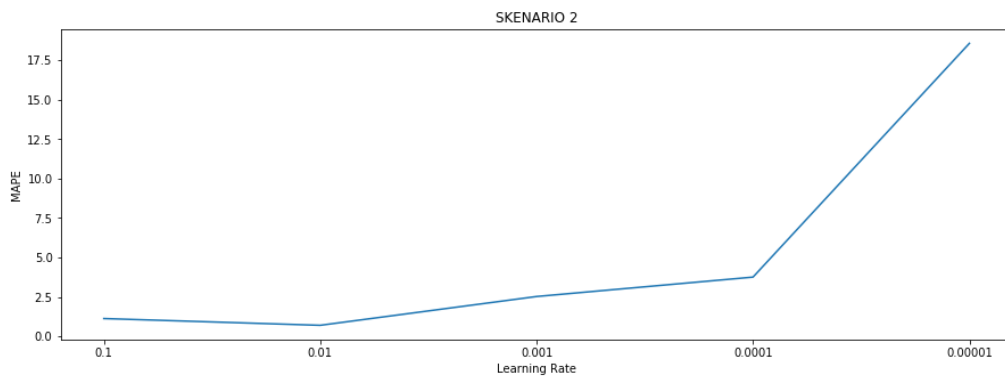
Gambar 6. Grafik skenario 1 penentuan neuron hidden

Pada Gambar 6 tidak dapat terlihat dengan jelas pergerakan perubahan pada *neuron hidden* 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, dan 25. Sehingga ditampilkan Gambar 7 untuk memperjelas pergerakan pada *neuron hidden* tersebut.



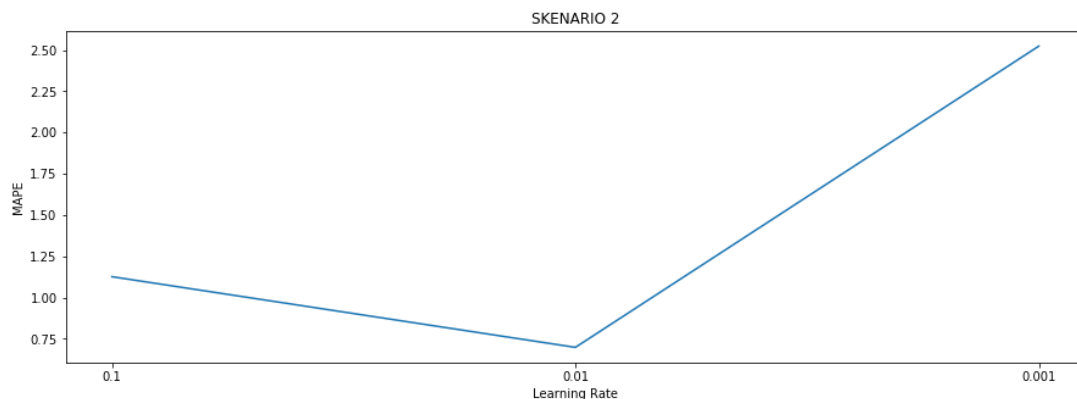
Gambar 7. Grafik skenario 1 pada *neuron hidden* 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, dan 25

Skenario 2 dilakukan perubahan nilai *learning rate* pada model dengan menggunakan 16 *neuron hidden* yang diperoleh dari skenario 1 dan *epoch* 100. Nilai MAPE terbaik pada skenario 2 tersebut adalah pada *learning rate* 0.01. Pada Gambar 8 dapat dapat diambil keputusan untuk menggunakan *learning rate* 0.01 dalam menjalankan skenario 3. Hal tersebut dikarenakan pada *learning rate* 0.01 menghasilkan nilai MAPE terbaik.



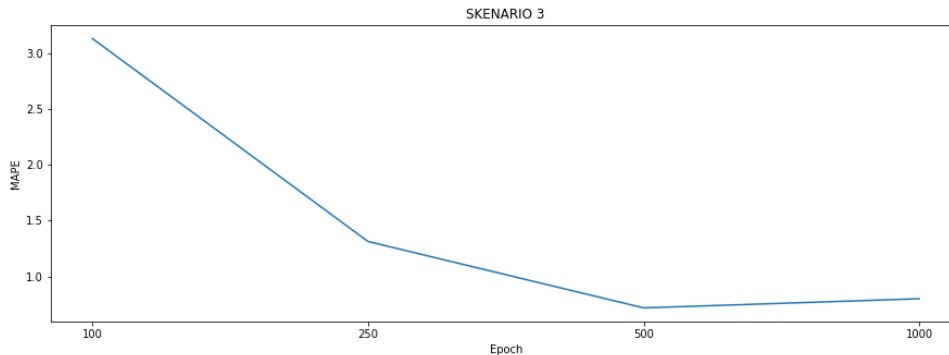
Gambar 8. Grafik skenario 2 penentuan *learning rate*

Pada Gambar 8 tidak dapat terlihat dengan jelas pergerakan perubahan antara *learning rate* 0,1, 0,01, dan 0,001. Sehingga ditampilkan Gambar 9 untuk memperjelas pergerakan pada *learning rate* tersebut.



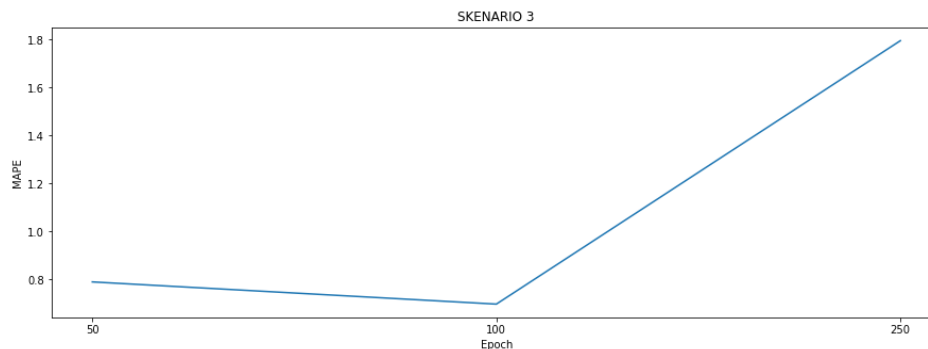
Gambar 8. Grafik skenario 2 pada *learning rate* 0,1, 0,01, dan 0,001

Skenario 3 melakukan pengubahan jumlah *epoch* dengan kombinasi parameter 16 *neuron hidden* dan *learning rate* 0.01 yang diperoleh dari skenario 2. Pada Gambar 9 menunjukkan nilai MAPE yang dihasilkan pada *epoch* 50, 250, 500, dan 1000 semakin besar namun pada *epoch* 100 mengalami penurunan. Pada skenario 3 menghasilkan MAPE terbaik pada *epoch* 100.



Gambar 9. Grafik skenario 3 penentuan epoch

Pada Gambar 9 tidak dapat terlihat dengan jelas pergerakan perubahan antara *epoch* 50, 100, dan 250. Sehingga ditampilkan Gambar 10 untuk memperjelas pergerakan pada *epoch* tersebut.



Gambar 10. Grafik skenario 3 pada epoch 50, 100, dan 250

Kombinasi parameter yang menghasilkan nilai RMSE dan MAPE terbaik yaitu 16 *neuron hidden*, *learning rate* 0.01, dan 100 *epoch*. Lalu akan dilakukan perbandingan nilai *error* MAPE terbaik yang dihasilkan pada penelitian ini dengan penelitian milik Nisa Afida berjudul "Prediksi Harga Emas Menggunakan Jaringan Saraf Tiruan *Feed Forward* dengan Metode *Extreme Learning Machine*". Pada Tabel 5 memperlihatkan perbandingan komponen yang digunakan pada penelitian pembandingan dengan penelitian ini saat pemodelan dalam menentukan komposisi yang tepat dalam mengoptimalkan peramalan.

Tabel 5. Perbandingan Komponen dalam Pemodelan

Komponen	Penelitian Pembandingan	Penelitian ini
Jumlah data	1319	1339
Pembagian Data	80% <i>train</i> : 20% <i>test</i>	70% <i>train</i> : 30% <i>test</i>
Fungsi aktivasi	sigmoid biner	adam
<i>Neuron hidden</i> terbaik	7	16
Nilai MAPE terbaik	0,8065%	0,69794%
Nilai RMSE terbaik	-	9139,14318

Error MAPE terbaik yang dihasilkan oleh penelitian tersebut adalah 0,8065% sementara pada penelitian ini dihasilkan 0,69794%. *Error* RMSE terbaik pada penelitian ini adalah 9139,14318 sedangkan penelitian pembandingan tidak melakukan perhitungan *error* RMSE. Hasil tersebut menunjukkan bahwa *error* MAPE pada penelitian ini lebih baik daripada penelitian tersebut. Selain itu, nilai MAPE yang dihasilkan pada penelitian ini dapat dikatakan sangat bagus karena dibawah 10%.

4. Kesimpulan

Tugas akhir ini menggunakan metode *Long Short - Term Memory Neural Network* dalam memprediksi harga emas. Pada tugas akhir ini didapatkan analisis hasil skenario 1 yang mengubah nilai *neuron hidden* dengan perhitungan *error* terbaik pada *neuron hidden* 16 menghasilkan RMSE 12848,87216 dan MAPE 1,12627%. Lalu pada analisis hasil skenario 2 yang mengubah nilai *learning rate* dengan perhitungan *error* terbaik pada RMSE 9139,14318 dan MAPE 0,69794% setelah memperkecil nilai *learning rate* menjadi 0,01 namun bila nilai tersebut diperkecil lagi maka *error* yang dihasilkan lebih besar. Pada analisis hasil skenario 3 yang mengubah jumlah *epoch* menunjukkan bahwa perhitungan *error* yang dihasilkan tidak mengalami penurunan dari skenario 2 setelah dilakukan penambahan maupun pengurangan pada jumlah *epoch*. Sehingga komposisi terbaik pada skenario 2 digunakan sebagai final model untuk meramalkan harga emas selama 300 hari mendatang. Dari ketiga skenario yang telah dijalankan ditemukan bahwa kombinasi *neuron hidden*, *learning rate*, dan *epoch* yang optimal bukanlah nilai yang terkecil maupun terbesar pada setiap parameternya.

Error MAPE terbaik yang dihasilkan pada penelitian ini menggunakan metode LSTM memiliki nilai yang lebih baik daripada penelitian pembandingnya yang menggunakan metode *Extreme Learning Machine* dengan *error* MAPE terbaik 0,8065%. Model yang dihasilkan pada penelitian ini dapat dikatakan sangat bagus karena nilai MAPE terbaiknya dibawah 10%.

Saran dalam pelaksanaan penelitian berikutnya adalah dengan menggunakan lebih banyak fitur dalam mengoptimalkan kinerja LSTM dalam memprediksi.

Daftar Notasi

x_i	: data real ke-i
x_i'	: data ke-i yang telah dinormalisasi
n	: jumlah data
x'_i	: hasil prediksi ke-i
f_t	: <i>forget gate</i> ke-t
σ	: fungsi sigmoid
W_f	: <i>weight</i> pada <i>forget gate</i>
H_{t-1}	: <i>hidden state</i> sebelum <i>hidden state</i> ke-t
x_t	: data <i>input</i>
$bias_f$: bias pada <i>forget gate</i>
i_t	: <i>input gate</i> ke-t
W_i	: <i>weight</i> pada <i>input gate</i>
$bias_i$: bias pada <i>input gate</i>
\check{C}_t	: kandidat vektor <i>cell state</i> baru
W_c	: <i>weight</i> pada <i>cell state</i>
$bias_c$: bias pada <i>cell state</i>
C_t	: <i>cell state</i> ke-t
C_{t-1}	: <i>cell state</i> sebelum <i>cell state</i> ke-t
H_t	: <i>hidden state</i> ke-t
O_t	: <i>output gate</i> ke-t
W_o	: <i>weight</i> pada <i>output gate</i>
$bias_o$: bias pada <i>output gate</i>

Referensi

- [1] Aldi, M. W. P., Jondri, & Aditsania, A. (2018). Analisis dan Implementasi Long Short Term Memory Neural Network untuk Prediksi Harga Bitcoin. *E-Proceeding of Engineering*, 5(2), 3548–3555.
- [2] Amin, A., Sari, Y. A., & Adinugroho, S. (2019). Klon Perilaku Menggunakan Jaringan Saraf Tiruan Konvolusional Dalam Game SuperTuxKart. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer (J-PTIIK) Universitas Brawijaya*, 3(1), 866–875.
- [3] Ardiana, A., & Amak, Y. (2017). Sistem Prediksi Penentuan Jenis Tanaman Sayuran Berdasarkan Kondisi Musim dengan Pendekatan Metode Trend Moment. *Bimasakti*.
- [4] Chang, P. C., Wang, Y. W., & Liu, C. H. (2007). The development of a weighted evolving fuzzy neural network for PCB sales forecasting. *Expert Systems with Applications*, 32(1), 86–96. <https://doi.org/10.1016/j.eswa.2005.11.021>

- [5] Desmonda, D., Tursina, T., & Irwansyah, M. A. (2018). Prediksi Besaran Curah Hujan Menggunakan Metode Fuzzy Time Series. *Jurnal Sistem Dan Teknologi Informasi (JUSTIN)*, 6(4), 141. <https://doi.org/10.26418/justin.v6i4.27036>
- [6] Faishol, M. A., Endroyono, & Irfansyah, A. N. (2020). Prediksi Polusi Udara Perkotaan di Surabaya Menggunakan Recurrent Neural Network - Long Short Term Memory. *JUTI : Jurnal Ilmiah Teknologi Informasi*, 18, 102–114.
- [7] Guntur, M., Santony, J., & Yuhandri, Y. (2018). Prediksi Harga Emas dengan Menggunakan Metode Naïve Bayes dalam Investasi untuk Meminimalisasi Resiko. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 2(1), 354–360. <https://doi.org/10.29207/resti.v2i1.276>
- [8] Haryadi, D., & Kusuma, G. P. (2019). Emotion detection in text using nested Long Short-Term Memory. *International Journal of Advanced Computer Science and Applications*, 10(6), 351–357. <https://doi.org/10.14569/ijacsa.2019.0100645>
- [9] Izati, N. A., Warsito, B., & Widiharih, T. (2019). Prediksi Harga Emas Menggunakan Feed Forward Neural Network Dengan Metode Extreme Learning Machine. *Jurnal Gaussian*, 8(2), 171–183. <https://doi.org/10.14710/j.gauss.v8i2.26641>
- [10] Kingma, D. P., & Ba, J. L. (2015). Adam: A method for stochastic optimization. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 1–15.
- [11] Mahena, Y., Rusli, M., & Winarso, E. (2015). Prediksi Harga Emas Dunia Sebagai Pendukung Keputusan Investasi Saham Emas Menggunakan Teknik Data Mining. *Kalbiscentia Jurnal Sains Dan Teknologi*, 2(1), 36–51. <http://files/511/Mahena et al. - 2015 - Prediksi Harga Emas Dunia Sebagai Pendukung Keputu.pdf>
- [12] Milatu, M. (2019). Pengaruh Hari Raya Idul Fitri Terhadap Inflasi Kota Tasikmalaya. *Jurnal Dinamika Ekonomi Pembangunan*, 2(1), 63. <https://doi.org/10.14710/jdep.2.1.63-69>
- [13] Pakaja, F., & Naba, A. (2015). Peramalan Penjualan Mobil Menggunakan Jaringan Syaraf Tiruan dan Certainty Factor. *Neural Networks*, 6(1), 23–28.
- [14] Rachmawati, Y. (2018). Pengaruh Inflasi dan Suku Bunga Terhadap Harga Saham Pada Perusahaan Perbankan Yang Terdaftar Di LQ45 Bursa Efek Indonesia. *Media Akuntansi*, 1(1), 69. <https://jurnal.univpgri-palembang.ac.id/index.php/mediasi/article/view/2368>
- [15] Riduan, M., Hariwijaya, I., Furqon, M. T., & Dewi, C. (2020). *Prediksi Harga Emas Dengan Menggunakan Metode Average-Based Fuzzy Time Series*. 4(4), 1258–1264.
- [16] Sanjaya, F. I., & Heksaputra, D. (2020). Prediksi Rerata Harga Beras Tingkat Grosir Indonesia dengan Long Short Term Memory. *JATISI (Jurnal Teknik Informatika Dan Sistem Informasi)*, 7(2), 163–174. <https://doi.org/10.35957/jatisi.v7i2.388>
- [17] Siami-Namini, S., Tavakoli, N., & Siami Namin, A. (2018). A Comparison of ARIMA and LSTM in Forecasting Time Series. *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 1394–1401. <https://doi.org/10.1109/ICMLA.2018.00227>
- [18] Sudirman, S., & Alhudhori, M. (2018). Pengaruh Konsumsi Rumah Tangga, Investasi Terhadap Pertumbuhan Ekonomi Di Provinsi Jambi. *EKONOMIS : Journal of Economics and Business*, 2(1), 81. <https://doi.org/10.33087/ekonomis.v2i1.33>
- [19] Walid, W. (2019). Peramalan Penjualan Harga Saham PT Bank Rakyat (Persero) Tbk BBRI Indonesia dengan Menggunakan Recurren Neural Network (RNN). *Prosiding Seminar Nasional Matematika*, 2, 139–147. <https://journal.unnes.ac.id/sju/index.php/prisma/article/view/28901>
- [20] Walid, W., Sukestiyarno, Y. L., & Sunarmi, S. (2018). Identifikasi Publikasi Dosen dalam Mewujudkan Internasionalisasi Universitas Negeri Semarang Menggunakan Neural Network. *Jurnal Mipa*, 41(2), 121–133.