

Prediksi Harga Bitcoin berdasarkan Data Historis Harian dan *Google Trend Index* menggunakan Algoritme *Extreme Learning Machine*

Panji Husni Padhila¹, Imam Cholissodin², Putra Pandu Adikara³

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: ¹panjihp10@ub.ac.id, ²imamcs@ub.ac.id, ³adikara.putra@ub.ac.id

Abstrak

Bitcoin telah menarik banyak perhatian dari media dan investor, mengingat fitur inovatifnya seperti desentralisasi dan keterlacakannya. Di beberapa negara telah menerima Bitcoin sebagai alat pembayaran. Bitcoin juga umum digunakan sebagai aset investasi walaupun cukup berbahaya karena harga dari Bitcoin sangat fluktuatif, yang mana harganya bisa naik dan turun dengan cepat dalam waktu yang singkat. Selain itu, Bitcoin juga dipercaya bersifat spekulatif, yang naik dan turun harganya bergantung pada pandangan orang terhadap *coin* tersebut. Penelitian ini bertujuan untuk memprediksi harga Bitcoin menggunakan algoritme Extreme Learning Machine (ELM) berdasarkan data historis harian dengan mempertimbangkan sifat spekulatifnya menggunakan Google Trend Index. Berdasarkan hasil pengujian yang dilakukan diperoleh hasil perhitungan Mean Absolute Percentage Error (MAPE) sebesar 3,089% menggunakan fungsi aktivasi Sigmoid, 5 fitur, 20 neuron pada hidden layer, dan menggunakan Google Trend Index dengan kata kunci 'Bitcoin'.

Kata kunci: *prediksi, bitcoin, extreme learning machine, google trend index*

Abstract

Bitcoin has attracted a lot of attention from the media and investors, given its innovative features such as its decentralization and traceability. In some countries have accepted Bitcoin as a means of payment. Bitcoin is also commonly used as an investment asset although it is quite dangerous because the price of Bitcoin is very volatile, which means that the price can go up and down quickly in a short time. In addition, Bitcoin is also believed to be speculative, whose price goes up and down depending on people's views of the coin. This study aims to predict the price of Bitcoin using the Extreme Learning Machine (ELM) algorithm based on daily historical data by considering its speculative nature using the Google Trend Index. Based on the results of the tests carried out, the results of the Mean Absolute Percentage Error (MAPE) calculation are 3,089% using the Sigmoid activation function, 5 features, 20 neurons in the hidden layer, and using Google Trend Index with the keyword 'Bitcoin'.

Keywords: *Prediction, bitcoin, extreme learning machine, google trend index*

1. PENDAHULUAN

Bitcoin adalah uang elektronik yang transaksinya dilakukan secara *online*, dikirimkan langsung dari satu pihak ke pihak lain tanpa melalui lembaga keuangan dengan menggunakan konsep *peer-to-peer* (Nakamoto, 2008). Bitcoin telah menarik banyak perhatian dari media dan investor, mengingat fitur inovatifnya seperti desentralisasi dan keterlacakannya (Fry & Cheah, 2016). Di beberapa negara telah menerima Bitcoin sebagai alat pembayaran. Bahkan untuk pertama kalinya, negara El-Salvador secara resmi menjadikan

Bitcoin sebagai mata uang resmi pembayaran negaranya sejak bulan Agustus 2021 (Roy, 2021).

Namun, Bitcoin juga cukup berbahaya jika dijadikan aset penyimpanan. Harga dari Bitcoin ini sangat fluktuatif, yang berarti harganya bisa naik dan turun dengan cepat dalam waktu yang singkat. Selain itu, Bitcoin juga bersifat spekulatif yang mana naik dan turun harganya bergantung pada pandangan orang terhadap *coin* tersebut. Efek spekulatif dapat langsung dilihat ketika muncul sebuah kebijakan terhadap Bitcoin oleh suatu negara. Seperti kebijakan yang dilakukan oleh negara China, yang pada

bulan Juni 2021 secara tegas melarang bank dan semua platform pembayarannya untuk melayani transaksi Bitcoin (Qin & Livni, 2021). Hal ini membuat harga Bitcoin dan *cryptocurrency* lainnya menjadi turun hingga kisaran 5%-15% dalam sehari. Bahkan, sebuah *tweet* atau kicauan pada media sosial Twitter yang dilakukan oleh tokoh terkenal Elon Musk juga dapat memengaruhi harga Bitcoin. Pada bulan Mei 2021, Elon Musk beberapa kali membuat heboh sosial media dengan pengaruh *tweet*-nya yang dapat membuat harga Bitcoin naik dan turun. Selain Bitcoin itu sendiri, sebuah *altcoin* (*coin* lain selain Bitcoin) yaitu Dogecoin mengalami kenaikan dan penurunan yang cukup parah. Perubahan lingkungan secara cepat terkait dengan pasar *cryptocurrency* mendesak kehati-hatian dalam melakukan pembelian Bitcoin (Choi & Shin, 2021). Untuk itu diperlukan sebuah sistem prediksi yang mampu memprediksi kapan harga Bitcoin tersebut akan naik atau turun.

Sebelumnya telah dilakukan penelitian untuk membangun sistem yang bisa memprediksi harga Bitcoin ini. Penelitian yang dilakukan oleh Radityo, et al. (2017) menggunakan algoritme Backpropagation mendapatkan hasil dengan MAPE sebesar 1,998%. Namun, kelemahan algoritme ini adalah waktu komputasinya yang cukup lama. Sebuah algoritme jaringan saraf tiruan lainnya yaitu algoritme Extreme Learning Machine atau ELM dianggap dapat melakukan *training* yang jauh lebih cepat. Seperti penelitian yang dilakukan oleh Faizal, et al. (2019) menggunakan algoritme ELM mendapatkan hasil MAPE sebesar 2,657%. ELM adalah sebuah jaringan saraf tiruan yang bersifat Single Layer Feedforward Neural Network (SLFNs) atau hanya memiliki satu *hidden layer*. ELM juga tidak membutuhkan perangkat komputer berspesifikasi tinggi untuk digunakan. Penelitian yang dilakukan oleh Nurdiansyah, et al. (2019) juga menggunakan ELM untuk prediksi harga Bitcoin. Namun, penelitian tersebut menambahkan metode optimasi Artificial Bee Colony (ABC) untuk inialisasi bobot dan bias awal untuk mendapatkan nilai yang optimal. ELM-ABC ini mendapatkan hasil MAPE sebesar 1,96983%. Angka tersebut sedikit lebih baik dari metode yang hanya menggunakan ELM saja. Namun, pada penelitian ini akan menggunakan ELM tanpa optimasi ABC karena optimasi tersebut akan menambah waktu komputasi, sedangkan perbedaan hasil MAPE-

nya tidak jauh berbeda.

Selain itu untuk meningkatkan efektifitas data, nilai indeks dari Google Trend Index akan digunakan. Nilai indeks yang ditampilkan oleh Google Trend berskala 0-100. Nilai ini menunjukkan seberapa banyak kata kunci tersebut dicari oleh seluruh pengguna Google Search atau mesin pencarian buatan Google. Pada penelitian yang dilakukan oleh Xu (2014) menunjukkan adanya korelasi yang kuat antara perubahan harga saham mingguan dengan indeks yang ada di website Google Trend. Mengingat sifat Bitcoin yang spekulatif membuat penulis tertarik menggunakan Google Trend Index sebagai salah satu tolak ukur dari prediksi harga Bitcoin. Berdasarkan masalah dan metode penelitian sebelumnya yang telah diuraikan, penulis memutuskan untuk melakukan penelitian dengan judul “Prediksi Harga Bitcoin Berdasarkan Data Historis Harian dan Google Trend Index Menggunakan Algoritme Extreme Learning Machine”.

2. METODOLOGI

2.1. Prediksi

Prediksi merupakan suatu proses memperkirakan sesuatu secara sistematis berdasarkan informasi yang ada atau dimiliki, untuk sesuatu yang akan terjadi atau belum diketahui. Prediksi berguna untuk membantu pengambilan keputusan serta perencanaan kedepannya. Dengan begitu, resiko yang akan terjadi bisa dihadapi dengan baik.

Terdapat dua jenis prediksi yaitu klasifikasi dan regresi. Klasifikasi adalah pendugaan kelompok atau golongan dari suatu entitas dengan aturan-aturan tertentu. Berbeda dengan klasifikasi, regresi adalah pendugaan nilai dari suatu entitas berdasarkan fitur yang dimiliki. Pada penelitian ini penulis menggunakan regresi untuk mendapatkan suatu nilai numerik yaitu harga Bitcoin.

2.2. Bitcoin

Bitcoin adalah salah satu jenis mata uang elektronik yang memungkinkan transaksi *online* terjadi secara langsung dari satu pihak ke pihak lain tanpa melalui lembaga keuangan dengan menggunakan konsep *peer-to-peer* (Nakamoto, 2008). Bitcoin merupakan *coin* yang pertama kali beredar dari sekian banyaknya jenis *cryptocurrency*. Bitcoin juga memiliki kapitalisasi pasar terbesar dibanding *coin*

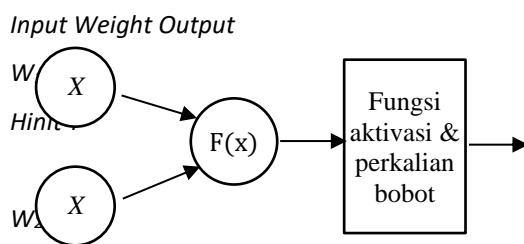
lainnya (CoinMarketCap, 2021). Hal ini membuat kenaikan dan penurunan harga Bitcoin dapat memengaruhi harga *coin* lainnya.

2.3. Google Trend

Google Trend merupakan salah satu layanan bebas biaya yang disediakan oleh Google yang berfungsi untuk menampilkan suatu nilai *index* dari *keyword* pencarian yang dilakukan oleh pengguna internet dari seluruh dunia. Pemilihan Google dibanding dengan mesin pencarian lain dikarenakan Google menguasai penggunaan mesin pencari dengan persentase sebesar 86,19% (Chaffey, 2022). Google Trend Index menormalisasikan jumlah pencarian dengan rentang nilai 0 – 100, yang jumlah pencarian terendah akan bernilai nol dan jumlah pencarian tertinggi akan bernilai seratus. Layanan Google ini mengizinkan pengguna untuk mengambil *trend index* dari sebuah *keyword* dengan rentang waktu yang bisa diatur, walaupun rentang waktu maksimal yang bisa diambil untuk mendapatkan *index* harian adalah 90.

2.4. Jaringan Saraf Tiruan

Jaringan Saraf Tiruan (JST) merupakan salah satu model komputasi yang menirukan cara kerja saraf pada otak manusia. JST memiliki kemiripan dengan neuron manusia yang berkaitan satu sama lain. Contoh struktur neuron sederhana pada JST dapat dilihat pada Gambar 1.



Gambar 1. Struktur JST Sederhana

2.5. Normalisasi

Proses ini bertujuan untuk menyamakan satuan dari semua fitur yang ada sehingga memiliki rentang nilai yang sama. Selain itu normalisasi juga membuat data terbebas dari *outlier* (pencilan). Terdapat beberapa jenis normalisasi, dan jenis yang digunakan pada penelitian ini adalah Min-Max Normalization. Rumus normalisasi tersebut dapat dilihat pada Persamaan 1.

$$x' = \frac{x - min}{max - min} \tag{1}$$

Keterangan:

- x = Nilai asli data
- min = Nilai minimum dari data
- max = Nilai maksimum dari data
- x' = Nilai hasil normalisasi data

2.6. Denormalisasi

Setelah data di-*training* dan *testing* menggunakan algoritme ELM, tahap selanjutnya adalah denormalisasi. Denormalisasi bertujuan untuk mengembalikan nilai yang sebelumnya telah dinormalisasi menjadi nilai asli kembali. Persamaan denormalisasi dapat dilihat pada Persamaan 2.

$$x = x' * (max - min) + min \tag{2}$$

Keterangan:

- x = Nilai hasil prediksi setelah denormalisasi
- min = Nilai minimum dari data
- max = Nilai maksimum dari data
- x' = Nilai hasil prediksi sebelum denormalisasi

2.7. Extreme Learning Machine

Extreme Learning Machine (ELM) adalah metode Jaringan Saraf Tiruan (JST) *feedforward* yang hanya memiliki satu *hidden layer* atau sering disebut Single Layer Feedforward Neural Network (SLFNs). Umumnya pada JST lain melakukan pembobotan dan menentukan nilai bias dari proses *training*, sedangkan ELM memperoleh pembobotan dan bias secara random. Hal itu membuat ELM memiliki *learning speed* atau kecepatan dalam mempelajari pola sangat cepat dibandingkan dengan JST lainnya (Huang, et al., 2005). ELM memiliki struktur yang cukup sederhana tanpa langkah yang panjang sehingga terhindar dari masalah *learning rate*, *overfitting*, dan *underfitting*.

Pada algoritme ELM ini terdapat dua tahapan yaitu fase *training* dan fase *testing*. Namun, perlu dilakukan normalisasi sebelum fase *training* dan denormalisasi setelah fase *testing*. Tujuannya agar data yang digunakan memiliki rentang yang sama. Setelah denormalisasi akan dilakukan perhitungan MAPE untuk evaluasi kinerja dari algoritme ELM. (Cholissodin, et al., 2019).

2.7.1 Fase *Training*

Pada fase *training* ELM, sistem dilatih agar bisa memprediksi secara akurat. Berikut ini langkah-langkah yang ada pada fase *training*:

1. Menentukan bobot dan bias. Nilai bobot dan bias ini ditentukan secara acak sesuai dengan rentang yang diberikan.

2. Mengalikan data yaitu matriks X_{latih} yaitu data untuk fase *training* dengan nilai bobot yang telah di-*transpose* (W^T) sehingga menghasilkan H_{init} . H_{init} diperlukan untuk mendapatkan *output hidden layer* (H). Persamaan H_{init} dapat dilihat pada Persamaan 3.

$$H_{init} = X_{latih} \times W^T \tag{3}$$

Keterangan:

H_{init} = Nilai hasil perkalian matriks X dan W^T

X = Nilai data (setelah normalisasi)

W^T = Nilai bobot yang di-*transpose*

3. Langkah selanjutnya adalah mendapatkan nilai *output hidden layer* (H). Nilai H ini didapatkan melalui perhitungan fungsi aktivasi. Terdapat beberapa jenis fungsi aktivasi yang dapat digunakan, salah satunya fungsi aktivasi *Sigmoid* Biner. Persamaan fungsi aktivasinya dapat dilihat pada Persamaan 4.

$$H = \frac{1}{1 + e^{-(H_{init} + bias)}} \tag{4}$$

Keterangan:

H = Nilai *output hidden layer*

e = Nilai bilangan *euler*

H_{init} = Nilai hasil perkalian matriks X dan W^T

bias = Nilai bias yang telah ditentukan pada langkah pertama

4. Kemudian nilai H ini diinverskan menggunakan Moore-Penrose Pseudo Inverse. *Inverse* matriks Moore-Penrose Pseudo Inverse ini dipilih karena bisa digunakan pada matriks berordo m x n yang singular. Hal tersebut berbeda dengan *invers* matriks biasa yang umumnya hanya bisa digunakan untuk meng-*invers*-kan matriks berordo n x n yang non-*singular* (Fikriya, et

al., 2017). Persamaannya dapat dilihat pada Persamaan 5.

$$H^+ = (H^T \times H)^{-1} \times H^T \tag{5}$$

Keterangan:

H^+ = Matriks Moore-Penrose Pseudo Inverse

H^T = Matriks H *transpose*

H = Nilai *output hidden layer*

5. Langkah selanjutnya adalah menghitung nilai *output weight*. Persamaannya bisa dilihat pada Persamaan 6.

$$\beta = H^+ \times T \tag{6}$$

Keterangan:

β = Matriks *output weight*

H^+ = Matriks Moore-Penrose Pseudo Inverse

T = Matriks target

6. Langkah terakhir pada fase *training* adalah menghitung nilai *output layer*. Rumusnya dapat dilihat pada Persamaan 7.

$$Y = H \times \beta \tag{7}$$

Keterangan:

Y = Matriks *output layer*

H = Nilai *output hidden layer*

β = Matriks *output weight*

2.7.2 Fase *Testing*

Fase ini bertujuan untuk melakukan pengetesan pada sisa data untuk memvalidasi nilai dari prediksi sistem. Hasil dari *testing* ini juga digunakan untuk evaluasi. Langkah-langkah yang diperlukan hampir sama dengan fase *training*. Perbedaannya terletak pada tidak adanya perhitungan Moore-Penrose Pseudo Inverse dan tidak adanya perhitungan *output weight*. Berikut adalah langkah-langkah fase *testing*:

1. Menentukan bobot dan bias. Nilai bobot dan bias ini ditentukan secara acak sesuai dengan rentang yang diberikan.
2. Menghitung nilai matriks H_{init} menggunakan persamaan yang mirip dengan fase *training*. Persamaan untuk menghitung nilai matriks dapat dilihat pada Persamaan 8.

$$H_{init} = X_{uji} \times W^T \tag{8}$$

3. Menghitung nilai matriks H menggunakan Persamaan 4.

4. Terakhir menghitung *output layer* menggunakan Persamaan 7.

2.8. Nested Cross Validation

Nested Cross Validation (NVC) merupakan salah satu metode pembagian data untuk mendapatkan parameter paling optimal pengujian. NVC terdiri dari *inner loop* untuk menentukan parameter optimal dan *outer loop* untuk estimasi kesalahan (Nasution, et al., 2020). Teknik NVC diimplementasikan dalam dua bentuk metode, yaitu *predict second half* yang menggunakan satu set training dan testing, dan *day forward chaining* yang membagi set menjadi banyak *split*. Penelitian ini akan menggunakan metode *predict second half* saat pengujian.

2.9. Mean Absolute Percentage Error

Mean Absolute Percentage Error (MAPE) merupakan salah satu metode evaluasi yang berfungsi untuk mengukur keakuratan dari suatu prediksi. MAPE lebih sering digunakan karena lebih mudah diinterpretasikan dan bersifat *scale-independent* atau tidak bergantung pada skala (Kim & Kim, 2016). Pada penelitian ini, rumus yang digunakan adalah rumus MAPE khusus yang terdapat pada buku Cholissodin, et al. (2019) yaitu untuk mengatasi permasalahan jika nilai hasil prediksi memiliki nilai nol. Hal tersebut disebabkan karena nilai hasil prediksi digunakan sebagai pembagi pada rumus ini. Rumus untuk menghitung MAPE khusus ini bisa dilihat pada Persamaan 9. Perbedaannya terletak pada adanya variabel *c* untuk memastikan tidak adanya pembagian dengan nilai nol.

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{(\hat{y}_i + c) - (y_i + c)}{(y_i + c)} \right| * 100 \tag{9}$$

Keterangan:

- \hat{y}_i = Hasil prediksi
- y_i = Nilai aktual
- n = Banyak data yang diuji
- c = konstanta

2.10. Data

Data untuk penelitian ini diperoleh dari *website* coinmarketcap.com yang berisi data historis harian (*time series*). Selain itu diambil juga data Google Trend Index dengan kata kunci “Bitcoin” dan “*cryptocurrency*” yang diharapkan bisa meminimalkan *error* dibanding penelitian terdahulu. Pemilihan kata kunci ini

berdasarkan penelitian yang dilakukan oleh Arratia & Barrantes (2021) dan Aslanidis, et al., (2022) yang menunjukkan ada korelasi yang berbanding lurus antara harga Bitcoin dengan Google Trend Index dengan kata kunci ‘Bitcoin’ dan ‘*cryptocurrency*’. Rentang waktu data adalah 240 hari mulai dari tanggal 23 Juli 2021 hingga tanggal 20 Maret 2021.

2.11. Diagram Alir Penelitian

Pada perancangan algoritme ini, akan dijabarkan langkah-langkah utama dalam penggunaan algoritme Extreme Learning Machine untuk memprediksi harga Bitcoin. Langkah-langkah yang diperlukan yaitu *input* data, *preprocessing* data, normalisasi data, fase *training*, fase *testing*, fase denormalisasi, fase perhitungan MAPE, dan output nilai prediksi dan nilai MAPE. Langkah-langkah perancangan algoritme digambarkan pada Gambar 2.

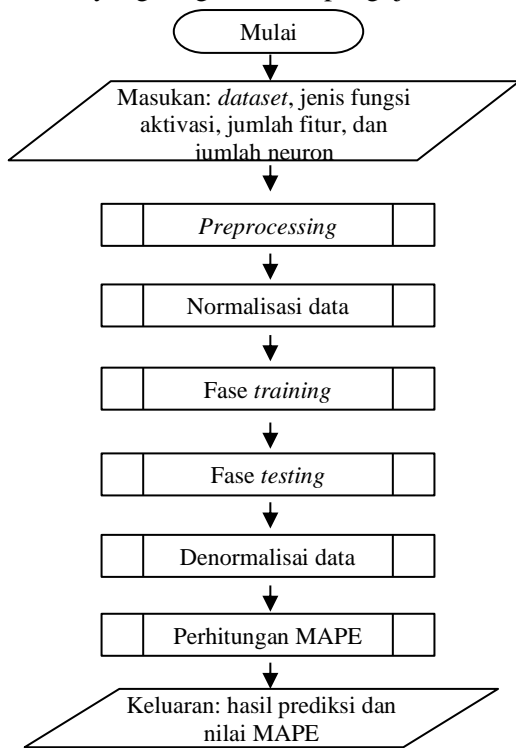
Langkah-langkah algoritme ELM yang terdapat pada Gambar 2 dijabarkan sebagai berikut:

1. Memasukkan data yang telah dikumpulkan, jenis fungsi aktivasi, banyak hari yang digunakan sebagai fitur, dan jumlah neuron pada *hidden layer*.
2. Melakukan *preprocessing* atau olah data untuk membentuk data agar bisa dimasukkan ke dalam algoritme ELM.
3. Melakukan normalisasi data menggunakan normalisasi Min-Max, normalisasi ini bertujuan untuk menyamakan skala data sesuai dari nilai terkecil dan nilai terbesar.
4. Melakukan *training* model algoritme ELM menggunakan data yang telah dinormalisasi, lalu mendapatkan nilai *beta* yang bisa digunakan pada fase *testing*.
5. Melakukan *testing* pada data *test* yang telah disiapkan sebelumnya untuk mendapatkan prediksi dari model algoritme ELM.
6. Menghitung nilai *error* menggunakan perhitungan MAPE agar diketahui kinerja dari algoritme ELM.
7. Menampilkan hasil prediksi dan hasil evaluasi perhitungan MAPE.

3. HASIL DAN PEMBAHASAN

Pada bagian ini akan dipaparkan hasil pengujian yang dilakukan sesuai rancangan skenario yang sudah dibuat. Hasil yang didapat dari pengujian adalah hasil dari perhitungan

MAPE. Sesuai dengan skenario, pengujian yang akan dilakukan adalah pengujian jenis fungsi aktivasi yang digunakan, pengujian rentang



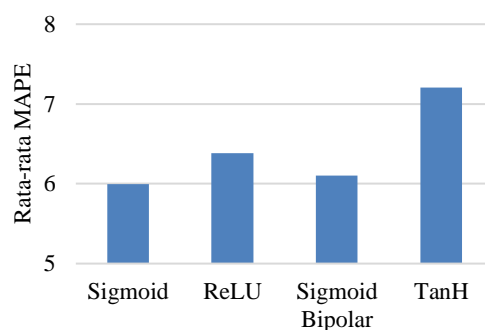
Gambar 2. Diagram Alir Penggunaan Algoritme ELM

bobot, pengujian banyaknya fitur, pengujian banyaknya neuron pada *hidden layer*, dan pengujian Google Trend Index. Setelah dilakukan pengujian, akan dilakukan pembahasan untuk setiap hasil pengujian.

4.1. Hasil Pengujian Fungsi Aktivasi

Pada pengujian fungsi aktivasi ini terdapat beberapa jenis fungsi aktivasi yang diujikan. Pengujian ini dilakukan karena setiap jenis fungsi aktivasi memiliki kelebihan dan kekurangan masing-masing. Oleh karena itu perlu diuji agar ditemukan fungsi aktivasi yang manakah yang paling sesuai dengan data dan algoritme yang digunakan. Hasil dari pengujian ini terdapat pada Tabel 1 dan Gambar 3.

Fungsi aktivasi	Rata-rata MAPE
Sigmoid	5,993%
reLU	6,387%
Sigmoid Bipolar	6,102%
TanH	7,206%



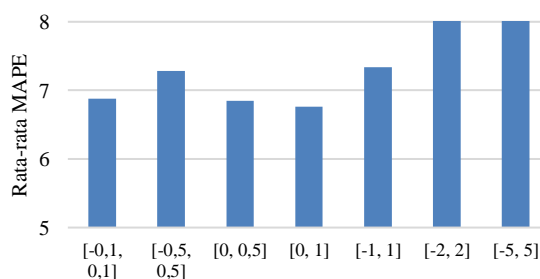
Gambar 3. Grafik Hasil Pengujian Fungsi Aktivasi

Pengujian ini menunjukkan bahwa fungsi aktivasi Sigmoid dan fungsi aktivasi Sigmoid Bipolar memiliki rata-rata MAPE terkecil. Namun, fungsi aktivasi Sigmoid memiliki nilai rata-rata MAPE sedikit lebih rendah. Di sisi lain fungsi aktivasi TanH mendapatkan nilai rata-rata MAPE paling tinggi.

4.2. Hasil Pengujian Rentang Bobot

Pengujian rentang bobot ini bertujuan untuk mendapatkan nilai rentang yang paling optimal untuk memaksimalkan kinerja model. Umumnya rentang bobot yang sering diberikan adalah [-1, 1]. Namun, rentang ini tidak selalu menghasilkan kinerja model yang optimal (Cao, et al., 2017). Pada Tabel 2 dan Gambar 4 ditampilkan hasil pengujian rentang bobot.

Rentang bobot	Rata-rata MAPE
[-0,1, 0,1]	6,882%
[-0,5, 0,5]	7,281%
[0, 0,5]	6,850%
[0, 1]	6,762%
[-1, 1]	7,339%
[-2, 2]	8,393%
[-5, 5]	9,329%



Gambar 4. Grafik Hasil Pengujian Rentang Bobot

Hasil pengujian ini menunjukkan bahwa rentang bobot sebesar [0, 1] memiliki rata-rata MAPE paling kecil. Besarnya rentang bobot ini dipengaruhi oleh fungsi aktivasi yang digunakan yaitu fungsi aktivasi Sigmoid yang juga memiliki rentang [0, 1]. Selain itu

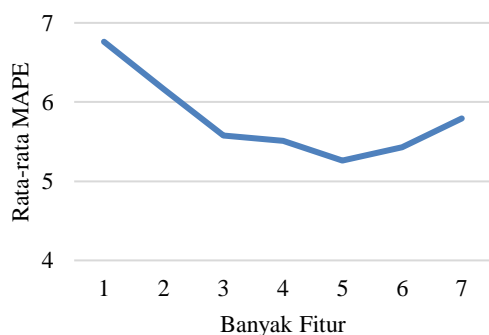
data yang dinormalisasikan juga memiliki rentang [0, 1] sehingga rentang bobot disini juga cocok menggunakan rentang 0 hingga 1.

4.3. Hasil Pengujian Banyak Fitur

Pengujian banyak fitur di sini maksudnya adalah berapa banyak hari yang digunakan untuk analisis teknikal. Pada bagian implementasi *preprocessing* didapatkan fitur *x1* sebagai harga *h-1* harga prediksi hingga *x7* sebagai harga *h-7* harga prediksi. Jika hanya menggunakan 1 fitur artinya hanya kolom *x1* yang digunakan sedangkan jika menggunakan 7 fitur, artinya ketujuh kolom akan digunakan. Pengujian ini bertujuan agar mengetahui berapa hari yang optimal untuk memprediksi harga besok. Hasil dari pengujian banyak fitur ini dapat dilihat pada Tabel 3 dan Gambar 5.

Tabel 3. Hasil Pengujian Banyak Fitur

Banyak Fitur	Rata-rata MAPE
1	6,762%
2	6,165%
3	5,579%
4	5,513%
5	5,261%
6	5,431%
7	5,792%



Gambar 5. Grafik Hasil Pengujian Banyak Fitur

Gambar 5 menunjukkan bahwa jumlah fitur sebanyak 5 menghasilkan rata-rata MAPE dengan nilai terkecil. Nilai rata-rata MAPE tertinggi yang diperoleh ketika hanya menggunakan 1 fitur. Hal ini dikarenakan 1 fitur saja belum cukup untuk memprediksi harga besoknya. Selain itu, ketika menggunakan lebih dari 5 fitur, rata-rata MAPE yang diperoleh kembali naik. Hal ini bisa terjadi karena sifat Bitcoin yang cukup fluktuatif sehingga harga *H-6* dari hari yang prediksi bisa berbeda jauh dengan harga hari prediksi.

4.1. Hasil Pengujian Banyak Neuron

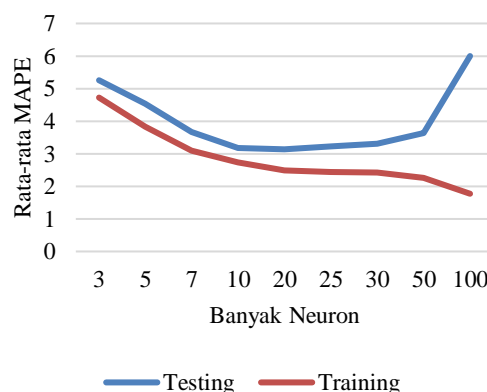
Pengujian banyaknya neuron pada *hidden layer* dilakukan untuk mengetahui berapa jumlah neuron yang optimal untuk digunakan (*best fit*) pada algoritme. Jumlah neuron ini dapat memengaruhi apakah hasil *training* mengalami *overfitting* atau *underfitting* terhadap data latih. Hasil dari pengujian ini terdapat pada Tabel 4 dan Tabel 5. Pada Gambar 6 ditampilkan grafik perbandingan hasil pengujian dengan data latih untuk melihat kondisi *overfitting* dan *underfitting*-nya.

Tabel 4. Hasil Pengujian Banyak Neuron Pada Fase *Training*

Banyak neuron	Rata-rata MAPE
3	4,725%
5	3,837%
7	3,102%
10	2,731%
20	2,497%
25	2,436%
30	2,425%
50	2,264%
100	1,772%

Tabel 5. Hasil Pengujian Banyak Neuron Pada Fase *Testing*

Banyak neuron	Rata-rata MAPE
3	4,725%
5	3,837%
7	3,102%
10	2,731%
20	2,497%
25	2,436%
30	2,425%
50	2,264%
100	1,772%



Gambar 6. Grafik Hasil Pengujian Banyak Neuron

Pengujian ini menunjukkan adanya pengaruh jumlah neuron terhadap kinerja model. Nilai MAPE yang didapatkan mengalami penurunan seiring dengan penambahan jumlah neuron. Saat neuron berjumlah 20, terjadi

kenaikan nilai MAPE kembali yang artinya di sini diketahui bahwa jumlah neuron yang paling optimal adalah 20. Jumlah neuron di bawah 20 akan menghasilkan model yang *underfitting* dan jumlah neuron di atas 20 akan menghasilkan model yang *overfitting*.

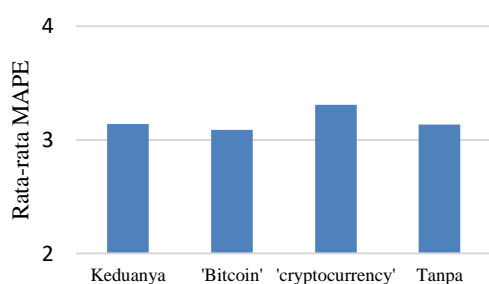
4.5. Hasil Pengujian Google Trend Index

Pengujian Google Trend Index dilakukan setelah mendapatkan parameter terbaik. Pengujian ini dilakukan untuk mengujikan apakah Google Trend Index menambah performa data atau sebaliknya. Hasil dari skenario pengujian ini dapat dilihat pada Tabel 6 dan grafik hasil dapat dilihat pada Gambar 7.

Hasil Pengujian ini menunjukkan adanya pengaruh ketika menggunakan Google Trend Index, yang mana ketika menggunakan kata kunci 'Bitcoin' rata-rata MAPE yang diperoleh turun menjadi 3,089%. Di sisi lain kata kunci 'cryptocurrency' membuat rata-rata MAPE naik menjadi 3,309%. Artinya Google Trend Index dapat membantu kinerja model jika hanya menggunakan kata kunci 'Bitcoin'.

Tabel 6. Hasil Pengujian Google Trend Index

Data	Rata-rata MAPE
Kedua kata kunci	3,137%
Hanya 'Bitcoin'	3,089%
Hanya 'cryptocurrency'	3,309%
Tanpa Google Trend Index	3,132%

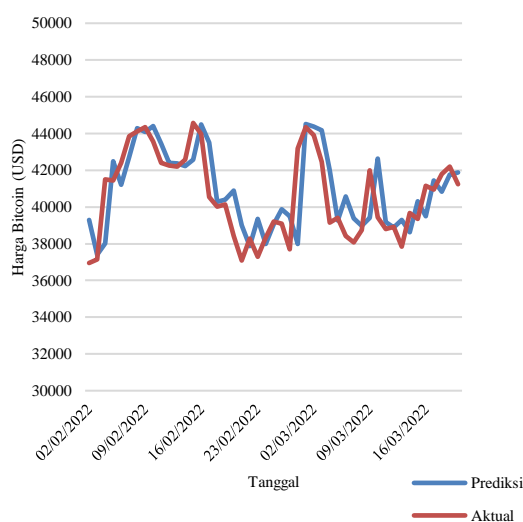


Gambar 7. Grafik Hasil Pengujian Google Trend Index

4.5. Pembahasan

Gambar 8 menampilkan grafik perbandingan antara hasil prediksi dengan data aktual. Grafik ini menunjukkan bahwa hasil prediksi cenderung terlambat satu hari. Hal ini disebabkan data yang digunakan untuk memprediksi adalah hari sebelumnya sehingga nilainya tidak jauh berbeda dengan hari sebelumnya. Mengingat harga Bitcoin yang sangat fluktuatif, hal ini

membuat hasil prediksi seperti terlambat satu hari.



Gambar 8. Grafik Perbandingan Hasil Prediksi dan Data Aktual

Berdasarkan pengujian yang telah dilakukan, didapatkan hasil nilai rata-rata MAPE terkecil dengan menggunakan parameter banyak fitur 5, fungsi aktivasi Sigmoid, jumlah neuron pada hidden layer sebanyak 20, dan menggunakan Google Trend Index dengan kata kunci 'Bitcoin'. Dengan parameter tersebut didapatkan nilai rata-rata MAPE sebesar 3,089%. Hasil ini membuktikan bahwa sebagian orang akan melakukan pencarian Google dengan kata kunci 'Bitcoin' sebelum melakukan transaksi Bitcoin sehingga Google Trend Index dengan kata kunci 'Bitcoin' bisa meningkatkan kinerja model.

Pada Pengujian ini juga ditemukan bahwa nilai MAPE setiap percobaan memiliki perbedaan nilai yang cukup signifikan. Hal ini menunjukkan bahwa nilai random yang didapatkan saat menentukan bobot pada algoritme training ELM bisa sangat memengaruhi kinerja algoritme-nya. Pembatasan rentang bobot pada saat inisialisasi juga belum dapat mengatasi perbedaan nilai yang signifikan ini.

4. KESIMPULAN

Perancangan algoritme ELM dengan menggunakan data historis harian dan Google Trend Index dalam memprediksi harga Bitcoin memerlukan beberapa tahapan. Data yang digunakan untuk memprediksi adalah data historis harian yang digabung dengan data Google Trend Index. Setelah itu, dilakukan preprocessing untuk mendapatkan data harga

hari-hari sebelum harga prediksi yang berbentuk time series. Algoritme ELM baru bisa dilakukan setelah data time series didapatkan. Namun, sebelum itu perlu dilakukan normalisasi untuk mengurangi pencilaan. Sebaliknya, setelah melakukan prediksi perlu dilakukan denormalisasi untuk mendapatkan hasil yang sebenarnya. Algoritme yang telah dirancang diujikan dalam empat jenis skenario untuk mendapatkan parameter yang yang paling optimal. Skenario pengujian yang dilakukan adalah pengujian banyak fitur, pengujian jenis fungsi aktivasi yang digunakan, pengujian banyak neuron pada hidden layer, serta pengujian Google Trend Index. Pengujian ini menunjukkan bahwa algoritme ELM akan optimal dengan menggunakan 5 fitur, fungsi aktivasi Sigmoid, 20 neuron pada hidden layer, dan menggunakan Google Trend Index dengan kata kunci 'Bitcoin'.

Hasil evaluasi algoritme ELM dapat dilihat pada nilai MAPE yang didapatkan dengan menggunakan parameter terbaik. Nilai MAPE yang didapatkan memiliki rata-rata sebesar 3,089% dan nilai MAPE terkecil menggunakan parameter tersebut sebesar 2,910%. Dibandingkan dengan data tanpa Google Trend Index, angka yang diperoleh adalah sebesar 3,132%. Angka ini menunjukkan bahwa Google Trend Index dapat menambah performa algoritme walaupun selisih rata-rata MAPE-nya hanya sebesar 0,043%. Selisih yang rendah ini disebabkan Google Trend Index hanya memberikan indeks seberapa banyak orang yang melakukan pencarian tentang Bitcoin tanpa memperhitungkan apakah spekulasinya positif atau negatif.

5. DAFTAR PUSTAKA

- Arratia, A. & Barrantes, A. . X. L., 2021. Do Google Trends Forecast Bitcoins? Stylized Facts and Statistical Evidence. *Journal of Banking and Financial Technology*, Volume 5, pp. 45-57.
- Aslanidis, N., Bariviera, A. F. & López, Ó. G., 2022. The link between cryptocurrencies and Google Trends attention. *Finance Research Letters*, Volume 47.
- Atsalakis, G. S., Atsalaki, I. G., Pasiouras, F. & Zopounidis, C., 2019. Bitcoin price forecasting with neuro-fuzzy techniques. *European Journal of Operational Research*, Volume 276, pp. 770-780.
- Cao, W., Gao, J., Ming, Z. & Cai, S., 2017. Some Tricks in Parameter Selection for Extreme Learning Machine. *IOP Conference Series: Materials Science and Engineering*, Volume 261, p. 012002.
- Chaffey, D., 2022. *Search engine marketing statistics 2022*. [Online] Available at: <https://www.smartinsights.com/search-engine-marketing/search-engine-statistics/> [Accessed 20 07 2022].
- Choi, S. & Shin, J., 2021. Bitcoin: An inflation hedge but not a safe haven. *Finance Research Letters*.
- Cholissodin, I. et al., 2019. *Buku Ajar AI, Machine Learning & Deep Learning*. Malang: Fakultas Ilmu Komputer (FILKOM), Universitas Brawijaya (UB).
- CoinMarketCap, 2021. *CoinMarketCap*, s.l.: s.n.
- Faizal, R., Setiawan, B. D. & Cholissodin, I., 2019. Prediksi Nilai Cryptocurrency Bitcoin menggunakan Algoritme Extreme Learning Machine (ELM). *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 3(5), pp. 4226-4233.
- Fikriya, Z. A., Irawan, M. I. & Soetrisno, 2017. Implementasi Extreme Learning Machine untuk Pengenalan Objek Citra Digital. *Jurnal Sains dan Seni ITS*, 6(1), pp. A 18 - A 23.
- Fry, J. & Cheah, E.-T., 2016. Negative bubbles and shocks in cryptocurrency markets. *International Review of Financial Analysis*, Volume 47, pp. 343-352.
- Huang, G.-B., Zhu, Q.-Y. & Siew, C.-K., 2005. Extreme Learning Machine: Theory and Applications. *Neurocomputing*, Volume 70, pp. 489-501.
- Kim, S. & Kim, H., 2016. A new metric of absolute percentage error for intermittent demand forecasts. *International Journal of Forecasting*, Volume 32, pp. 669-679.
- Liu, M. et al., 2021. Forecasting the price of Bitcoin using deep learning. *Finance Research Letters*, Volume 40.
- Nakamoto, S., 2008. Bitcoin: A Peer-to-Peer Electronic Cash System.

- Nasution, M. A. P., Cholissodin, I. & Indriati, 2020. Prediksi Price Earning Ratio Saham Menggunakan Algoritme Kernel Extreme Learning Machine (Studi Kasus: PT TELKOM). *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 4(10), pp. 3455-3462.
- Nurdiansyah, A., Furqon, M. T. & Rahayudi, B., 2019. Prediksi Harga Bitcoin Menggunakan Metode Extreme Learning Machine (ELM) dengan Optimasi Artificial Bee Colony (ABC). *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 3(6), pp. 5531-5539.
- Qin, A. & Livni, E., 2021. *China Cracks Down Harder on Cryptocurrency With New Ban*, New York: The New York Times.
- Radityo, A., Munajat, Q. & Budi, I., 2017. Prediction of Bitcoin Exchange Rate to American Dollar Using Artificial Neural Network Methods. *2017 International Conference on Advanced Computer Science and Information Systems, ICACISIS 2017*, Volume 2018-January, pp. 433-437.
- Roy, A., 2021. *In El Salvador, More People Have Bitcoin Wallets Than Traditional Bank Accounts*. [Online] Available at: <https://www.forbes.com/sites/theapothecary/2021/10/07/in-el-salvador-more-people-have-bitcoin-wallets-than-traditional-bank-accounts/?sh=4196f90420b5> [Accessed 09 10 2021].
- Xu, S. Y., 2014. Stock Price Forecasting Using Information from Yahoo Finance and Google Trend. *UC Berkeley*.