

OPTIMIZATION OF STOCK PRICE PREDICTION WITH RIDGE REGRESSION AND HYPERPARAMETER SELECTIONS

Adeline Fellita Marwa^{*1}, Sitti Ayuningrum Setiyawan^{*2}, Yonaka Titin Nur Cahyani^{*3}, Hasan Dwi Cahyono⁴

^{1,2,3,4}Faculty of Information Technology and Data Science, Sebelas Maret University
Jl. Ir. Sutami No. 36 Kentingan, Jebres, Surakarta, Jawa Tengah, Indonesia 57126

Email: ¹adelinefellita@student.uns.ac.id, ²sittiayuningrum@student.uns.ac.id, ³yonakatitin@student.uns.ac.id,
⁴hasandc@staff.uns.ac.id

(Article received: June 26, 2024; Revision: September 6, 2024; published: February 20, 2025)

Abstract

Stock price prediction is a topic that has garnered significant attention in the investment world and has been the subject of various studies. Despite the massive attention, predicting stock price movements using algorithms remains challenging as the algorithms must be agile and highly adaptive to movement trends. Recent studies using deep learning methods for stock price prediction show that deep learning methods have high reliability. However, their computational complexity limits widespread implementation. This study aims to predict Netflix stock prices using a linear regression model with ridge and hyperparameter optimization. The research consists of three stages: data preprocessing, building a linear regression model with ridge, and predicting and visualizing results. The dataset used is historical Netflix stock price data from 2017 to 2022. In the preprocessing stage, the data was normalized using *MinMaxScaler* and split into training and test sets. A ridge regression model was built with hyperparameter alpha optimization using *GridSearch*. Predictions were compared to stock prices and evaluated using *Root Mean Squared Error (RMSE)*. The ridge regression model with hyperparameter optimization performed best with an RMSE of 13.8082. Although the linear regression model demonstrated the fastest execution time of 0.7717 seconds, the ridge regression model with hyperparameter optimization provided an optimal balance between prediction accuracy and time efficiency.

Keywords: hyperparameter, linear regression, prediction, ridge, stock price

OPTIMASI PREDIKSI HARGA SAHAM MENGGUNAKAN RIDGE REGRESSION DAN SELEKSI HIPERPARAMETER

Abstrak

Prediksi harga saham merupakan topik yang banyak menarik perhatian dalam dunia investasi dan telah menjadi subjek berbagai penelitian. Meskipun demikian, prediksi pergerakan harga saham menggunakan algoritma tetap menantang karena algoritma harus gesit dan sangat adaptif terhadap tren pergerakan. Berbagai penelitian terbaru yang menggunakan metode *deep learning* untuk melakukan prediksi harga saham dan menunjukkan bahwa metode *deep learning* memiliki keandalan tinggi. Namun, kompleksitas komputasionalnya membatasi implementasinya secara luas. Penelitian ini bertujuan memprediksi harga saham Netflix menggunakan model regresi linear dengan *ridge* dan optimasi *hyperparameter*. Penelitian terdiri dari tiga tahap: pra-pemrosesan data, membangun model regresi linear dengan *ridge*, serta prediksi dan visualisasi hasil. Dataset yang digunakan adalah data historis harga saham Netflix dari tahun 2017 hingga 2022. Pada tahap pra-pemrosesan, data dinormalisasi dengan *MinMaxScaler* dan dibagi menjadi *training set* serta *test set*. Model *ridge regression* dibangun dengan optimasi *hyperparameter alpha* menggunakan *Grid Search*. Hasil prediksi dibandingkan dengan harga saham sebenarnya dan dievaluasi menggunakan *Root Mean Squared Error (RMSE)*. Model *ridge regression* dengan optimasi *hyperparameter* menunjukkan performa terbaik dengan RMSE sebesar 13,8082. Meskipun model *linear regression* menunjukkan waktu eksekusi tercepat sebesar 0,7717 detik, model *ridge regression* dengan optimasi *hyperparameter* memberikan keseimbangan optimal antara akurasi prediksi dan efisiensi waktu.

Kata kunci: harga saham, hiperparameter, prediksi, regresi linear, ridge

1. PENDAHULUAN

1.1 Saham

Suatu investasi yang mendapatkan peluang memperoleh return yang tinggi, jangka pendek, dan cepat disebut dengan Saham. Indeks Harga Saham Gabungan (IHSG) atau Indeks Harga Saham Komposit dapat digunakan untuk melihat perubahan harga saham untuk melihat peningkatan harga saham dan antusiasme pasar dan sebaliknya [1]. Pasar saham adalah tempat di mana orang dapat membeli dan menjual saham perusahaan yang diperdagangkan secara umum untuk menghasilkan uang [2].

Hukum penawaran dan permintaan dalam menentukan harga saham sangat berpengaruh. Artinya, jika menurut pembeli nilai saham akan naik, maka mereka akan sukarela membayar untuk lebih dari satu saham dan begitu pun dengan penjual. Apabila nilai saham tersebut turun, maka mereka bersedia menerima lebih sedikit hasilnya [3]. Analisis sentimen digunakan pada berita utama artikel terkait saham, tweet, berita keuangan dari "*Economic Times*" dan komentar Facebook [4]. Metode tradisional seperti analisis teknikal dan fundamental sering kali terbatas dalam memprediksi pergerakan harga saham yang kompleks dan dinamis [5]. Oleh karena itu, penggunaan metode statistik dan pembelajaran mesin seperti *Deep Learning* telah meningkat secara signifikan untuk menangani kompleksitas ini [6].

1.2 Analisis Regresi

Analisis regresi (*linear regression*) adalah salah satu teknik statistik untuk menyelidiki dan memodelkan hubungan antara variabel dependen (prediksi stok) dan satu atau lebih variabel independen (fitur yang mempengaruhi harga saham) dengan asumsi hubungan linear [7]. Meskipun begitu, metode ini rentan terhadap *overfitting*, terutama ketika jumlah fitur yang digunakan sangat banyak. Untuk mengatasi *overfitting*, salah satu teknik yang efektif adalah penggunaan regularisasi [8]. Penalti ini membantu menjaga koefisien tetap kecil, yang akan mengurangi kompleksitas model dan risiko *overfitting* [9]. *Ridge regression*, juga dikenal sebagai *L2 regularization*, teknik yang menambahkan penalti pada besarnya koefisien dalam model *linear regression* [10]. Tetapi menggunakan *ridge regression* saja belum tentu menghasilkan model yang ideal. Oleh karena itu, diperlukan optimasi *hyperparameter*, untuk menemukan nilai parameter terbaik. Metode yang dikenal sebagai optimalisasi *hyperparameter* digunakan untuk menentukan parameter model yang

paling ideal, seperti nilai *alpha* dalam *ridge regression*, untuk meningkatkan kinerja model prediksi.

1.3 Random Forest

Selain *ridge regression* dengan optimasi *hyperparameter*, metode *random forest* dan LSTM (*Long Short-Term Memory*) juga sering digunakan dalam prediksi harga saham. *Random forest* adalah metode *ensemble learning* yang menggunakan banyak pohon keputusan untuk meningkatkan akurasi prediksi dan mengurangi *overfitting* [11]. Metode ini efektif dalam menangani data yang tidak linier dan dapat menangkap hubungan kompleks antara fitur-fitur yang mempengaruhi harga saham. Namun, *random forest* memiliki kekurangan, seperti kecenderungan untuk memerlukan banyak waktu dan sumber daya komputasi saat menangani *dataset* yang sangat besar. Selain itu, *random forest* dapat menghasilkan model yang sulit untuk diinterpretasikan karena kompleksitasnya. Untuk mengatasi kekurangan ini, sering kali menggunakan teknik-teknik tambahan seperti seleksi fitur. Gunanya untuk melihat fitur mana yang memiliki efek paling besar dibandingkan fitur lainnya dan untuk mengurangi jumlah fitur yang digunakan dan mempercepat proses komputasi.

1.4 Long Short Term Memory

LSTM adalah jenis jaringan saraf berulang (*Recurrent Neural Network*) yang mampu menangani data urutan waktu (*time series*) dengan baik, membuatnya sangat cocok untuk memprediksi harga saham berdasarkan data historis [12]. Namun, LSTM juga memiliki kekurangan, termasuk kebutuhan komputasi yang tinggi dan waktu pelatihan yang panjang. Selain itu, LSTM rentan terhadap masalah *overfitting* jika tidak diatur dengan benar dan dapat menjadi sulit untuk diimplementasikan dan dioptimalkan karena banyaknya *hyperparameter* yang perlu disesuaikan [13]. LSTM cenderung memakan waktu lama untuk dilatih dalam mencapai konvergensi dan menyesuaikan bobot dengan benar berdasarkan data urutan waktu yang panjang [14]. Arsitektur LSTM yang kompleksitas, membuat proses pelatihan melibatkan banyak data historis dan parameter sehingga memerlukan waktu dan sumber daya komputasi yang besar [15]. Pada penelitian [16] membandingkan model ARIMA dan LSTM, menunjukkan bahwa LSTM memerlukan waktu pelatihan lebih lama dibandingkan dengan ARIMA, terutama untuk *dataset* yang besar.

Model CNN, setelah menyesuaikan *hyperparameter* melalui *GridSearchCV*, meningkatkan

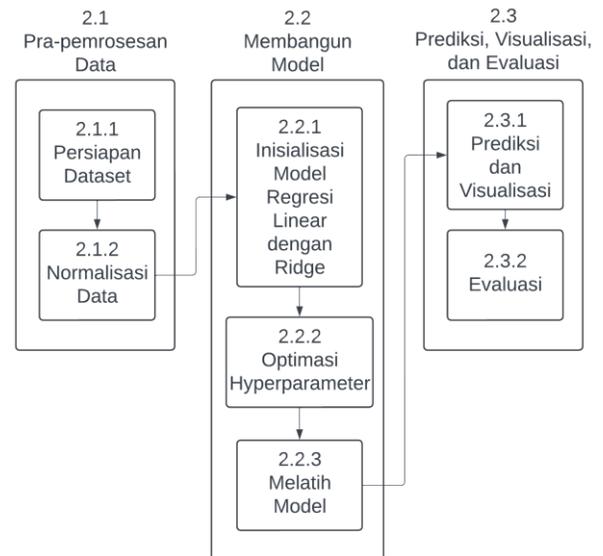
kinerja dibandingkan dengan CNN tanpa optimasi, mengoptimalkan *hyperparameter* untuk meningkatkan presisi dan akurasi prediksi [17]. *Hyperparameter* merupakan parameter yang digunakan untuk membentuk model *machine learning* (ML) misalnya, parameter penalti C untuk mesin vektor dukungan, dan kecepatan pembelajaran untuk melatih jaringan syaraf atau untuk menentukan algoritma yang digunakan untuk meminimalkan fungsi kerugian [18]. Dengan mengidentifikasi kombinasi parameter yang menghasilkan prediksi yang paling akurat, optimasi *hyperparameter* dapat meningkatkan kinerja model. Dalam penelitian ini, kami akan menerapkan *Ridge regression* dan optimasi *hyperparameter* untuk memprediksi harga saham, dengan tujuan untuk meningkatkan akurasi prediksi dan mengurangi risiko *overfitting*.

2. METODE PENELITIAN

Penelitian ini terdiri dari tiga tahap utama yakni pra-pemrosesan data, membangun model *ridge regression*, dan prediksi serta visualisasi hasil. Pada tahap pertama, yaitu pra-pemrosesan data dilakukan persiapan *dataset* dengan membagi data menjadi *training set* dan *test set* kemudian normalisasi data menggunakan skala fitur. Kemudian pada tahap kedua yaitu membangun model *ridge regression*, menambahkan optimasi *hyperparameter* menggunakan *GridSearch* untuk menemukan nilai *alpha* terbaik, melatih model menggunakan *training set* dengan *hyperparameter* terbaik. Tahap terakhir adalah melakukan prediksi dan visualisasi hasil, yaitu dengan membuat prediksi harga saham menggunakan model yang telah dilatih, memvisualisasikan hasil prediksi dan membandingkannya dengan data harga saham sebenarnya, mengevaluasi model menggunakan *Root Mean Squared Error* (RMSE) untuk mengukur akurasi prediksi. Metode penelitian untuk prediksi harga saham menggunakan *ridge regression* dan optimasi *hyperparameter* digambarkan pada Gambar 1. Agar hasil dari penelitian ini dapat dilakukan replikasi dan juga dikembangkan lebih lanjut, kami mengunggah *source code* pada github.com/adelinefellita/Optimasi-Prediksi-Harga-Saham.

2.1. Pra-pemrosesan Data

Pada tahap pra-pemrosesan data adalah tahapan di mana melibatkan persiapan *dataset* sebelum digunakan untuk membangun model. Untuk memastikan bahwa semua variabel berada dalam rentang yang sama. Proses ini mencakup pembagian data menjadi *train set* dan *test set*, lalu kemudian normalisasi data menggunakan skala fitur.



Gambar 1. Kerangka kerja penelitian.

2.1.1 Persiapan *Dataset*

Data yang digunakan dalam penelitian ini terdiri dari atribut 'Date', 'Open', 'High', 'Low', 'Close', 'Adj Close', dan 'Volume' yang digambarkan dalam Tabel 1.

Tabel 1. Atribut *dataset*

No	RMSE	T(s)
1	Date	2017-02-23; 2019-01-02; dll
2	Open	259.279999; 260.160004; dll
3	High	270.100006; 269.750000; dll
4	Low	260.000000; 256.579987; dll
5	Close	267.660004; 267.660004; dll
6	Adj Close	267.660004; 267.660004; dll
7	Volume	13508900; 11679500; dll

Dataset ini dibagi menjadi dua bagian, yaitu *training set* dan *test set*, yang disimpan dalam *file CSV* (*comma-separated values*). *Training set* digunakan untuk melatih model, sementara *test set* digunakan untuk mengevaluasi kinerja model. Pemisahan ini dilakukan untuk memastikan bahwa model dapat digeneralisasi dengan baik pada data yang tidak terlihat selama pelatihan.

2.1.2 Normalisasi Data

Dalam proses pra-pemrosesan, skala fitur dilakukan untuk menormalkan data ke dalam rentang nilai [0, 1]. Ini dilakukan untuk mendapatkan konvergensi yang lebih cepat. Skala fitur dilakukan menggunakan formula berikut:

$$x_{norm} = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (1)$$

dimana x_{norm} adalah nilai hasil normalisasi, x adalah nilai sebenarnya yang akan dinormalisasi, $\max(x)$ adalah nilai maksimum dari data sebenarnya, dan $\min(x)$ adalah nilai minimum dari data sebenarnya.

2.2. Membangun Model

Pada tahap ini, model *Ridge Regression* akan dibangun melalui beberapa langkah utama. Langkah-langkah tersebut meliputi inialisasi model, optimasi *hyperparameter*, dan pelatihan model. Masing-masing langkah ini bertujuan untuk menghasilkan model yang optimal dalam memprediksi harga saham.

2.2.1 Inialisasi Model Regresi Linear dengan Ridge

Model *ridge regression* dapat membuat koefisien model lebih kecil dan mengurangi dampak multikolinieritas [19]. Regresi linear sederhana dapat dirumuskan sebagai berikut:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_n x_n \quad (2)$$

dimana y adalah variabel dependen, β_0 adalah intercept, $\beta_1, \beta_2, \dots, \beta_n$ adalah koefisien regresi, dan x_1, x_2, \dots, x_n adalah variabel independen. *Ridge regression* menambahkan regularisasi terhadap besarnya koefisien regresi untuk mengatasi *overfitting* [20], yang dirumuskan sebagai berikut:

$$J(\beta) = \sum_{i=1}^m (y_i - \hat{y}_i)^2 + \alpha \sum_{j=1}^n \beta_j^2 \quad (3)$$

dimana $J(\beta)$ adalah fungsi cost, y_i adalah nilai sebenarnya, \hat{y}_i adalah nilai prediksi, α adalah parameter regulasi (*hyperparameter*), dan β_j adalah koefisien regresi.

2.2.2 Optimasi Hyperparameter

Optimasi *hyperparameter* dilakukan menggunakan *GridSearch* untuk mencari nilai optimal [21] dari α dalam rentang 0.001 hingga 0.1. *GridSearch* mengevaluasi model dengan berbagai kombinasi *hyperparameter* dan memilih yang menghasilkan kinerja terbaik berdasarkan metrik yang telah ditentukan, seperti *mean squared error* (MSE) atau RMSE.

2.2.3 Melatih Model

Setelah menemukan *hyperparameter* terbaik, model dilatih menggunakan data pelatihan dengan *fitting* model pada *training set* dan menyimpan model terbaik. Selain *ridge regression*, penelitian ini juga melibatkan pelatihan model menggunakan algoritma lain. *Recurrent Neural Network - Long Short Term Memory* (RNN_LSTM) adalah jenis RNN yang dirancang untuk menangani masalah *long-term dependencies* dengan mengingat informasi dalam jangka waktu yang panjang.

Gated Recurrent Unit (GRU) adalah variasi dari LSTM yang lebih sederhana dan cepat, dengan

performa yang sering kali sebanding. *Linear Regression Ridge* (LRR) menambahkan regularisasi α ke model regresi linear untuk mengatasi masalah multikolinieritas dan *overfitting*. *Linear Regression* (LR) adalah model regresi linear dasar digunakan untuk memprediksi nilai kontinu berdasarkan hubungan linear antara variabel independen dan dependen. *Random Forest* (RF) merupakan algoritma *ensemble learning* yang menggabungkan prediksi dari beberapa pohon keputusan untuk meningkatkan akurasi dan mengurangi *overfitting*.

2.3. Prediksi, Visualisasi, dan Evaluasi

Pada tahap terakhir ini, model yang telah dilatih digunakan untuk melakukan prediksi harga saham. Hasil prediksi kemudian divisualisasikan dan dibandingkan dengan data harga saham sebenarnya. Evaluasi model dilakukan menggunakan RMSE untuk mengukur akurasi prediksi.

2.3.1 Prediksi dan Visualisasi

Setelah pelatihan model pada data harga saham *Netflix*, prediksi dilakukan menggunakan data pengujian. Penentuan akurasi prediksi didasarkan pada jumlah kesalahan (*loss*) yang dihasilkan, di mana semakin kecil kesalahan, semakin baik model tersebut [22]. Prediksi harga saham *Netflix* dari tahun 2017-2022 divisualisasikan dalam grafik untuk membandingkan kondisi sebenarnya dengan prediksi (Gambar 2).



Gambar 2. Harga Saham *Netflix* tahun 2017-2022.

2.3.2 Evaluasi

Evaluasi model dilakukan menggunakan metode RMSE untuk mengukur akurasi hasil prediksi dengan menghitung persentase kesalahan antara nilai sebenarnya dan nilai prediksi [23]. Semakin kecil nilai RMSE, semakin baik model dalam memprediksi data. RMSE dihitung menggunakan formula berikut:

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_{pred} - y_{ref})^2} \quad (4)$$

dimana y_{pred} adalah nilai prediksi, y_{ref} adalah nilai sebenarnya, dan N adalah jumlah data.

3. HASIL DAN PEMBAHASAN

Dalam hasil dan pembahasan, langkah-langkah yang dilakukan pada percobaan ini mengacu pada metodologi penelitian (*Gambar 1*). Tahapan awal dimulai dengan mempersiapkan *dataset*. Selanjutnya, *dataset* yang telah dipersiapkan tersebut, digunakan untuk membangun model. Hingga, model yang telah dihasilkan dapat digunakan untuk melakukan prediksi. Hasil dari tahapan-tahapan tersebut tersaji dalam beberapa sub bab.

3.1. Pra-pemrosesan Data

Untuk memulai proses prediksi harga saham *Netflix*, langkah pertama yang harus dilakukan yaitu persiapan *dataset*. *Dataset* yang digunakan terdiri dari dua, yaitu: data pelatihan dan data pengujian, untuk data pelatihan diperoleh dari file 'Netflix-train.csv', sementara untuk data pengujian diperoleh dari file 'Netflix-test.csv'. Kemudian, dilakukan skala fitur pada data pelatihan dengan menggunakan *MinMaxScaler* untuk menormalisasi data ke dalam rentang 0 hingga 1. Kemudian, data pelatihan diubah menjadi struktur data dengan 60 *timestep* dan 1 *output*.

Setelah persiapan data pelatihan selesai. Selanjutnya yaitu mempersiapkan data pengujian. Data pengujian digabungkan dengan data pelatihan untuk membangun struktur data pengujian. Proses ini diikuti dengan membentuk struktur data pengujian serupa dengan data pelatihan, dengan 60 *timestep* sebagai *input*.

3.2. Membangun Model

Pada proses pembangunan model untuk memprediksi harga saham *Netflix* menggunakan regresi linear dengan regularisasi *Ridge* dimulai dengan menentukan parameter optimal melalui *Grid Search*. Langkah pertama yaitu mendefinisikan model *Ridge* dan menentukan *grid hyperparameter* yang akan dicari. *Hyperparameter* yang dioptimalkan adalah *alpha*, yang mengontrol kekuatan regularisasi. *Grid Search* dilakukan dengan rentang *alpha* 0,001 hingga 0,1. Setelah dilakukan pencarian *hyperparameter*, nilai *alpha* terbaik diperoleh, yaitu nilai yang memberikan skor *Mean Squared Error* (MSE) negatif terendah. Nilai ini digunakan untuk melatih model akhir. Dapat diamati pada Tabel 2. Yaitu performa model *Linear Regression Ridge* dengan *Optimasi Hyperparameter* (LRR_OH), *Linear Regression Ridge* (LRR), *Linear Regression* (LR), dan *Random Forest* (RF).

Tabel 2. Performa model jaringan saraf tanpa iterasi berulang

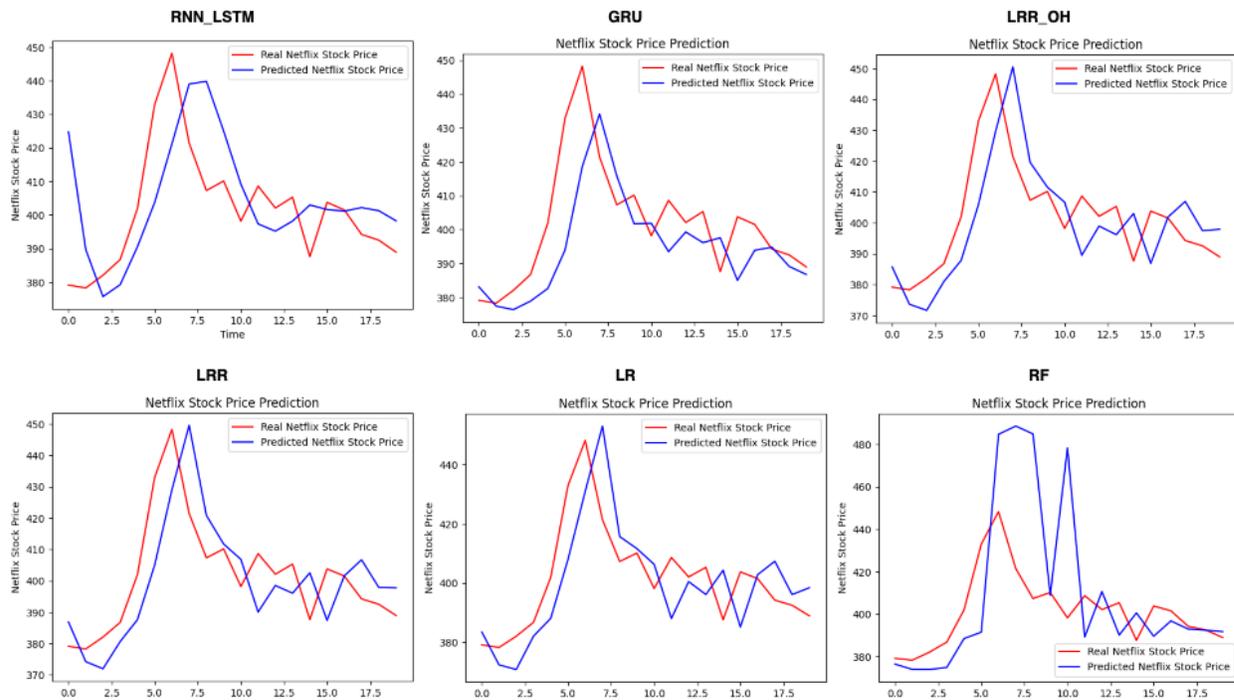
Model	RMSE	T(s)
LRR_OH	13.8028	1.5960
LRR	13.8194	0.9522
LR	13.9136	0.7717
RF	32.7955	7.8303

Untuk model *Linear Regression Ridge* dengan *Optimasi Hyperparameter* (LRR_OH) pada tahap Pra-Pemrosesan Data, struktur data dibuat dengan 60 *timesteps* untuk memprediksi harga saham berikutnya, model *Ridge Regression* dibangun dan dilakukan optimasi *hyperparameter* untuk menemukan nilai *alpha* terbaik, Optimasi dilakukan menggunakan *GridSearchCV* untuk menemukan nilai *alpha* terbaik dalam rentang 0,001 hingga 0,1. Nilai *Ridge* (*alpha*) merupakan parameter untuk menentukan kekuatan regulasi. Semakin tinggi nilai *alpha* akan meningkatkan penalti terhadap koefisien besar, sehingga menghasilkan model yang lebih sederhana.

Model *Linear Regression* (LR) sederhana tanpa regularisasi. Model ini cepat dalam eksekusi namun memiliki RMSE yang relatif tinggi dibandingkan dengan model lainnya. kemudian model selanjutnya *Linear Regression Ridge* (LRR) sedikit menurunkan nilai RMSE, menunjukkan peningkatan kecil dalam akurasi prediksi. Namun, waktu eksekusi sedikit lebih lama dibandingkan regresi linear biasa, tetapi perbedaannya tidak signifikan. Selanjutnya, model ketiga *Linear Regression Ridge* dengan *Optimasi Hyperparameter* (LRR_OH) yaitu dengan melakukan optimasi *hyperparameter* menggunakan *GridSearchCV*, model ini mencapai RMSE yang sedikit lebih rendah dibandingkan model *Ridge* tanpa optimasi. Namun, waktu eksekusi meningkat karena proses pencarian *hyperparameter*. Model yang terakhir yaitu *Random Forest* (RF), memberikan hasil yang kurang memuaskan dengan RMSE yang jauh lebih tinggi dan waktu eksekusi juga jauh lebih lama dibandingkan dengan model linear.

3.3. Prediksi, Visualisasi, dan Evaluasi

Hasil prediksi yang telah didapatkan selanjutnya akan dibandingkan dengan harga saham yang sebenarnya melalui visualisasi grafik. Selain itu, kami menerapkan penggunaan warna yang berbeda agar hasil yang tersaji dapat lebih terbaca. Gambar 3 menunjukkan perbandingan antara harga saham sebenarnya (warna merah) dan harga saham prediksi (warna biru).



Gambar 3. Perbandingan grafik antara model prediksi harga saham *Netflix* dengan harga sebenarnya.

Tabel 3. Performa model jaringan saraf dengan iterasi berulang

Model	RMSE	$T(s)$	$\underline{T}(s)$
RNN_LSTM[1]	17.9362	410	4.1
GRU	14.2239	631	6.31

Dapat diamati pada Gambar 3 menampilkan perbandingan antara grafik harga prediksi saham *Netflix* dengan grafik harga sebenarnya untuk kelima model yang diuji. Grafik harga sebenarnya ditampilkan dengan garis berwarna merah, grafik ini menunjukkan fluktuasi harga saham *Netflix* yang sebenarnya dalam kurun waktu 23/02/2017 sampai 24/01/2022. Grafik harga prediksi ditampilkan dengan warna biru. Model LRR_OH dan model LRR adalah dua model yang memberikan prediksi paling mendekati harga saham *Netflix* sebenarnya. Model *Random Forest* menunjukkan hasil prediksi yang paling tidak mendekati harga sebenarnya.

4. DISKUSI

Pada langkah data preprocessing model *Recurrent Neural Network and Long Short Term Memory* (RNN-LSTM) [1] dan *Gated Recurrent Unit* (GRU), kedua nya sama-sama dilakukan *feature scaling* menggunakan *MinMaxScaler* agar data berada dalam rentang 0 hingga 1 (Tabel 3). Proses *MinMaxScaler* dari *library* *sklearn* untuk hasil yang konsisten. Normalisasi penting untuk mempercepat proses pelatihan model dan meningkatkan akurasi dengan mengurangi skala variabel yang berbeda.

Setelah normalisasi, dilakukan proses *reshaping* data untuk mengubah struktur data agar sesuai untuk pelatihan model. Pada pelatihan model, keduanya sama-sama dilatih selama 100 *epoch* dengan ukuran *batch* sebesar 37.

Model GRU memiliki kinerja sedikit lebih baik dibandingkan RNN_LSTM karena nilai RMSE yang lebih rendah, keduanya sama-sama memiliki arsitektur yang serupa dengan jumlah lapisan dan unit neuron yang sama. Namun, model RNN-LSTM memiliki waktu eksekusi yang lebih cepat daripada model GRU. RMSE yang lebih rendah pada model GRU menunjukkan keakuratannya yang lebih tinggi dalam memprediksi harga saham *Netflix*.

Untuk model jaringan saraf tanpa menggunakan iterasi perulangan yang hasilnya dapat diamati pada Tabel 2. Model LR cepat dalam eksekusi namun memiliki nilai RMSE yang relatif tinggi dibandingkan dengan model lainnya. Model LRR sedikit menurunkan nilai RMSE, namun hanya menunjukkan peningkatan kecil dalam akurasi prediksi. Model LRR_OH mencapai RMSE yang paling baik, namun waktu eksekusi meningkat karena proses pencarian *hyperparameter*. Model RF memberikan hasil yang kurang memuaskan dengan RMSE yang jauh lebih tinggi dan waktu eksekusi yang jauh lebih lama dibandingkan dengan model linear, menunjukkan bahwa RF tidak ideal untuk data ini.

5. KESIMPULAN

Berdasarkan metrik RMSE, model yang menunjukkan hasil paling optimal adalah *Linear Regression* dengan *Ridge* dan optimasi *hyperparameter*, yang menghasilkan nilai RMSE sebesar 13,8082. Sementara itu, jika dilihat dari segi waktu eksekusi, model yang memberikan hasil paling optimal adalah *Linear Regression* dengan waktu total sebesar 0,7717 detik. Meskipun model *Linear Regression* dengan *Ridge* dan optimasi *hyperparameter* memiliki kinerja yang lebih baik dalam hal akurasi prediksi, namun model *Linear Regression* paling efisien dalam hal waktu eksekusi.

Saran untuk penelitian selanjutnya adalah dapat menginvestigasi kombinasi metode regularisasi seperti Lasso atau ElasticNet untuk mengoptimalkan trade-off antara akurasi dan waktu eksekusi. Selain itu, eksplorasi teknik *ensemble* atau model non-linear seperti *decision trees* atau *support vector machines*, serta pembuatan fitur baru seperti indikator teknis atau analisis sentimen dari artikel berita juga dapat menjadi langkah berikutnya untuk memperbaiki performa prediksi secara keseluruhan.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Program Studi Informatika, Fakultas Teknologi Informasi dan Sains Data, Universitas Sebelas Maret, atas dukungan, fasilitas, dan sumber daya yang tak ternilai yang membuat penelitian ini dapat terlaksana. Tidak lupa, kami mengucapkan terima kasih kepada semua pihak atas dukungan dan wawasan yang sangat membantu dalam keberhasilan penelitian ini.

DAFTAR PUSTAKA

- [1] I. H, 'Netflix Stock Price Trend Prediction Using Recurrent Neural Network', *J. CoreIT J. Has. Penelit. Ilmu Komput. Dan Teknol. Inf.*, vol. 8, no. 2, p. 97, Dec. 2022, doi: 10.24014/coreit.v8i2.16599.
- [2] B. Gülmez, 'Stock price prediction with optimized deep LSTM network with artificial rabbits optimization algorithm', *Expert Syst. Appl.*, vol. 227, p. 120346, Oct. 2023, doi: 10.1016/j.eswa.2023.120346.
- [3] Department of Banking and Finance, College of Management and Social sciences Rhema University Aba, Nigeria and Dr. O. U. Godfrey, 'Money Supply and Stock Prices – A Case Study of Nigeria', *J. Econ. Finance Manag. Stud.*, vol. 04, no. 10, Oct. 2021, doi: 10.47191/jefms/v4-i10-10.
- [4] E. A. Shams and A. Rizaner, 'A novel support vector machine based intrusion detection system for mobile ad hoc networks', *Wirel. Netw.*, vol. 24, no. 5, pp. 1821–1829, Jul. 2018, doi: 10.1007/s11276-016-1439-0.
- [5] J. Li, W. Chen, Z. Zhou, J. Yang, and D. Zeng, 'DeepAR-Attention probabilistic prediction for stock price series', *Neural Comput. Appl.*, May 2024, doi: 10.1007/s00521-024-09916-3.
- [6] M. A. D. Suyudi, E. C. Djamal, and A. Maspupah, 'Prediksi Harga Saham menggunakan Metode Recurrent Neural Network', *Semin. Nas. Apl. Teknol. Inf. SNATI*, Jul. 2019.
- [7] D. C. Montgomery, E. A. Peck, and G. G. Vining, *Introduction to Linear Regression Analysis*. Wiley-Interscience, 2001.
- [8] D. S. Ramadhansyah, 'Perbandingan Metode Seleksi Fitur Filter, Wrapper, Dan Embedded Prediksi Kandungan Vitamin C Pada Buah Mangga Menggunakan Metode Linear Regression Dan Random Forest Regression', *Dspace.uui.ac.id*, 2022.
- [9] G. Bathla, R. Rani, and H. Aggarwal, 'Stocks of year 2020: prediction of high variations in stock prices using LSTM', *Multimed. Tools Appl.*, vol. 82, no. 7, pp. 9727–9743, Mar. 2023, doi: 10.1007/s11042-022-12390-5.
- [10] S. Behera, C. Prakash, and N. Sharma, 'A Combined Model for INDEX Price Forecasting Using LSTM, RNN, and GRU', in *Advances in Data-Driven Computing and Intelligent Systems*, vol. 890, S. Das, S. Saha, C. A. C. Coello, H. Rathore, and J. C. Bansal, Eds., in *Lecture Notes in Networks and Systems*, vol. 890. , Singapore: Springer Nature Singapore, 2024, pp. 499–514. doi: 10.1007/978-981-99-9531-8_40.
- [11] C. Che, H. Hu, X. Zhao, S. Li, and Q. Lin, 'Advancing Cancer Document Classification with R andom Forest', *Acad. J. Sci. Technol.*, vol. 8, no. 1, pp. 278–280, Nov. 2023, doi: 10.54097/ajst.v8i1.14333.
- [12] N. Kumar, R. Chauhan, and G. Dubey, 'Forecasting of Stock Price Using LSTM and Prophet Algorithm', in *Applications of Artificial Intelligence and Machine Learning*, vol. 778, A. Choudhary, A. P. Agrawal, R. Logeswaran, and B. Unhelkar, Eds., in *Lecture Notes in Electrical Engineering*, vol. 778. , Singapore: Springer Singapore, 2021, pp. 141–155. doi: 10.1007/978-981-16-3067-5_12.

- [13] V. Vimal, A. Singh, and P. Meel, 'Human Activity Recognition Using Long Short-Term Memory', in Proceedings of 6th International Conference on Recent Trends in Computing, vol. 177, R. P. Mahapatra, B. K. Panigrahi, B. K. Kaushik, and S. Roy, Eds., in Lecture Notes in Networks and Systems, vol. 177, Singapore: Springer Singapore, 2021, pp. 613–623. doi: 10.1007/978-981-33-4501-0_57.
- [14] H. Abbasimehr and R. Paki, 'Improving time series forecasting using LSTM and attention models', *J. Ambient Intell. Humaniz. Comput.*, vol. 13, no. 1, pp. 673–691, Jan. 2022, doi: 10.1007/s12652-020-02761-x.
- [15] F. Gao, J. Zhang, C. Zhang, S. Xu, and C. Ma, 'Long Short-Term Memory Networks with Multiple Variables for Stock Market Prediction', *Neural Process. Lett.*, vol. 55, no. 4, pp. 4211–4229, Aug. 2023, doi: 10.1007/s11063-022-11037-8.
- [16] S. Siami-Namini, N. Tavakoli, and A. Siami Namin, 'A Comparison of ARIMA and LSTM in Forecasting Time Series', in 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA), Orlando, FL: IEEE, Dec. 2018, pp. 1394–1401. doi: 10.1109/ICMLA.2018.00227.
- [17] D. P. Mishra, H. K. Gupta, G. Saajith, and R. Bag, 'Optimizing Heart Disease Prediction Model with GridsearchCV for Hyperparameter Tuning', in 2024 1st International Conference on Cognitive, Green and Ubiquitous Computing (IC-CGU), Bhubaneswar, India: IEEE, Mar. 2024, pp. 1–6. doi: 10.1109/IC-CGU58078.2024.10530772.
- [18] L. Yang and A. Shami, 'On hyperparameter optimization of machine learning algorithms: Theory and practice', *Neurocomputing*, vol. 415, pp. 295–316, Nov. 2020, doi: 10.1016/j.neucom.2020.07.061.
- [19] Y. I. Kurniawan, A. Rahmawati, N. Chasanah, and A. Hanifa, 'Application for determining the modality preference of student learning', *J. Phys. Conf. Ser.*, vol. 1367, no. 1, p. 012011, Nov. 2019, doi: 10.1088/1742-6596/1367/1/012011.
- [20] Y. I. Kurniawan, E. Soviana, and I. Yuliana, 'Merging Pearson Correlation and TAN-ELR algorithm in recommender system', presented at the HUMAN-DEDICATED SUSTAINABLE PRODUCT AND PROCESS DESIGN: MATERIALS, RESOURCES, AND ENERGY: Proceedings of the 4th International Conference on Engineering, Technology, and Industrial Application (ICETIA) 2017, Surakarta, Indonesia, 2018, p. 040028. doi: 10.1063/1.5042998.
- [21] Y. Guo, S. Han, Y. Li, C. Zhang, and Y. Bai, 'K-Nearest Neighbor combined with guided filter for hyperspectral image classification', *Procedia Comput. Sci.*, vol. 129, pp. 159–165, 2018, doi: 10.1016/j.procs.2018.03.066.
- [22] M. Obthong, N. Tantisantiwong, W. Jeamwattananachai, and G. Wills, 'A Survey on Machine Learning for Stock Price Prediction: Algorithms and Techniques', in Proceedings of the 2nd International Conference on Finance, Economics, Management and IT Business, Prague, Czech Republic: SCITEPRESS - Science and Technology Publications, 2020, pp. 63–71. doi: 10.5220/0009340700630071.
- [23] I. Nabillah and I. Ranggadara, 'Mean Absolute Percentage Error untuk Evaluasi Hasil Prediksi Komoditas Laut', *JOINS J. Inf. Syst.*, vol. 5, no. 2, pp. 250–255, Nov. 2020, doi: 10.33633/joins.v5i2.3900.